Open Source Used In HULK 13

Cisco Systems, Inc.
www.cisco.com

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

Text Part Number: 78EE117C99-191778193

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

In your requests please include the following reference number 78EE117C99-191778193

Contents

1.1 acitoolkit 0.4
1.2 ansible 2.4.2.0 :1.el7
1.2.1 Available under license
1.3 arrow 0.12.1
1.3.1 Available under license
1.4 boto 2.49.0
1.5 boto3 1.9.70
1.5.1 Available under license
1.6 elasticsearch 2.4.6
1.7 elasticsearch 6.5.4
1.7.1 Available under license
1.8 epel-release 7-11
1.9 erlang 20.1.7
1.9.1 Available under license
1.10 Flask 1.0.2
1.10.1 Available under license
1.11 gcc 4.8.5 :28.el7
1.11.1 Available under license
1.12 git 1.8.3.1
1.12.1 Available under license
1.13 graphviz 2.30.1 :21.el7
1.13.1 Available under license
1.14 httpd-tools 2.4.6 :67.el7_4.6
1.14.1 Available under license
1.15 httpd-tools/GPLv3 2.4.6 :67.el7_4.6
1.15.1 Available under license

1.16 kafka 1.3.5
1.16.1 Available under license
1.17 kibana 6.5.4
1.17.1 Available under license
1.18 logstash 6.5.4
1.18.1 Available under license
1.19 mongodb-org-shell 4.0.5
1.19.1 Available under license
1.20 MySQL Community Server 5.7 5.7 :21
1.20.1 Available under license
1.21 mysql-connector-java 5.1.45
1.21.1 Available under license
1.22 nginx 1.14.2
1.22.1 Available under license
1.23 pygraphviz 1.5
1.23.1 Available under license
1.24 pymongo 3.7.2
1.24.1 Available under license
1.25 rabbitmq-server 3.7.9
1.25.1 Available under license
1.26 redhat-lsb-core 4.1 :27.el7
1.26.1 Available under license
1.27 slowhttptest 1.7
1.27.1 Available under license
1.28 tetpyclient 1.0.7
1.28.1 Available under license
1.29 tomcat-native 1.1.34 :1.el7
1.29.1 Available under license
1.30 wget 1.19.5
1.30.1 Available under license

1.1 acitoolkit 0.4

1.2 ansible 2.4.2.0 :1.el7

1.2.1 Available under license :

From RPM File Metadata:GPLv3+

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

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

To protect your rights, we need to prevent others from denying you
these rights or asking you to surrender the rights. Therefore, you have
certain responsibilities if you distribute copies of the software, or if
you modify it: responsibilities to respect the freedom of others.

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

Developers that use the GNU GPL protect your rights with two steps:
(1) assert copyright on the software, and (2) offer you this License
giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains
that there is no warranty for this free software. For both users' and
authors' sake, the GPL requires that modified versions be marked as

changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

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

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a

computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically

linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or

modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an

"aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and

protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that

material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under

this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or

hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment

to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

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

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software

Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

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

16. Limitation of Liability.

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

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

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

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

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

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

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

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

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

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

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

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you

may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.

(c) 2012-2014, Michael DeHaan <michael.dehaan@gmail.com>
#
This file is part of Ansible
#
Ansible is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
#
Ansible is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
#
You should have received a copy of the GNU General Public License
along with Ansible. If not, see <http://www.gnu.org/licenses/>.

# Make coding more python3-ish
from __future__ import (absolute_import, division, print_function)
__metaclass__ = type

__version__ = '2.4.2.0'
__author__ = 'Ansible, Inc.'
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

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

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

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

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and

modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not

convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product

model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own

removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions;

the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone

to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey

the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

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

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

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM

IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

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

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

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

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

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

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

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

GNU General Public License for more details.

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

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

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

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

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.

1.3 arrow 0.12.1

1.3.1 Available under license :

Copyright (c) 2011 Vimalkumar Velayudhan

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

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

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>
Copyright 2013 Chris Smith

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

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

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

1.4 boto 2.49.0

1.5 boto3 1.9.70

1.5.1 Available under license :

Copyright 2013-2017 Amazon.com, Inc. or its affiliates. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License"). You
may not use this file except in compliance with the License. A copy of
the License is located at

http://aws.amazon.com/apache2.0/

or in the "license" file accompanying this file. This file is
distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF
ANY KIND, either express or implied. See the License for the specific
language governing permissions and limitations under the License.

1.6 elasticsearch 2.4.6

1.7 elasticsearch 6.5.4

1.7.1 Available under license :

Apache log4j
Copyright 2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache Lucene
Copyright 2014 The Apache Software Foundation

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

Includes software from other Apache Software Foundation projects,
including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license
and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such
as the Unicode Character Database. See http://unicode.org/copyright.html for more
details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is
BSD-licensed, created by Anders Mller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were
automatically generated with the moman/finenight FSA library, created by
Jean-Philippe Barrette-LaPierre. This library is available under an MIT license,
see http://sites.google.com/site/rrettesite/moman and
http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from
the Apache CXF project and is Apache License 2.0.

The Google Code Prettify is Apache License 2.0.
See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin
g Package (jaspell): http://jaspell.sourceforge.net/
License: The BSD License (http://www.opensource.org/licenses/bsd-license.php)

The snowball stemmers in
analysis/common/src/java/net/sf/snowball
were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in
analysis/common/src/resources/org/apache/lucene/analysis/snowball
were developed by Martin Porter and Richard Boulton.
The full snowball package is available from
http://snowball.tartarus.org/

The KStem stemmer in
analysis/common/src/org/apache/lucene/analysis/en
was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst)
under the BSD-license.

The Arabic,Persian,Romanian,Bulgarian, Hindi and Bengali analyzers (common) come with a default stopword list that is BSD-licensed created by Jacques Savoy. These files reside in:
analysis/common/src/resources/org/apache/lucene/analysis/ar/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/fa/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/ro/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/bg/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/hi/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/bn/stopwords.txt
See http://members.unine.ch/jacques.savoy/clef/index.html.

The German,Spanish,Finnish,French,Hungarian,Italian,Portuguese,Russian and Swedish light stemmers (common) are based on BSD-licensed reference implementations created by Jacques Savoy and Ljiljana Dolamic. These files reside in:
analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fi/FinnishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/sv/SwedishLightStemmer.java

The Stempel analyzer (stempel) includes BSD-licensed software developed
by the Egothor project http://egothor.sf.net/, created by Leo Galambos, Martin Kvapil,
and Edmond Nolan.

The Polish analyzer (stempel) comes with a default
stopword list that is BSD-licensed created by the Carrot2 project. The file resides
in stempel/src/resources/org/apache/lucene/analysis/pl/stopwords.txt.
See http://project.carrot2.org/license.html.

The SmartChineseAnalyzer source code (smartcn) was
provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_*.java (under modules/analysis/common/src/test/)
is derived from Unicode data such as the Unicode Character Database.
See http://unicode.org/copyright.html for more details.

The Morfologik analyzer (morfologik) includes BSD-licensed software
developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).

Morfologik uses data from Polish ispell/myspell dictionary
(http://www.sjp.pl/slownik/en/) licenced on the terms of (inter alia)
LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP)
(http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-*.jar are under the CDDL license, the original
source code for this can be found at http://www.eclipse.org/jetty/downloads.php

==========
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
==========

This software includes a binary and/or source version of data from

mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

==========
mecab-ipadic-2.7.0-20070801 Notice
==========

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the
research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
implied, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
the program and the nonexistence of any infringement or violation of
any right of any third party.

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
"use" as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.

In the case where the program, whether in its original form or

modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.
// ----------
// Transitive dependencies of this project determined from the
// maven pom organized by organization.
// ----------

Apache Extras for Apache log4j.

From: 'an unknown organization'
- geronimo-jms_1.1_spec org.apache.geronimo.specs:geronimo-jms_1.1_spec:jar:1.0

From: 'Apache Software Foundation' (http://www.apache.org)
- Apache Log4j (http://logging.apache.org/log4j/1.2/) log4j:log4j:bundle:1.2.17
License: The Apache Software License, Version 2.0 (http://www.apache.org/licenses/LICENSE-2.0.txt)

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

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

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

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

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

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

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

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

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

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

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

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

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

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

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

Copyright 2010-2013, Carrot Search s.c., Boznicza 11/56, Poznan, Poland

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

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

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

==========
= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =
==========
This product includes software developed by
Joda.org (http://www.joda.org/).

About This Content

May 22, 2015

License

The Eclipse Foundation makes available all content in this plug-in ("Content"). Unless otherwise indicated below, the
Content is provided to you under the terms and conditions of the Apache License, Version 2.0. A copy of the Apache
License, Version 2.0 is available at http://www.apache.org/licenses/LICENSE-2.0.txt

If you did not receive this Content directly from the Eclipse Foundation, the Content is being redistributed by another
party ("Redistributor") and different terms and conditions may apply to your use of any object code in the Content. Check the Redistributors license that was provided with the Content. If no such license exists, contact the Redistributor. Unless otherwise indicated below, the terms and conditions of the Apache License, Version 2.0 still apply
to any source code in the Content and such source code may be obtained at
http://www.eclipse.org](http://www.eclipse.org.
The code in this repository code was Written by Gil Tene, Michael Barker,
and Matt Warren, and released to the public domain, as explained at
http://creativecommons.org/publicdomain/zero/1.0/

For users of this code who wish to consume it under the "BSD" license
rather than under the public domain or CC0 contribution text mentioned
above, the code found under this directory is *also* provided under the
following license (commonly referred to as the BSD 2-Clause License). This
license does not detract from the above stated release of the code into
the public domain, and simply represents an additional license granted by
the Author.

** Beginning of "BSD 2-Clause License" text. **

Copyright (c) 2012, 2013, 2014 Gil Tene
Copyright (c) 2014 Michael Barker
Copyright (c) 2014 Matt Warren
All rights reserved.

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

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

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

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

THE POSSIBILITY OF SUCH DAMAGE.
The code for the t-digest was originally authored by Ted Dunning

A number of small but very helpful changes have been contributed by Adrien Grand (https://github.com/jpountz)

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

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

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

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

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

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

meet the following conditions:

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

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

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

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

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

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

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright 1999-2005 The Apache Software Foundation

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

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

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

Eclipse Foundation Software User Agreement

April 9, 2014

Usage Of Content

THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE
PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR
THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE
THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE
AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT
AND THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW, THEN YOU MAY
NOT USE THE CONTENT.

Applicable Licenses

Unless otherwise indicated, all Content made available by the Eclipse Foundation is provided to you under the terms and
conditions of the Eclipse Public License Version 1.0 ("EPL"). A copy of the EPL is provided with this Content and is
also available at http://www.eclipse.org/legal/epl-v10.html. For purposes of the EPL, "Program" will mean the Content.

Content includes, but is not limited to, source code, object code, documentation and other files maintained in the Eclipse Foundation source code repository ("Repository") in software modules ("Modules") and made available as downloadable archives ("Downloads").

* Content may be structured and packaged into modules to facilitate delivering, extending, and upgrading the

Content.
Typical modules may include plug-ins ("Plug-ins"), plug-in fragments ("Fragments"), and features ("Features").
* Each Plug-in or Fragment may be packaged as a sub-directory or JAR (Java ARchive) in a directory named "plugins".
* A Feature is a bundle of one or more Plug-ins and/or Fragments and associated material. Each Feature may be packaged
as a sub-directory in a directory named "features". Within a Feature, files named "feature.xml" may contain a list of the names and version numbers of the Plug-ins and/or Fragments associated with that Feature.
* Features may also include other Features ("Included Features"). Within a Feature, files named "feature.xml" may contain a list of the names and version numbers of Included Features.

The terms and conditions governing Plug-ins and Fragments should be contained in files named "about.html" ("Abouts").
The terms and conditions governing Features and Included Features should be contained in files named "license.html"
("Feature Licenses"). Abouts and Feature Licenses may be located in any directory of a Download or Module including, but
not limited to the following locations:

* The top-level (root) directory
* Plug-in and Fragment directories
* Inside Plug-ins and Fragments packaged as JARs
* Sub-directories of the directory named "src" of certain Plug-ins
* Feature directories

Note: if a Feature made available by the Eclipse Foundation is installed using the Provisioning Technology (as defined
below), you must agree to a license ("Feature Update License") during the installation process. If the Feature contains
Included Features, the Feature Update License should either provide you with the terms and conditions governing the
Included Features or inform you where you can locate them. Feature Update Licenses may be found in the "license" property of files named "feature.properties" found within a Feature. Such Abouts, Feature Licenses, and Feature Update
Licenses contain the terms and conditions (or references to such terms and conditions) that govern your use of the associated Content in that directory.

THE ABOUTS, FEATURE LICENSES, AND FEATURE UPDATE LICENSES MAY REFER TO THE EPL OR OTHER LICENSE AGREEMENTS, NOTICES OR
TERMS AND CONDITIONS. SOME OF THESE OTHER LICENSE AGREEMENTS MAY INCLUDE (BUT ARE NOT LIMITED TO):

* Eclipse Distribution License Version 1.0 (available at http://www.eclipse.org/licenses/edl-v10.html)
* Common Public License Version 1.0 (available at http://www.eclipse.org/legal/cpl-v10.html)
* Apache Software License 1.1 (available at http://www.apache.org/licenses/LICENSE)
* Apache Software License 2.0 (available at http://www.apache.org/licenses/LICENSE-2.0)
* Mozilla Public License Version 1.1 (available at http://www.mozilla.org/MPL/MPL-1.1.html)

IT IS YOUR OBLIGATION TO READ AND ACCEPT ALL SUCH TERMS AND CONDITIONS PRIOR TO USE OF THE CONTENT. If no About, Feature
License, or Feature Update License is provided, please contact the Eclipse Foundation to determine what terms and conditions govern that particular Content.

Use of Provisioning Technology

The Eclipse Foundation makes available provisioning software, examples of which include, but are not limited to, p2 and
the Eclipse Update Manager ("Provisioning Technology") for the purpose of allowing users to install software, documentation, information and/or other materials (collectively "Installable Software"). This capability is provided with the intent of allowing such users to install, extend and update Eclipse-based products. Information about packaging
Installable Software is available at http://eclipse.org/equinox/p2/repository_packaging.html ("Specification").

You may use Provisioning Technology to allow other parties to install Installable Software. You shall be responsible for
enabling the applicable license agreements relating to the Installable Software to be presented to, and accepted by, the
users of the Provisioning Technology in accordance with the Specification. By using Provisioning Technology in such a
manner and making it available in accordance with the Specification, you further acknowledge your agreement to, and the
acquisition of all necessary rights to permit the following:

1. A series of actions may occur ("Provisioning Process") in which a user may execute the Provisioning Technology on a
   machine ("Target Machine") with the intent of installing, extending or updating the functionality of an
   Eclipse-based product.
2. During the Provisioning Process, the Provisioning Technology may cause third party Installable Software or a portion
   thereof to be accessed and copied to the Target Machine.
3. Pursuant to the Specification, you will provide to the user the terms and conditions that govern the use of the
   Installable Software ("Installable Software Agreement") and such Installable Software Agreement shall be accessed
   from the Target Machine in accordance with the Specification. Such Installable Software Agreement must inform the
   user of the terms and conditions that govern the Installable Software and must solicit acceptance by the end user in
   the manner prescribed in such Installable Software Agreement. Upon such indication of agreement by the user, the
   provisioning Technology will complete installation of the Installable Software.

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import,
possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software,
please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of

encryption software, to see if this is permitted.

Java and all Java-based trademarks are trademarks of Oracle Corporation in the United States, other countries, or both.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

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

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

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

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

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

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

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

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

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

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

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

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

origin of the Work and reproducing the content of the NOTICE file.

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright {yyyy} {name of copyright owner}

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

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

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

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

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

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

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

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

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

5. Submission of Contributions. Unless You explicitly state otherwise,

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

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

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

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

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

END OF TERMS AND CONDITIONS

Apache License

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions

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

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

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

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

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

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

(b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

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

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

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

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

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

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

Eclipse Distribution License - v 1.0

Copyright (c) 2007, Eclipse Foundation, Inc. and its licensors.

All rights reserved.

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

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

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

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

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

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

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

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

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

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

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

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

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

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

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

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

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

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

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

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

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

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

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

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

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

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

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from unicode conversion examples available at
http://www.unicode.org/Public/PROGRAMS/CVTUTF. Here is the copyright
from those sources:

```
/*
* Copyright 2001-2004 Unicode, Inc.
*
* Disclaimer
*
* This source code is provided as is by Unicode, Inc. No claims are
* made as to fitness for any particular purpose. No warranties of any
* kind are expressed or implied. The recipient agrees to determine
* applicability of information provided. If this file has been
* purchased on magnetic or optical media from Unicode, Inc., the
* sole remedy for any claim will be exchange of defective media
* within 90 days of receipt.
*
* Limitations on Rights to Redistribute This Code
*
* Unicode, Inc. hereby grants the right to freely use the information
* supplied in this file in the creation of products supporting the
* Unicode Standard, and to make copies of this file in any form
* for internal or external distribution as long as this notice
* remains attached.
*/
```

Some code in core/src/java/org/apache/lucene/util/ArrayUtil.java was
derived from Python 2.4.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/2.4.2/license/

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from Python 3.1.2 sources available at
http://www.python.org. Full license is here:

http://www.python.org/download/releases/3.1.2/license/

Some code in core/src/java/org/apache/lucene/util/automaton was
derived from Brics automaton sources available at
www.brics.dk/automaton/. Here is the copyright from those sources:

```
/*
```

* Copyright (c) 2001-2009 Anders Moeller
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

The levenshtein automata tables in core/src/java/org/apache/lucene/util/automaton
were automatically generated with the moman/finenight FSA package.
Here is the copyright for those sources:

Copyright (c) 2010, Jean-Philippe Barrette-LaPierre, <jpb@rrette.com>
#
Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:
#
The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.
#
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT

HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Some code in core/src/java/org/apache/lucene/util/UnicodeUtil.java was
derived from ICU (http://www.icu-project.org)
The full license is available here:
http://source.icu-project.org/repos/icu/icu/trunk/license.html

/*
* Copyright (C) 1999-2010, International Business Machines
* Corporation and others. All Rights Reserved.
*
* Permission is hereby granted, free of charge, to any person obtaining a copy
* of this software and associated documentation files (the "Software"), to deal
* in the Software without restriction, including without limitation the rights
* to use, copy, modify, merge, publish, distribute, and/or sell copies of the
* Software, and to permit persons to whom the Software is furnished to do so,
* provided that the above copyright notice(s) and this permission notice appear
* in all copies of the Software and that both the above copyright notice(s) and
* this permission notice appear in supporting documentation.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
* IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE
* LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR
* ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER
* IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT
* OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
*
* Except as contained in this notice, the name of a copyright holder shall not
* be used in advertising or otherwise to promote the sale, use or other
* dealings in this Software without prior written authorization of the
* copyright holder.
*/

The following license applies to the Snowball stemmers:

Copyright (c) 2001, Dr Martin Porter
Copyright (c) 2002, Richard Boulton
All rights reserved.

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

* Redistributions of source code must retain the above copyright notice,

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

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

The following license applies to the KStemmer:

Copyright 2003,
Center for Intelligent Information Retrieval,
University of Massachusetts, Amherst.
All rights reserved.

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

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

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

3. The names "Center for Intelligent Information Retrieval" and "University of Massachusetts" must not be used to endorse or promote products derived from this software without prior written permission. To obtain permission, contact info@ciir.cs.umass.edu.

THIS SOFTWARE IS PROVIDED BY UNIVERSITY OF MASSACHUSETTS AND OTHER CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The following license applies to the Morfologik project:

Copyright (c) 2006 Dawid Weiss
Copyright (c) 2007-2011 Dawid Weiss, Marcin Mikowski
All rights reserved.

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

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

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

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

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

The dictionary comes from Morfologik project. Morfologik uses data from Polish ispell/myspell dictionary hosted at http://www.sjp.pl/slownik/en/ and is licenced on the terms of (inter alia) LGPL and Creative Commons ShareAlike. The part-of-speech tags were added in Morfologik project and are not found in the data from sjp.pl. The tagset is similar to IPI PAN tagset.

The following license applies to the Morfeusz project,

used by org.apache.lucene.analysis.morfologik.

BSD-licensed dictionary of Polish (SGJP)
http://sgjp.pl/morfeusz/

Copyright 2011 Zygmunt Saloni, Wodzimierz Gruszczyski,
Marcin Woliski, Robert Woosz

All rights reserved.

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

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

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

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

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

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

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

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

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

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

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

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

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

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

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

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

ACKNOWLEDGEMENT
==========

HPPC borrowed code, ideas or both from:

* Apache Lucene, http://lucene.apache.org/
  (Apache license)
* Fastutil, http://fastutil.di.unimi.it/
  (Apache license)
* Koloboke, https://github.com/OpenHFT/Koloboke
  (Apache license)
Elasticsearch
Copyright 2009-2018 Elasticsearch

This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).
this is a test notice file
this is a test license file
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*   http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/
Source code in this repository is variously licensed under the Apache License
Version 2.0, an Apache compatible license, or the Elastic License. Outside of
the "x-pack" folder, source code in a given file is licensed under the Apache
License Version 2.0, unless otherwise noted at the beginning of the file or a
LICENSE file present in the directory subtree declares a separate license.
Within the "x-pack" folder, source code in a given file is licensed under the
Elastic License, unless otherwise noted at the beginning of the file or a
LICENSE file present in the directory subtree declares a separate license.

The build produces two sets of binaries - one set that falls under the Elastic
License and another set that falls under Apache License Version 2.0. The
binaries that contain `-oss` in the artifact name are licensed under the Apache

License Version 2.0.
This package is free to use under the Elastic license. It contains open source
and free commercial features and access to paid commercial features.
{stack-ov}/license-management.html[Start a 30-day trial] to try out all of the
paid commercial features. See the
https://www.elastic.co/subscriptions[Subscriptions] page for information about
Elastic license levels.
[role="xpack"]
[[license-settings]]
=== {xpack} License Settings
++++
<titleabbrev>License Settings</titleabbrev>
++++

You can configure this licensing setting in the `elasticsearch.yml` file.
For more information, see
{xpack-ref}/license-management.html[{xpack} License Management].

`xpack.license.self_generated.type`::
Set to `basic` (default) to enable basic {xpack} features. +
+
--
If set to `trial`, the self-generated license gives access only to all the features
of a x-pack for 30 days. You can later downgrade the cluster to a basic license if
needed.
--
[role="xpack"]
[testenv="basic"]
[[update-license]]
=== Update License API

This API enables you to update your license.

[float]
==== Request

`PUT _xpack/license`

[float]
==== Description

You can update your license at runtime without shutting down your nodes.
License updates take effect immediately. If the license you are installing does
not support all of the features that were available with your previous license,
however, you are notified in the response. You must then re-submit the API
request with the `acknowledge` parameter set to `true`.

For more information about the different types of licenses, see

https://www.elastic.co/subscriptions.

[float]
==== Query Parameters

`acknowledge`::
(boolean) Specifies whether you acknowledge the license changes. The default value is `false`.

[float]
==== Request Body

`licenses`::
(array) A sequence of one or more JSON documents containing the license information.

[float]
==== Authorization

If {security} is enabled, you need `manage` cluster privileges to install the license.

If {security} is enabled and you are installing a gold or platinum license, you must enable TLS on the transport networking layer before you install the license. See <<configuring-tls>>.

[float]
==== Examples

The following example updates to a basic license:

[source,js]
```
----------------------------------------------------------
POST _xpack/license
{
 "licenses": [
   {
    "uid":"893361dc-9749-4997-93cb-802e3d7fa4xx",
    "type":"basic",
    "issue_date_in_millis":1411948800000,
    "expiry_date_in_millis":1914278399999,
    "max_nodes":1,
    "issued_to":"issuedTo",
    "issuer":"issuer",
    "signature":"xx"
   }
   ]
```

}

// CONSOLE
// TEST[skip:license testing issues]

NOTE: These values are invalid; you must substitute the appropriate content
from your license file.

You can alternatively use a `curl` command, for example:

[source,js]
[source,shell]
---
curl -XPUT -u <user> 'http://<host>:<port>/_xpack/license' -H "Content-Type: application/json" -d @license.json
---
// NOTCONSOLE

On Windows machine, use the following command:

[source,shell]
---
gc .\license.json | Invoke-WebRequest -uri http://<host>:<port>/_xpack/license -Credential elastic -Method Put -
ContentType "application/json"

In these examples,

* `<user>` is a user ID with the appropriate authority.
* `<host>` is the hostname of the {es} node (`localhost` if executing
 locally)
* `<port>` is the http port (defaults to `9200`)
* `license.json` is the license JSON file

NOTE: If your {es} node has SSL enabled on the HTTP interface, you must
 start your URL with `https://`

If you previously had a license with more features than the basic license, you
receive the following response:

[source,js]

 {
 "acknowledged": false,
 "license_status": "valid",
 "acknowledge": {
   "message": """This license update requires acknowledgement. To acknowledge the license, please read the
following messages and update the license again, this time with the "acknowledge=true" parameter:""",
   "watcher": [

```
    "Watcher will be disabled"
  ],
  "logstash": [
    "Logstash will no longer poll for centrally-managed pipelines"
  ],
  "security": [
    "The following X-Pack security functionality will be disabled: ..." ]
  }
}
-----------------------------------------------------------
// NOTCONSOLE
```

To complete the update, you must re-submit the API request and set the
`acknowledge` parameter to `true`. For example:

```
[source,js]
-----------------------------------------------------------
POST _xpack/license?acknowledge=true
{
 "licenses": [
  {
    "uid":"893361dc-9749-4997-93cb-802e3d7fa4xx",
    "type":"basic",
    "issue_date_in_millis":1411948800000,
    "expiry_date_in_millis":1914278399999,
    "max_nodes":1,
    "issued_to":"issuedTo",
    "issuer":"issuer",
    "signature":"xx"
  }
  ]
}
-----------------------------------------------------------
// CONSOLE
// TEST[skip:license testing issues]
```

Alternatively:

```
[source,sh]
-----------------------------------------------------------
curl -XPUT -u elastic 'http://<host>:<port>/_xpack/license?acknowledge=true' -H "Content-Type: application/json"
-d @license.json
-----------------------------------------------------------
// NOTCONSOLE
```

For more information about the features that are disabled when you downgrade
your license, see {xpack-ref}/license-expiration.html[License Expiration].

[role="xpack"]
[testenv="basic"]
[[delete-license]]
=== Delete License API

This API enables you to delete licensing information.

[float]
==== Request

`DELETE /_xpack/license`

[float]
==== Description

When your license expires, {xpack} operates in a degraded mode. For more information, see {xpack-ref}/license-expiration.html[License Expiration].

[float]
==== Authorization

You must have `manage` cluster privileges to use this API.
For more information, see
{xpack-ref}/security-privileges.html[Security Privileges].

[float]
==== Examples

The following example queries the info API:

[source,js]
```
DELETE _xpack/license
```
// CONSOLE
// TEST[skip:license testing issues]

When the license is successfully deleted, the API returns the following response:
[source,js]
```
{
 "acknowledged": true
}
```
// NOTCONSOLE
[role="xpack"]
[testenv="basic"]
[[get-license]]

=== Get License API

This API enables you to retrieve licensing information.

[float]
==== Request

`GET /_xpack/license`

[float]
==== Description

This API returns information about the type of license, when it was issued, and
when it expires, for example.

For more information about the different types of licenses, see
https://www.elastic.co/subscriptions.

[float]
==== Query Parameters

`local`::
 (boolean) Specifies whether to retrieve local information. The default value
 is `false`, which means the information is retrieved from the master node.

[float]
==== Authorization

You must have `monitor` cluster privileges to use this API.
For more information, see
{xpack-ref}/security-privileges.html[Security Privileges].

[float]
==== Examples

The following example provides information about a trial license:

[source,js]

GET _xpack/license

// CONSOLE

[source,js]

```
{
 "license" : {
   "status" : "active",
   "uid" : "cbff45e7-c553-41f7-ae4f-9205eabd80xx",
   "type" : "trial",
   "issue_date" : "2018-10-20T22:05:12.332Z",
   "issue_date_in_millis" : 1540073112332,
   "expiry_date" : "2018-11-19T22:05:12.332Z",
   "expiry_date_in_millis" : 1542665112332,
   "max_nodes" : 1000,
   "issued_to" : "test",
   "issuer" : "elasticsearch",
   "start_date_in_millis" : -1
 }
}
--------------------------------------------------
// TESTRESPONSE[s/"cbff45e7-c553-41f7-ae4f-9205eabd80xx"/$body.license.uid/]
// TESTRESPONSE[s/"basic"/$body.license.type/]
// TESTRESPONSE[s/"2018-10-20T22:05:12.332Z"/$body.license.issue_date/]
// TESTRESPONSE[s/1540073112332/$body.license.issue_date_in_millis/]
// TESTRESPONSE[s/"2018-11-19T22:05:12.332Z"/$body.license.expiry_date/]
// TESTRESPONSE[s/1542665112332/$body.license.expiry_date_in_millis/]
// TESTRESPONSE[s/1000/$body.license.max_nodes/]
// TESTRESPONSE[s/"test"/$body.license.issued_to/]
// TESTRESPONSE[s/"elasticsearch"/$body.license.issuer/]
== License

Copyright 2013-2018 Elasticsearch

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

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

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

[[java-rest-high-delete-license-execution]]
==== Execution

The license can be deleted using the `deleteLicense()` method:
```

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[delete-license-execute]
--------------------------------------------------
```

[[java-rest-high-delete-license-response]]
==== Response

The returned `DeleteLicenseResponse` contains the `acknowledged` flag, which returns true if the request was processed by all nodes.

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[delete-license-response]
--------------------------------------------------
<1> Check the acknowledge flag. It should be true if license deletion is acknowledged.
```

[[java-rest-high-delete-license-async]]
==== Asynchronous Execution

This request can be executed asynchronously:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[delete-license-execute-async]
--------------------------------------------------
<1> The `DeleteLicenseRequest` to execute and the `ActionListener` to use when
the execution completes
```

The asynchronous method does not block and returns immediately. Once it is completed the `ActionListener` is called back using the `onResponse` method if the execution successfully completed or using the `onFailure` method if it failed.

A typical listener for `DeleteLicenseResponse` looks like:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[delete-license-execute-listener]
--------------------------------------------------
<1> Called when the execution is successfully completed. The response is
provided as an argument
<2> Called in case of failure. The raised exception is provided as an argument
```

[[java-rest-high-put-license]]
=== Update License

[[java-rest-high-put-license-execution]]
==== Execution

The license can be added or updated using the `putLicense()` method:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[put-license-execute]
--------------------------------------------------
<1> Set the categories of information to retrieve. The default is to
return no information which is useful for checking if {xpack} is installed
but not much else.
<2> A JSON document containing the license information.

[[java-rest-high-put-license-response]]
==== Response

The returned `PutLicenseResponse` contains the `LicensesStatus`,
`acknowledged` flag and possible acknowledge messages. The acknowledge messages
are present if you previously had a license with more features than one you
are trying to update and you didn't set the `acknowledge` flag to `true`. In this case
you need to display the messages to the end user and if they agree, resubmit the
license with the `acknowledge` flag set to `true`. Please note that the request will
still return a 200 return code even if requires an acknowledgement. So, it is
necessary to check the `acknowledged` flag.

["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[put-license-response]
--------------------------------------------------
<1> The status of the license
<2> Make sure that the license is valid.
<3> Check the acknowledge flag. It should be true if license is acknowledged.
<4> Otherwise we can see the acknowledge messages in `acknowledgeHeader()`
<5> and check component-specific messages in `acknowledgeMessages()`.

[[java-rest-high-put-license-async]]
==== Asynchronous Execution

This request can be executed asynchronously:

["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[put-license-execute-async]
--------------------------------------------------
<1> The `PutLicenseRequest` to execute and the `ActionListener` to use when
the execution completes

The asynchronous method does not block and returns immediately. Once it is
completed the `ActionListener` is called back using the `onResponse` method
```

if the execution successfully completed or using the `onFailure` method if
it failed.

A typical listener for `PutLicenseResponse` looks like:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[put-license-execute-listener]
--------------------------------------------------
<1> Called when the execution is successfully completed. The response is
provided as an argument
<2> Called in case of failure. The raised exception is provided as an argument
```

[[java-rest-high-get-license]]
=== Get License

[[java-rest-high-get-license-execution]]
==== Execution

The license can be added or updated using the `getLicense()` method:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[get-license-execute]
--------------------------------------------------
```

[[java-rest-high-get-license-response]]
==== Response

The returned `GetLicenseResponse` contains the license in the JSON format.

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[get-license-response]
--------------------------------------------------
<1> The text of the license.
```

[[java-rest-high-get-license-async]]
==== Asynchronous Execution

This request can be executed asynchronously:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[get-license-execute-async]
--------------------------------------------------
<1> The `GetLicenseRequest` to execute and the `ActionListener` to use when
the execution completes
```

The asynchronous method does not block and returns immediately. Once it is completed the `ActionListener` is called back using the `onResponse` method if the execution successfully completed or using the `onFailure` method if it failed.

A typical listener for `GetLicenseResponse` looks like:

```
["source","java",subs="attributes,callouts,macros"]
--------------------------------------------------
include-tagged::{doc-tests}/LicensingDocumentationIT.java[get-license-execute-listener]
--------------------------------------------------
<1> Called when the execution is successfully completed. The response is
provided as an argument
<2> Called in case of failure. The raised exception is provided as an argument
== Copyright and License
```

This software is Copyright (c) 2013-2018 by Elasticsearch BV.

This is free software, licensed under The Apache License Version 2.0.
ELASTIC LICENSE AGREEMENT

PLEASE READ CAREFULLY THIS ELASTIC LICENSE AGREEMENT (THIS "AGREEMENT"), WHICH CONSTITUTES A LEGALLY BINDING AGREEMENT AND GOVERNS ALL OF YOUR USE OF ALL OF THE ELASTIC SOFTWARE WITH WHICH THIS AGREEMENT IS INCLUDED ("ELASTIC SOFTWARE") THAT IS PROVIDED IN OBJECT CODE FORMAT, AND, IN ACCORDANCE WITH SECTION 2 BELOW, CERTAIN OF THE ELASTIC SOFTWARE THAT IS PROVIDED IN SOURCE CODE FORMAT. BY INSTALLING OR USING ANY OF THE ELASTIC SOFTWARE GOVERNED BY THIS AGREEMENT, YOU ARE ASSENTING TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH SUCH TERMS AND CONDITIONS, YOU MAY NOT INSTALL OR USE THE ELASTIC SOFTWARE GOVERNED BY THIS AGREEMENT. IF YOU ARE INSTALLING OR USING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY.

Posted Date: April 20, 2018

This Agreement is entered into by and between Elasticsearch BV ("Elastic") and You, or the legal entity on behalf of whom You are acting (as applicable, "You").

1. OBJECT CODE END USER LICENSES, RESTRICTIONS AND THIRD PARTY OPEN SOURCE SOFTWARE

1.1 Object Code End User License. Subject to the terms and conditions of Section 1.2 of this Agreement, Elastic hereby grants to You, AT NO CHARGE and for so long as you are not in breach of any provision of this Agreement, a License to the Basic Features and Functions of the Elastic Software.

1.2 Reservation of Rights; Restrictions. As between Elastic and You, Elastic and its licensors own all right, title and interest in and to the Elastic Software, and except as expressly set forth in Sections 1.1, and 2.1 of this Agreement, no other license to the Elastic Software is granted to You under this Agreement, by implication, estoppel or otherwise. You agree not to: (i) reverse engineer or decompile, decrypt, disassemble or otherwise reduce any Elastic Software provided to You in Object Code, or any portion thereof, to Source Code, except and only to the extent any such restriction is prohibited by applicable law, (ii) except as expressly permitted in this Agreement, prepare derivative works from, modify, copy or use the Elastic Software Object Code or the Commercial Software Source Code in any manner; (iii) except as expressly permitted in Section 1.1 above, transfer, sell, rent, lease, distribute, sublicense, loan or otherwise transfer, Elastic Software Object Code, in whole or in part, to any third party; (iv) use Elastic Software Object Code for providing time-sharing services, any software-as-a-service, service bureau services or as part of an application services provider or other service offering (collectively, "SaaS Offering") where obtaining access to the Elastic Software or the features and functions of the Elastic Software is a primary reason or substantial motivation for users of the SaaS Offering to access and/or use the SaaS Offering ("Prohibited SaaS Offering"); (v) circumvent the limitations on use of Elastic Software provided to You in Object Code format that are imposed or preserved by any License Key, or (vi) alter or remove any Marks and Notices in the Elastic Software. If You have any question as to whether a specific SaaS Offering constitutes a Prohibited SaaS Offering, or are interested in obtaining Elastic's permission to engage in commercial or non-commercial distribution of the Elastic Software, please contact elastic_license@elastic.co.

1.3 Third Party Open Source Software. The Commercial Software may contain or be provided with third party open source libraries, components, utilities and other open source software (collectively, "Open Source Software"), which Open Source Software may have applicable license terms as identified on a website designated by Elastic. Notwithstanding anything to the contrary herein, use of the Open Source Software shall be subject to the license terms and conditions applicable to such Open Source Software, to the extent required by the applicable licensor (which terms shall not restrict the license rights granted to You hereunder, but may contain additional rights). To the extent any condition of this Agreement conflicts with any license to the Open Source Software, the Open Source Software license will govern with respect to such Open Source Software only. Elastic may also separately provide you with certain open source software that is licensed by Elastic. Your use of such Elastic open source software will not be governed by this Agreement, but by the applicable open source license terms.

2. COMMERCIAL SOFTWARE SOURCE CODE

2.1 Limited License. Subject to the terms and conditions of Section 2.2 of this Agreement, Elastic hereby grants to You, AT NO CHARGE and for so long as

you are not in breach of any provision of this Agreement, a limited, non-exclusive, non-transferable, fully paid up royalty free right and license to the Commercial Software in Source Code format, without the right to grant or authorize sublicenses, to prepare Derivative Works of the Commercial Software, provided You (i) do not hack the licensing mechanism, or otherwise circumvent the intended limitations on the use of Elastic Software to enable features other than Basic Features and Functions or those features You are entitled to as part of a Subscription, and (ii) use the resulting object code only for reasonable testing purposes.

2.2 Restrictions. Nothing in Section 2.1 grants You the right to (i) use the Commercial Software Source Code other than in accordance with Section 2.1 above, (ii) use a Derivative Work of the Commercial Software outside of a Non-production Environment, in any production capacity, on a temporary or permanent basis, or (iii) transfer, sell, rent, lease, distribute, sublicense, loan or otherwise make available the Commercial Software Source Code, in whole or in part, to any third party. Notwithstanding the foregoing, You may maintain a copy of the repository in which the Source Code of the Commercial Software resides and that copy may be publicly accessible, provided that you include this Agreement with Your copy of the repository.

3. TERMINATION

3.1 Termination. This Agreement will automatically terminate, whether or not You receive notice of such Termination from Elastic, if You breach any of its provisions.

3.2 Post Termination. Upon any termination of this Agreement, for any reason, You shall promptly cease the use of the Elastic Software in Object Code format and cease use of the Commercial Software in Source Code format. For the avoidance of doubt, termination of this Agreement will not affect Your right to use Elastic Software, in either Object Code or Source Code formats, made available under the Apache License Version 2.0.

3.3 Survival. Sections 1.2, 2.2. 3.3, 4 and 5 shall survive any termination or expiration of this Agreement.

4. DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITY

4.1 Disclaimer of Warranties. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, THE ELASTIC SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, AND ELASTIC AND ITS LICENSORS MAKE NO WARRANTIES WHETHER EXPRESSED, IMPLIED OR STATUTORY REGARDING OR RELATING TO THE ELASTIC SOFTWARE. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, ELASTIC AND ITS LICENSORS SPECIFICALLY DISCLAIM ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT WITH RESPECT TO THE ELASTIC SOFTWARE, AND WITH RESPECT TO THE USE OF THE FOREGOING. FURTHER, ELASTIC DOES NOT WARRANT RESULTS OF USE OR THAT THE ELASTIC SOFTWARE WILL BE ERROR FREE OR THAT THE USE OF THE

ELASTIC SOFTWARE WILL BE UNINTERRUPTED.

4.2 Limitation of Liability. IN NO EVENT SHALL ELASTIC OR ITS LICENSORS BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY DIRECT OR INDIRECT DAMAGES, INCLUDING, WITHOUT LIMITATION, FOR ANY LOSS OF PROFITS, LOSS OF USE, BUSINESS INTERRUPTION, LOSS OF DATA, COST OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND, IN CONNECTION WITH OR ARISING OUT OF THE USE OR INABILITY TO USE THE ELASTIC SOFTWARE, OR THE PERFORMANCE OF OR FAILURE TO PERFORM THIS AGREEMENT, WHETHER ALLEGED AS A BREACH OF CONTRACT OR TORTIOUS CONDUCT, INCLUDING NEGLIGENCE, EVEN IF ELASTIC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

5. MISCELLANEOUS

This Agreement completely and exclusively states the entire agreement of the parties regarding the subject matter herein, and it supersedes, and its terms govern, all prior proposals, agreements, or other communications between the parties, oral or written, regarding such subject matter. This Agreement may be modified by Elastic from time to time, and any such modifications will be effective upon the "Posted Date" set forth at the top of the modified Agreement. If any provision hereof is held unenforceable, this Agreement will continue without said provision and be interpreted to reflect the original intent of the parties. This Agreement and any non-contractual obligation arising out of or in connection with it, is governed exclusively by Dutch law. This Agreement shall not be governed by the 1980 UN Convention on Contracts for the International Sale of Goods. All disputes arising out of or in connection with this Agreement, including its existence and validity, shall be resolved by the courts with jurisdiction in Amsterdam, The Netherlands, except where mandatory law provides for the courts at another location in The Netherlands to have jurisdiction. The parties hereby irrevocably waive any and all claims and defenses either might otherwise have in any such action or proceeding in any of such courts based upon any alleged lack of personal jurisdiction, improper venue, forum non conveniens or any similar claim or defense. A breach or threatened breach, by You of Section 2 may cause irreparable harm for which damages at law may not provide adequate relief, and therefore Elastic shall be entitled to seek injunctive relief without being required to post a bond. You may not assign this Agreement (including by operation of law in connection with a merger or acquisition), in whole or in part to any third party without the prior written consent of Elastic, which may be withheld or granted by Elastic in its sole and absolute discretion. Any assignment in violation of the preceding sentence is void. Notices to Elastic may also be sent to legal@elastic.co.

6. DEFINITIONS

The following terms have the meanings ascribed:

6.1 "Affiliate" means, with respect to a party, any entity that controls, is

controlled by, or which is under common control with, such party, where "control" means ownership of at least fifty percent (50%) of the outstanding voting shares of the entity, or the contractual right to establish policy for, and manage the operations of, the entity.

6.2 "Basic Features and Functions" means those features and functions of the Elastic Software that are eligible for use under a Basic license, as set forth at https://www.elastic.co/subscriptions, as may be modified by Elastic from time to time.

6.3 "Commercial Software" means the Elastic Software Source Code in any file containing a header stating the contents are subject to the Elastic License or which is contained in the repository folder labeled "x-pack", unless a LICENSE file present in the directory subtree declares a different license.

6.4 "Derivative Work of the Commercial Software" means, for purposes of this Agreement, any modification(s) or enhancement(s) to the Commercial Software, which represent, as a whole, an original work of authorship.

6.5 "License" means a limited, non-exclusive, non-transferable, fully paid up, royalty free, right and license, without the right to grant or authorize sublicenses, solely for Your internal business operations to (i) install and use the applicable Features and Functions of the Elastic Software in Object Code, and (ii) permit Contractors and Your Affiliates to use the Elastic software as set forth in (i) above, provided that such use by Contractors must be solely for Your benefit and/or the benefit of Your Affiliates, and You shall be responsible for all acts and omissions of such Contractors and Affiliates in connection with their use of the Elastic software that are contrary to the terms and conditions of this Agreement.

6.6 "License Key" means a sequence of bytes, including but not limited to a JSON blob, that is used to enable certain features and functions of the Elastic Software.

6.7 "Marks and Notices" means all Elastic trademarks, trade names, logos and notices present on the Documentation as originally provided by Elastic.

6.8 "Non-production Environment" means an environment for development, testing or quality assurance, where software is not used for production purposes.

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

6.10 "Source Code" means the preferred form of computer software for making modifications, including but not limited to software source code, documentation source, and configuration files.

6.11 "Subscription" means the right to receive Support Services and a License to the Commercial Software.

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

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

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

Joni is released under the MIT License.

JCodings is released under the MIT License.

/*

The MIT License

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

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

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

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

*/

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.

It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

Licensing

Jackson core and extension components may licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (http://fasterxml.com).

Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.
This copy of Jackson JSON processor streaming parser/generator is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0
The art of simplicity is a puzzle of complexity.

Overview
[YAML](http://yaml.org) is a data serialization format designed for human readability and interaction with scripting languages.

SnakeYAML is a YAML processor for the Java Virtual Machine.

SnakeYAML features

* a **complete** [YAML 1.1 processor](http://yaml.org/spec/1.1/current.html). In particular, SnakeYAML can parse all examples from the specification.
* Unicode support including UTF-8/UTF-16 input/output.
* high-level API for serializing and deserializing native Java objects.
* support for all types from the [YAML types repository](http://yaml.org/type/index.html).
* relatively sensible error messages.

Info
* [Changes](https://bitbucket.org/asomov/snakeyaml/wiki/Changes)
* [Documentation](https://bitbucket.org/asomov/snakeyaml/wiki/Documentation)

Contribute
* Mercurial DVCS is used to dance with the [source code](https://bitbucket.org/asomov/snakeyaml/src).

* If you find a bug in SnakeYAML, please [file a bug report](https://bitbucket.org/asomov/snakeyaml/issues?status=new&status=open).
* You may discuss SnakeYAML at [the mailing list](http://groups.google.com/group/snakeyaml-core).

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

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

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

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

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

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

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

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

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

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

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

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

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

origin of the Work and reproducing the content of the NOTICE file.

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

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

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

END OF TERMS AND CONDITIONS

Copyright (c) 2012 France Tlcom
All rights reserved.

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

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its

contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
[The "BSD license"]
Copyright (c) 2015 Terence Parr, Sam Harwell
All rights reserved.

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

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

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

The Netty Project
==========

Please visit the Netty web site for more information:

* http://netty.io/

Copyright 2011 The Netty Project

The Netty Project licenses this file to you under the Apache License,
version 2.0 (the "License"); you may not use this file except in compliance
with the License. You may obtain a copy of the License at:

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

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

Also, please refer to each LICENSE.<component>.txt file, which is located in
the 'license' directory of the distribution file, for the license terms of the
components that this product depends on.

This product contains the extensions to Java Collections Framework which has
been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

 * LICENSE:
   * license/LICENSE.jsr166y.txt (Public Domain)
 * HOMEPAGE:
   * http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/
   * http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jbosscache/experimental/jsr166/

This product contains a modified version of Robert Harder's Public Domain
Base64 Encoder and Decoder, which can be obtained at:

 * LICENSE:
   * license/LICENSE.base64.txt (Public Domain)
 * HOMEPAGE:
   * http://iharder.sourceforge.net/current/java/base64/

This product contains a modified version of 'JZlib', a re-implementation of
zlib in pure Java, which can be obtained at:

 * LICENSE:
   * license/LICENSE.jzlib.txt (BSD Style License)
 * HOMEPAGE:
   * http://www.jcraft.com/jzlib/

This product contains a modified version of 'Webbit', a Java event based
WebSocket and HTTP server:

 * LICENSE:

  * license/LICENSE.webbit.txt (BSD License)
 * HOMEPAGE:
  * https://github.com/joewalnes/webbit

This product optionally depends on 'Protocol Buffers', Google's data
interchange format, which can be obtained at:

 * LICENSE:
  * license/LICENSE.protobuf.txt (New BSD License)
 * HOMEPAGE:
  * http://code.google.com/p/protobuf/

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate
a temporary self-signed X.509 certificate when the JVM does not provide the
equivalent functionality. It can be obtained at:

 * LICENSE:
  * license/LICENSE.bouncycastle.txt (MIT License)
 * HOMEPAGE:
  * http://www.bouncycastle.org/

This product optionally depends on 'SLF4J', a simple logging facade for Java,
which can be obtained at:

 * LICENSE:
  * license/LICENSE.slf4j.txt (MIT License)
 * HOMEPAGE:
  * http://www.slf4j.org/

This product optionally depends on 'Apache Commons Logging', a logging
framework, which can be obtained at:

 * LICENSE:
  * license/LICENSE.commons-logging.txt (Apache License 2.0)
 * HOMEPAGE:
  * http://commons.apache.org/logging/

This product optionally depends on 'Apache Log4J', a logging framework,
which can be obtained at:

 * LICENSE:
  * license/LICENSE.log4j.txt (Apache License 2.0)
 * HOMEPAGE:
  * http://logging.apache.org/log4j/

This product optionally depends on 'JBoss Logging', a logging framework,
which can be obtained at:

* LICENSE:
  * license/LICENSE.jboss-logging.txt (GNU LGPL 2.1)
* HOMEPAGE:
  * http://anonsvn.jboss.org/repos/common/common-logging-spi/

This product optionally depends on 'Apache Felix', an open source OSGi
framework implementation, which can be obtained at:

* LICENSE:
  * license/LICENSE.felix.txt (Apache License 2.0)
* HOMEPAGE:
  * http://felix.apache.org/

Copyright 2010 RightTime, Inc.

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

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

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

Apache HttpComponents Client

Copyright 1999-2016 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

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

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

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

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

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

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

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

2. Grant of Copyright License. Subject to the terms and conditions of

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

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

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

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

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

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

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

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

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

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

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

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

has been advised of the possibility of such damages.

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

END OF TERMS AND CONDITIONS

==========

This project includes Public Suffix List copied from
<https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <http://mozilla.org/MPL/2.0/>

Mozilla Public License Version 2.0
==========

1. Definitions

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

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

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

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

1.5. "Incompatible With Secondary Licenses"
means

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

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

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

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

1.8. "License"
means this document.

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

1.10. "Modifications"
means any of the following:

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

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

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

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

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

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

2. License Grants and Conditions

2.1. Grants

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

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

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

2.2. Effective Date

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

2.3. Limitations on Grant Scope

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

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

(b) for infringements caused by: (i) Your and any other third party's

modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

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

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

2.4. Subsequent Licenses

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

2.5. Representation

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

2.6. Fair Use

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

2.7. Conditions

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

3. Responsibilities

3.1. Distribution of Source Form

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

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

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

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

3.3. Distribution of a Larger Work

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

3.4. Notices

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

3.5. Application of Additional Terms

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

4. Inability to Comply Due to Statute or Regulation

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

5. Termination

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

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

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

```
************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
```

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

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

8. Litigation
-------------

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

9. Miscellaneous
```

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

10. Versions of the License

10.1. New Versions

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

10.2. Effect of New Versions

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

10.3. Modified Versions

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

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

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

Exhibit A - Source Code Form License Notice

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

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

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as
defined by the Mozilla Public License, v. 2.0.
Apache Commons Codec
Copyright 2002-2014 The Apache Software Foundation

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

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==========

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/
package org.elasticsearch.client.license;

```
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.client.common.ProtocolUtils;
import org.elasticsearch.test.ESTestCase;

import java.io.IOException;
import java.util.Collections;
import java.util.HashMap;
import java.util.Map;

import static org.hamcrest.CoreMatchers.equalTo;

public class StartBasicResponseTests extends ESTestCase {

  public void testFromXContent() throws Exception {
    StartBasicResponse.Status status = randomFrom(StartBasicResponse.Status.values());

    boolean acknowledged = status != StartBasicResponse.Status.NEED_ACKNOWLEDGEMENT;
    String acknowledgeMessage = null;
    Map<String, String[]> ackMessages = Collections.emptyMap();
    if (status != StartBasicResponse.Status.GENERATED_BASIC) {
      acknowledgeMessage = randomAlphaOfLength(10);
      ackMessages = randomAckMessages();
    }

    final StartBasicResponse startBasicResponse = new StartBasicResponse(status, ackMessages,
acknowledgeMessage);

    XContentType xContentType = randomFrom(XContentType.values());
    XContentBuilder builder = XContentFactory.contentBuilder(xContentType);

    toXContent(startBasicResponse, builder);

    final StartBasicResponse response = StartBasicResponse.fromXContent(createParser(builder));
    assertThat(response.isAcknowledged(), equalTo(acknowledged));
    assertThat(response.isBasicStarted(), equalTo(status.isBasicStarted()));
    assertThat(response.getAcknowledgeMessage(), equalTo(acknowledgeMessage));
    assertThat(ProtocolUtils.equals(response.getAcknowledgeMessages(), ackMessages), equalTo(true));
  }

  private static void toXContent(StartBasicResponse response, XContentBuilder builder) throws IOException {
    builder.startObject();
    builder.field("acknowledged", response.isAcknowledged());
    if (response.isBasicStarted()) {
      builder.field("basic_was_started", true);
    } else {
```

```
            builder.field("basic_was_started", false);
            builder.field("error_message", response.getErrorMessage());
        }
        if (response.getAcknowledgeMessages().isEmpty() == false) {
            builder.startObject("acknowledge");
            builder.field("message", response.getAcknowledgeMessage());
            for (Map.Entry<String, String[]> entry : response.getAcknowledgeMessages().entrySet()) {
                builder.startArray(entry.getKey());
                for (String message : entry.getValue()) {
                    builder.value(message);
                }
                builder.endArray();
            }
            builder.endObject();
        }
        builder.endObject();
    }

    private static Map<String, String[]> randomAckMessages() {
        int nFeatures = randomIntBetween(1, 5);

        Map<String, String[]> ackMessages = new HashMap<>();

        for (int i = 0; i < nFeatures; i++) {
            String feature = randomAlphaOfLengthBetween(9, 15);
            int nMessages = randomIntBetween(1, 5);
            String[] messages = new String[nMessages];
            for (int j = 0; j < nMessages; j++) {
                messages[j] = randomAlphaOfLengthBetween(10, 30);
            }
            ackMessages.put(feature, messages);
        }

        return ackMessages;
    }

}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
```

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

import org.elasticsearch.client.Validatable;


public class GetLicenseRequest implements Validatable {

  protected boolean local = false;

  public GetLicenseRequest() {
  }

  public boolean isLocal() {
     return local;
  }

  public void setLocal(boolean local) {
     this.local = local;
  }
}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

package org.elasticsearch.client.license;
```

```
import org.elasticsearch.client.Validatable;
import org.elasticsearch.common.Nullable;

public class StartTrialRequest implements Validatable {

  private final boolean acknowledge;
  private final String licenseType;

  public StartTrialRequest() {
     this(false);
  }

  public StartTrialRequest(boolean acknowledge) {
     this(acknowledge, null);
  }

  public StartTrialRequest(boolean acknowledge, @Nullable String licenseType) {
     this.acknowledge = acknowledge;
     this.licenseType = licenseType;
  }

  public boolean isAcknowledge() {
     return acknowledge;
  }

  public String getLicenseType() {
     return licenseType;
  }
}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
```

```
/**
* Request and Response objects for the default distribution's License
* APIs.
*/
package org.elasticsearch.client.license;
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
package org.elasticsearch.client.license;

import org.elasticsearch.client.TimedRequest;

public class StartBasicRequest extends TimedRequest {
  private final boolean acknowledge;

  public StartBasicRequest() {
      this(false);
  }

  public StartBasicRequest(boolean acknowledge) {
      this.acknowledge = acknowledge;
  }

  public boolean isAcknowledge() {
      return acknowledge;
  }
}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
```

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

package org.elasticsearch.client.license;

import org.elasticsearch.client.TimedRequest;

public class PutLicenseRequest extends TimedRequest {

  private String licenseDefinition;
  private boolean acknowledge = false;

  public PutLicenseRequest() {
  }

  public void setLicenseDefinition(String licenseDefinition) {
      this.licenseDefinition = licenseDefinition;
  }

  public String getLicenseDefinition() {
      return licenseDefinition;
  }

  public void setAcknowledge(boolean acknowledge) {
      this.acknowledge = acknowledge;
  }

  public boolean isAcknowledge() {
      return acknowledge;
  }
}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
package org.elasticsearch.client.license;

import org.elasticsearch.common.ParseField;
import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.xcontent.ConstructingObjectParser;
import org.elasticsearch.common.xcontent.XContentParseException;
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.rest.RestStatus;

import java.io.IOException;
import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Objects;

import static org.elasticsearch.common.xcontent.ConstructingObjectParser.constructorArg;
import static org.elasticsearch.common.xcontent.ConstructingObjectParser.optionalConstructorArg;
import static org.elasticsearch.common.xcontent.XContentParserUtils.ensureExpectedToken;

public class StartBasicResponse {

  private static final ConstructingObjectParser<StartBasicResponse, Void> PARSER = new
ConstructingObjectParser<>(
      "start_basic_response", true, (a, v) -> {
      boolean basicWasStarted = (Boolean) a[0];
      String errorMessage = (String) a[1];

      if (basicWasStarted) {
         return new StartBasicResponse(StartBasicResponse.Status.GENERATED_BASIC);
      }
      StartBasicResponse.Status status = StartBasicResponse.Status.fromErrorMessage(errorMessage);
      @SuppressWarnings("unchecked") Tuple<String, Map<String, String[]>> acknowledgements = (Tuple<String,
Map<String, String[]>>) a[2];
      return new StartBasicResponse(status, acknowledgements.v2(), acknowledgements.v1());
  });
```

```
  static {
    PARSER.declareBoolean(constructorArg(), new ParseField("basic_was_started"));
    PARSER.declareString(optionalConstructorArg(), new ParseField("error_message"));
    PARSER.declareObject(optionalConstructorArg(), (parser, v) -> {
        Map<String, String[]> acknowledgeMessages = new HashMap<>();
        String message = null;
        XContentParser.Token token;
        String currentFieldName = null;
        while ((token = parser.nextToken()) != XContentParser.Token.END_OBJECT) {
          if (token == XContentParser.Token.FIELD_NAME) {
            currentFieldName = parser.currentName();
          } else {
            if (currentFieldName == null) {
              throw new XContentParseException(parser.getTokenLocation(), "expected message header or
acknowledgement");
            }
            if (new ParseField("message").getPreferredName().equals(currentFieldName)) {
              ensureExpectedToken(XContentParser.Token.VALUE_STRING, token,
parser::getTokenLocation);
              message = parser.text();
            } else {
              if (token != XContentParser.Token.START_ARRAY) {
                throw new XContentParseException(parser.getTokenLocation(), "unexpected acknowledgement
type");
              }
              List<String> acknowledgeMessagesList = new ArrayList<>();
              while ((token = parser.nextToken()) != XContentParser.Token.END_ARRAY) {
                ensureExpectedToken(XContentParser.Token.VALUE_STRING, token,
parser::getTokenLocation);
                acknowledgeMessagesList.add(parser.text());
              }
              acknowledgeMessages.put(currentFieldName, acknowledgeMessagesList.toArray(new String[0]));
            }
          }
        }
        return new Tuple<>(message, acknowledgeMessages);
      },
      new ParseField("acknowledge"));
  }

  private Map<String, String[]> acknowledgeMessages;
  private String acknowledgeMessage;

  enum Status {
    GENERATED_BASIC(true, null, RestStatus.OK),
    ALREADY_USING_BASIC(false, "Operation failed: Current license is basic.", RestStatus.FORBIDDEN),
    NEED_ACKNOWLEDGEMENT(false, "Operation failed: Needs acknowledgement.", RestStatus.OK);
```

```
    private final boolean isBasicStarted;
    private final String errorMessage;
    private final RestStatus restStatus;

    Status(boolean isBasicStarted, String errorMessage, RestStatus restStatus) {
        this.isBasicStarted = isBasicStarted;
        this.errorMessage = errorMessage;
        this.restStatus = restStatus;
    }

    String getErrorMessage() {
        return errorMessage;
    }

    boolean isBasicStarted() {
        return isBasicStarted;
    }

    static StartBasicResponse.Status fromErrorMessage(final String errorMessage) {
        final StartBasicResponse.Status[] values = StartBasicResponse.Status.values();
        for (StartBasicResponse.Status status : values) {
            if (Objects.equals(status.errorMessage, errorMessage)) {
                return status;
            }
        }
        throw new IllegalArgumentException("No status for error message ['" + errorMessage + "']");
    }
}

private StartBasicResponse.Status status;

public StartBasicResponse() {
}

StartBasicResponse(StartBasicResponse.Status status) {
    this(status, Collections.emptyMap(), null);
}

StartBasicResponse(StartBasicResponse.Status status,
                   Map<String, String[]> acknowledgeMessages, String acknowledgeMessage) {
    this.status = status;
    this.acknowledgeMessages = acknowledgeMessages;
    this.acknowledgeMessage = acknowledgeMessage;
}

public boolean isAcknowledged() {
    return status != StartBasicResponse.Status.NEED_ACKNOWLEDGEMENT;
```

```
    }

    public boolean isBasicStarted() {
        return status.isBasicStarted;
    }

    public String getErrorMessage() {
        return status.errorMessage;
    }

    public String getAcknowledgeMessage() {
        return acknowledgeMessage;
    }

    public Map<String, String[]> getAcknowledgeMessages() {
        return acknowledgeMessages;
    }

    public static StartBasicResponse fromXContent(XContentParser parser) throws IOException {
        return PARSER.parse(parser, null);
    }

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

package org.elasticsearch.client.license;

import org.elasticsearch.common.ParseField;
import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.xcontent.ConstructingObjectParser;
import org.elasticsearch.common.xcontent.XContentParseException;
```

```
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.client.common.ProtocolUtils;

import java.io.IOException;
import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Objects;

import static org.elasticsearch.common.xcontent.ConstructingObjectParser.constructorArg;
import static org.elasticsearch.common.xcontent.ConstructingObjectParser.optionalConstructorArg;

public final class PutLicenseResponse {

  private static final ConstructingObjectParser<PutLicenseResponse, Void> PARSER = new
ConstructingObjectParser<>(
      "put_license_response", true, (a, v) -> {
      boolean acknowledged = (Boolean) a[0];
      LicensesStatus licensesStatus = LicensesStatus.fromString((String) a[1]);
      @SuppressWarnings("unchecked") Tuple<String, Map<String, String[]>> acknowledgements = (Tuple<String,
Map<String, String[]>>) a[2];
      if (acknowledgements == null) {
         return new PutLicenseResponse(acknowledged, licensesStatus);
      } else {
         return new PutLicenseResponse(acknowledged, licensesStatus, acknowledgements.v1(),
acknowledgements.v2());
      }

  });

  static {
      PARSER.declareBoolean(constructorArg(), new ParseField("acknowledged"));
      PARSER.declareString(constructorArg(), new ParseField("license_status"));
      PARSER.declareObject(optionalConstructorArg(), (parser, v) -> {
            Map<String, String[]> acknowledgeMessages = new HashMap<>();
            String message = null;
            XContentParser.Token token;
            String currentFieldName = null;
            while ((token = parser.nextToken()) != XContentParser.Token.END_OBJECT) {
               if (token == XContentParser.Token.FIELD_NAME) {
                  currentFieldName = parser.currentName();
               } else {
                  if (currentFieldName == null) {
                     throw new XContentParseException(parser.getTokenLocation(), "expected message header or
acknowledgement");
                  }
```

```
                if ("message".equals(currentFieldName)) {
                    if (token != XContentParser.Token.VALUE_STRING) {
                        throw new XContentParseException(parser.getTokenLocation(), "unexpected message header
type");
                    }
                    message = parser.text();
                } else {
                    if (token != XContentParser.Token.START_ARRAY) {
                        throw new XContentParseException(parser.getTokenLocation(), "unexpected acknowledgement
type");
                    }
                    List<String> acknowledgeMessagesList = new ArrayList<>();
                    while ((token = parser.nextToken()) != XContentParser.Token.END_ARRAY) {
                        if (token != XContentParser.Token.VALUE_STRING) {
                            throw new XContentParseException(parser.getTokenLocation(), "unexpected
acknowledgement text");
                        }
                        acknowledgeMessagesList.add(parser.text());
                    }
                    acknowledgeMessages.put(currentFieldName, acknowledgeMessagesList.toArray(new String[0]));
                }
            }
        }
        return new Tuple<>(message, acknowledgeMessages);
    },
    new ParseField("acknowledge"));
}

private boolean acknowledged;
private LicensesStatus status;
private Map<String, String[]> acknowledgeMessages;
private String acknowledgeHeader;

public PutLicenseResponse() {
}

public PutLicenseResponse(boolean acknowledged, LicensesStatus status) {
    this(acknowledged, status, null, Collections.<String, String[]>emptyMap());
}

public PutLicenseResponse(boolean acknowledged, LicensesStatus status, String acknowledgeHeader,
                          Map<String, String[]> acknowledgeMessages) {
    this.acknowledged = acknowledged;
    this.status = status;
    this.acknowledgeHeader = acknowledgeHeader;
    this.acknowledgeMessages = acknowledgeMessages;
}
```

```
    public boolean isAcknowledged() {
        return acknowledged;
    }

    public LicensesStatus status() {
        return status;
    }

    public Map<String, String[]> acknowledgeMessages() {
        return acknowledgeMessages;
    }

    public String acknowledgeHeader() {
        return acknowledgeHeader;
    }

    public static PutLicenseResponse fromXContent(XContentParser parser) throws IOException {
        return PARSER.parse(parser, null);
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;
        if (!super.equals(o)) return false;
        PutLicenseResponse that = (PutLicenseResponse) o;

        return status == that.status &&
            ProtocolUtils.equals(acknowledgeMessages, that.acknowledgeMessages) &&
            Objects.equals(acknowledgeHeader, that.acknowledgeHeader);
    }

    @Override
    public int hashCode() {
        return Objects.hash(super.hashCode(), status, ProtocolUtils.hashCode(acknowledgeMessages),
acknowledgeHeader);
    }

}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
```

```
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
package org.elasticsearch.client.license;

public class GetLicenseResponse {

  private String license;

  GetLicenseResponse() {
  }

  public GetLicenseResponse(String license) {
     this.license = license;
  }

  public String getLicenseDefinition() {
     return license;
  }

}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
package org.elasticsearch.client.license;

/**
```

```
* Status of an X-Pack license.
*/
public enum LicenseStatus {

  ACTIVE("active"),
  INVALID("invalid"),
  EXPIRED("expired");

  private final String label;

  LicenseStatus(String label) {
     this.label = label;
  }

  public String label() {
     return label;
  }

  public static LicenseStatus fromString(String value) {
     switch (value) {
        case "active":
           return ACTIVE;
        case "invalid":
           return INVALID;
        case "expired":
           return EXPIRED;
        default:
           throw new IllegalArgumentException("unknown license status [" + value + "]");
     }
  }
}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
```

```
*/
package org.elasticsearch.client.license;

import org.elasticsearch.client.TimedRequest;

public class DeleteLicenseRequest extends TimedRequest {

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

package org.elasticsearch.client.license;

import org.elasticsearch.common.ParseField;
import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.xcontent.ConstructingObjectParser;
import org.elasticsearch.common.xcontent.XContentParseException;
import org.elasticsearch.common.xcontent.XContentParser;

import java.io.IOException;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import static org.elasticsearch.common.xcontent.ConstructingObjectParser.constructorArg;
import static org.elasticsearch.common.xcontent.ConstructingObjectParser.optionalConstructorArg;

public class StartTrialResponse {

   private static final ConstructingObjectParser<StartTrialResponse, Void> PARSER = new
ConstructingObjectParser<>(
```

```
        "start_trial_response",
        true,
        (Object[] arguments, Void aVoid) -> {
            final boolean acknowledged = (boolean) arguments[0];
            final boolean trialWasStarted = (boolean) arguments[1];
            final String licenseType = (String) arguments[2];
            final String errorMessage = (String) arguments[3];

            @SuppressWarnings("unchecked")
            final Tuple<String, Map<String, String[]>> acknowledgeDetails = (Tuple<String, Map<String, String[]>>)
arguments[4];
            final String acknowledgeHeader;
            final Map<String, String[]> acknowledgeMessages;

            if (acknowledgeDetails != null) {
                acknowledgeHeader = acknowledgeDetails.v1();
                acknowledgeMessages = acknowledgeDetails.v2();
            } else {
                acknowledgeHeader = null;
                acknowledgeMessages = null;
            }

            return new StartTrialResponse(acknowledged, trialWasStarted, licenseType, errorMessage,
acknowledgeHeader,
                acknowledgeMessages);
        }
    );

    static {
        PARSER.declareBoolean(constructorArg(), new ParseField("acknowledged"));
        PARSER.declareBoolean(constructorArg(), new ParseField("trial_was_started"));
        PARSER.declareString(optionalConstructorArg(), new ParseField("type"));
        PARSER.declareString(optionalConstructorArg(), new ParseField("error_message"));
        // todo consolidate this parsing with the parsing in PutLicenseResponse
        PARSER.declareObject(optionalConstructorArg(), (parser, aVoid) -> {
            final Map<String, String[]> acknowledgeMessages = new HashMap<>();
            String message = null;

            final Map<String, Object> parsedMap = parser.map();
            for (Map.Entry<String, Object> entry : parsedMap.entrySet()) {
                if (entry.getKey().equals("message")) {
                    if (entry.getValue() instanceof String) {
                        message = (String) entry.getValue();
                    } else {
                        throw new XContentParseException(parser.getTokenLocation(), "unexpected acknowledgement
header type");
                    }
                } else {
```

```
                if (entry.getValue() instanceof List) {
                    final List<String> messageStrings = new ArrayList<>();
                    @SuppressWarnings("unchecked")
                    final List<Object> messageObjects = (List<Object>) entry.getValue();
                    for (Object messageObject : messageObjects) {
                        if (messageObject instanceof String) {
                            messageStrings.add((String) messageObject);
                        } else {
                            throw new XContentParseException(parser.getTokenLocation(), "expected text in
acknowledgement message");
                        }
                    }

                    acknowledgeMessages.put(entry.getKey(), messageStrings.toArray(new
String[messageStrings.size()]));
                } else {
                    throw new XContentParseException(parser.getTokenLocation(), "unexpected acknowledgement
message type");
                }
            }
        }

        if (message == null) {
            throw new XContentParseException(parser.getTokenLocation(), "expected acknowledgement header");
        }

        return new Tuple<>(message, acknowledgeMessages);

    }, new ParseField("acknowledge"));
}

public static StartTrialResponse fromXContent(XContentParser parser) throws IOException {
    return PARSER.apply(parser, null);
}

private final boolean acknowledged;
private final boolean trialWasStarted;
private final String licenseType;
private final String errorMessage;
private final String acknowledgeHeader;
private final Map<String, String[]> acknowledgeMessages;

public StartTrialResponse(boolean acknowledged,
                          boolean trialWasStarted,
                          String licenseType,
                          String errorMessage,
                          String acknowledgeHeader,
                          Map<String, String[]> acknowledgeMessages) {
```

```
        this.acknowledged = acknowledged;
        this.trialWasStarted = trialWasStarted;
        this.licenseType = licenseType;
        this.errorMessage = errorMessage;
        this.acknowledgeHeader = acknowledgeHeader;
        this.acknowledgeMessages = acknowledgeMessages;
    }

    /**
     * Returns true if the request that corresponds to this response acknowledged license changes that would occur as a result of starting
     * a trial license
     */
    public boolean isAcknowledged() {
        return acknowledged;
    }

    /**
     * Returns true if a trial license was started as a result of the request corresponding to this response. Returns false if the cluster
     * did not start a trial, or a trial had already been started before the corresponding request was made
     */
    public boolean isTrialWasStarted() {
        return trialWasStarted;
    }

    /**
     * If a trial license was started as a result of the request corresponding to this response (see {@link #isTrialWasStarted()}) then
     * returns the type of license that was started on the cluster. Returns null otherwise
     */
    public String getLicenseType() {
        return licenseType;
    }

    /**
     * If a trial license was not started as a result of the request corresponding to this response (see {@link #isTrialWasStarted()} then
     * returns a brief message explaining why the trial could not be started. Returns false otherwise
     */
    public String getErrorMessage() {
        return errorMessage;
    }

    /**
     * If the request corresponding to this response did not acknowledge licensing changes that would result from starting a trial license
```

```
    * (see {@link #isAcknowledged()}), returns a message describing how the user must acknowledge licensing
changes as a result of
    * such a request. Returns null otherwise
    */
   public String getAcknowledgeHeader() {
       return acknowledgeHeader;
   }

   /**
    * If the request corresponding to this response did not acknowledge licensing changes that would result from
starting a trial license
    * (see {@link #isAcknowledged()}, returns a map. The map's keys are names of commercial Elasticsearch
features, and their values are
    * messages about how those features will be affected by licensing changes as a result of starting a trial license
    */
   public Map<String, String[]> getAcknowledgeMessages() {
       return acknowledgeMessages;
   }
}
/*
* Licensed to Elasticsearch under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
package org.elasticsearch.client.license;

import java.util.Locale;

public enum LicensesStatus {
   VALID((byte) 0),
   INVALID((byte) 1),
   EXPIRED((byte) 2);

   private final byte id;
```

```
  LicensesStatus(byte id) {
    this.id = id;
  }

  public int id() {
    return id;
  }

  public static LicensesStatus fromId(int id) {
    if (id == 0) {
      return VALID;
    } else if (id == 1) {
      return INVALID;
    } else if (id == 2) {
      return EXPIRED;
    } else {
      throw new IllegalStateException("no valid LicensesStatus for id=" + id);
    }
  }


  @Override
  public String toString() {
    return this.name().toLowerCase(Locale.ROOT);
  }

  public static LicensesStatus fromString(String value) {
    switch (value) {
      case "valid":
        return VALID;
      case "invalid":
        return INVALID;
      case "expired":
        return EXPIRED;
      default:
        throw new IllegalArgumentException("unknown licenses status [" + value + "]");
    }
  }
}
```

Apache HttpComponents AsyncClient
Copyright 2010-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache Commons Logging
Copyright 2003-2013 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (http://www.apache.org/).

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

7. Disclaimer of Warranty. Unless required by applicable law or

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

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

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

END OF TERMS AND CONDITIONS

This project contains annotations derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls.
See http://www.jcip.net and the Creative Commons Attribution License
(http://creativecommons.org/licenses/by/2.5)

Apache HttpCore NIO
Copyright 2005-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache HttpComponents Core
Copyright 2005-2016 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (http://www.apache.org/).
Elasticsearch X-Pack
Copyright 2009-2017 Elasticsearch

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license.licensor.tools;

import org.elasticsearch.cli.Command;
import org.elasticsearch.cli.CommandTestCase;
import org.elasticsearch.cli.ExitCodes;
import org.elasticsearch.cli.UserException;
import org.elasticsearch.common.bytes.BytesArray;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.license.License;
import org.elasticsearch.license.licensor.TestUtils;
import org.junit.Before;

import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Path;

public class LicenseGenerationToolTests extends CommandTestCase {

  protected Path pubKeyPath = null;
  protected Path priKeyPath = null;

  @Before
  public void setup() throws Exception {
    pubKeyPath = getDataPath(TestUtils.PUBLIC_KEY_RESOURCE);
    priKeyPath = getDataPath(TestUtils.PRIVATE_KEY_RESOURCE);
  }

  @Override
  protected Command newCommand() {
    return new LicenseGeneratorTool();
  }

  public void testMissingKeyPaths() throws Exception {
    Path pub = createTempDir().resolve("pub");
    Path pri = createTempDir().resolve("pri");
    UserException e = expectThrows(
        UserException.class,
        () -> execute(
            "--publicKeyPath",
            pub.toString(),
```

```
                "--privateKeyPath",
                pri.toString()));
    assertTrue(e.getMessage(), e.getMessage().contains("pri does not exist"));
    assertEquals(ExitCodes.USAGE, e.exitCode);

    Files.createFile(pri);
    e = expectThrows(
            UserException.class,
            () -> execute(
                "--publicKeyPath",
                pub.toString(),
                "--privateKeyPath",
                pri.toString()));
    assertTrue(e.getMessage(), e.getMessage().contains("pub does not exist"));
    assertEquals(ExitCodes.USAGE, e.exitCode);
}

public void testMissingLicenseSpec() throws Exception {
    UserException e = expectThrows(
            UserException.class,
            () -> execute(
                "--publicKeyPath",
                pubKeyPath.toString(),
                "--privateKeyPath",
                priKeyPath.toString()));
    assertTrue(
            e.getMessage(),
            e.getMessage().contains("Must specify either --license or --licenseFile"));
    assertEquals(ExitCodes.USAGE, e.exitCode);
}

public void testLicenseSpecString() throws Exception {
    TestUtils.LicenseSpec inputLicenseSpec =
            TestUtils.generateRandomLicenseSpec(License.VERSION_CURRENT);
    String licenseSpecString = TestUtils.generateLicenseSpecString(inputLicenseSpec);
    String output = execute(
            "--publicKeyPath",
            pubKeyPath.toString(),
            "--privateKeyPath",
            priKeyPath.toString(),
            "--license",
            licenseSpecString);
    final BytesArray bytes = new BytesArray(output.getBytes(StandardCharsets.UTF_8));
    License outputLicense = License.fromSource(bytes, XContentType.JSON);
    TestUtils.assertLicenseSpec(inputLicenseSpec, outputLicense);
}

public void testLicenseSpecFile() throws Exception {
```

```
        TestUtils.LicenseSpec inputLicenseSpec =
                TestUtils.generateRandomLicenseSpec(License.VERSION_CURRENT);
        String licenseSpecString = TestUtils.generateLicenseSpecString(inputLicenseSpec);
        Path licenseSpecFile = createTempFile();
        Files.write(licenseSpecFile, licenseSpecString.getBytes(StandardCharsets.UTF_8));
        String output = execute(
                "--publicKeyPath",
                pubKeyPath.toString(),
                "--privateKeyPath",
                priKeyPath.toString(),
                "--licenseFile",
                licenseSpecFile.toString());
        final BytesArray bytes = new BytesArray(output.getBytes(StandardCharsets.UTF_8));
        License outputLicense = License.fromSource(bytes, XContentType.JSON);
        TestUtils.assertLicenseSpec(inputLicenseSpec, outputLicense);
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license.licensor.tools;

import org.elasticsearch.cli.Command;
import org.elasticsearch.cli.CommandTestCase;
import org.elasticsearch.cli.ExitCodes;
import org.elasticsearch.cli.UserException;

import java.nio.file.Files;
import java.nio.file.Path;

import static org.hamcrest.CoreMatchers.containsString;

public class KeyPairGenerationToolTests extends CommandTestCase {

    @Override
    protected Command newCommand() {
        return new KeyPairGeneratorTool();
    }

    public void testMissingKeyPaths() throws Exception {
        Path exists = createTempFile("", "existing");
        Path dne = createTempDir().resolve("dne");
        UserException e = expectThrows(
                UserException.class,
                () -> execute(
```

```
                "--publicKeyPath",
                exists.toString(),
                "--privateKeyPath",
                dne.toString()));
        assertThat(e.getMessage(), containsString("existing"));
        assertEquals(ExitCodes.USAGE, e.exitCode);
        e = expectThrows(
            UserException.class,
            () -> execute(
                "--publicKeyPath",
                dne.toString(),
                "--privateKeyPath",
                exists.toString()));
        assertThat(e.getMessage(), containsString("existing"));
        assertEquals(ExitCodes.USAGE, e.exitCode);
    }

    public void testTool() throws Exception {
        Path keysDir = createTempDir();
        Path publicKeyFilePath = keysDir.resolve("public");
        Path privateKeyFilePath = keysDir.resolve("private");

        execute(
            "--publicKeyPath",
            publicKeyFilePath.toString(),
            "--privateKeyPath",
            privateKeyFilePath.toString());
        assertTrue(publicKeyFilePath.toString(), Files.exists(publicKeyFilePath));
        assertTrue(privateKeyFilePath.toString(), Files.exists(privateKeyFilePath));
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license.licensor.tools;

import org.elasticsearch.cli.Command;
import org.elasticsearch.cli.CommandTestCase;
import org.elasticsearch.cli.ExitCodes;
import org.elasticsearch.cli.UserException;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.license.License;
import org.elasticsearch.license.licensor.TestUtils;
import org.junit.Before;
```

```
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Path;

public class LicenseVerificationToolTests extends CommandTestCase {
  protected Path pubKeyPath = null;
  protected Path priKeyPath = null;

  @Before
  public void setup() throws Exception {
    logger.error("project.basedir [{}]", System.getProperty("project.basedir"));
    pubKeyPath = getDataPath(TestUtils.PUBLIC_KEY_RESOURCE);
    priKeyPath = getDataPath(TestUtils.PRIVATE_KEY_RESOURCE);
  }

  @Override
  protected Command newCommand() {
    return new LicenseVerificationTool();
  }

  public void testMissingKeyPath() throws Exception {
    Path pub = createTempDir().resolve("pub");
    UserException e = expectThrows(
        UserException.class,
        () -> execute("--publicKeyPath", pub.toString()));
    assertTrue(e.getMessage(), e.getMessage().contains("pub does not exist"));
    assertEquals(ExitCodes.USAGE, e.exitCode);
  }

  public void testMissingLicenseSpec() throws Exception {
    UserException e = expectThrows(UserException.class, () -> {
      execute("--publicKeyPath", pubKeyPath.toString());
    });
    assertTrue(
        e.getMessage(),
        e.getMessage().contains("Must specify either --license or --licenseFile"));
    assertEquals(ExitCodes.USAGE, e.exitCode);
  }

  public void testBrokenLicense() throws Exception {
    final TimeValue oneHour = TimeValue.timeValueHours(1);
    License signedLicense = TestUtils.generateSignedLicense(oneHour, pubKeyPath, priKeyPath);
    License tamperedLicense = License.builder()
      .fromLicenseSpec(signedLicense, signedLicense.signature())
      .expiryDate(signedLicense.expiryDate() + randomIntBetween(1, 1000)).build();
    UserException e = expectThrows(
        UserException.class,
        () -> execute(
```

```
                "--publicKeyPath",
                pubKeyPath.toString(),
                "--license",
                TestUtils.dumpLicense(tamperedLicense)));
        assertEquals("Invalid License!", e.getMessage());
        assertEquals(ExitCodes.DATA_ERROR, e.exitCode);
    }

    public void testLicenseSpecString() throws Exception {
        final TimeValue oneHour = TimeValue.timeValueHours(1);
        License signedLicense = TestUtils.generateSignedLicense(oneHour, pubKeyPath, priKeyPath);
        String output = execute(
                "--publicKeyPath",
                pubKeyPath.toString(),
                "--license",
                TestUtils.dumpLicense(signedLicense));
        assertFalse(output, output.isEmpty());
    }

    public void testLicenseSpecFile() throws Exception {
        final TimeValue oneHour = TimeValue.timeValueHours(1);
        License signedLicense = TestUtils.generateSignedLicense(oneHour, pubKeyPath, priKeyPath);
        Path licenseSpecFile = createTempFile();
        Files.write(
                licenseSpecFile,
                TestUtils.dumpLicense(signedLicense).getBytes(StandardCharsets.UTF_8));
        String output = execute(
                "--publicKeyPath",
                pubKeyPath.toString(),
                "--licenseFile",
                licenseSpecFile.toString());
        assertFalse(output, output.isEmpty());
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license.licensor;

import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.license.DateUtils;
import org.elasticsearch.license.License;
import org.elasticsearch.license.LicenseVerifier;
import org.elasticsearch.test.ESTestCase;
import org.junit.After;
```

```
import org.junit.Before;

import java.nio.file.Files;
import java.nio.file.Path;

public class LicenseVerificationTests extends ESTestCase {

  protected Path pubKeyPath = null;
  protected Path priKeyPath = null;

  @Before
  public void setup() throws Exception {
    pubKeyPath = getDataPath("/public.key");
    priKeyPath = getDataPath("/private.key");
  }

  @After
  public void cleanUp() {
    pubKeyPath = null;
    priKeyPath = null;
  }

  public void testGeneratedLicenses() throws Exception {
    final TimeValue fortyEightHours = TimeValue.timeValueHours(2 * 24);
    final License license =
        TestUtils.generateSignedLicense(fortyEightHours, pubKeyPath, priKeyPath);
    assertTrue(LicenseVerifier.verifyLicense(license, Files.readAllBytes(pubKeyPath)));
  }

  public void testLicenseTampering() throws Exception {
    final TimeValue twoHours = TimeValue.timeValueHours(2);
    License license = TestUtils.generateSignedLicense(twoHours, pubKeyPath, priKeyPath);

    final License tamperedLicense = License.builder()
        .fromLicenseSpec(license, license.signature())
        .expiryDate(license.expiryDate() + 10 * 24 * 60 * 60 * 1000L)
        .validate()
        .build();

    assertFalse(LicenseVerifier.verifyLicense(tamperedLicense, Files.readAllBytes(pubKeyPath)));
  }

  public void testRandomLicenseVerification() throws Exception {
    TestUtils.LicenseSpec licenseSpec = TestUtils.generateRandomLicenseSpec(
        randomIntBetween(License.VERSION_START, License.VERSION_CURRENT));
    License generatedLicense = generateSignedLicense(licenseSpec, pubKeyPath, priKeyPath);
    assertTrue(LicenseVerifier.verifyLicense(generatedLicense, Files.readAllBytes(pubKeyPath)));
  }
```

```
  private static License generateSignedLicense(
        TestUtils.LicenseSpec spec, Path pubKeyPath, Path priKeyPath) throws Exception {
    LicenseSigner signer = new LicenseSigner(priKeyPath, pubKeyPath);
    License.Builder builder = License.builder()
            .uid(spec.uid)
            .feature(spec.feature)
            .type(spec.type)
            .subscriptionType(spec.subscriptionType)
            .issuedTo(spec.issuedTo)
            .issuer(spec.issuer)
            .maxNodes(spec.maxNodes);

    if (spec.expiryDate != null) {
        builder.expiryDate(DateUtils.endOfTheDay(spec.expiryDate));
    } else {
        builder.expiryDate(spec.expiryDateInMillis);
    }
    if (spec.issueDate != null) {
        builder.issueDate(DateUtils.beginningOfTheDay(spec.issueDate));
    } else {
        builder.issueDate(spec.issueDateInMillis);
    }
    builder.version(spec.version);
    return signer.sign(builder.build());
  }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license.licensor;

import org.elasticsearch.common.Strings;
import org.elasticsearch.common.joda.FormatDateTimeFormatter;
import org.elasticsearch.common.joda.Joda;
import org.elasticsearch.common.time.DateMathParser;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.license.DateUtils;
import org.elasticsearch.license.License;
import org.elasticsearch.test.ESTestCase;
import org.hamcrest.MatcherAssert;
```

```
import org.joda.time.format.DateTimeFormatter;

import java.io.IOException;
import java.nio.file.Path;
import java.util.UUID;

import static com.carrotsearch.randomizedtesting.RandomizedTest.randomBoolean;
import static com.carrotsearch.randomizedtesting.RandomizedTest.randomInt;
import static com.carrotsearch.randomizedtesting.RandomizedTest.randomIntBetween;
import static org.elasticsearch.common.xcontent.XContentFactory.jsonBuilder;
import static org.elasticsearch.test.ESTestCase.randomFrom;
import static org.hamcrest.core.IsEqual.equalTo;

public class TestUtils {

  public static final String PUBLIC_KEY_RESOURCE = "/public.key";
  public static final String PRIVATE_KEY_RESOURCE = "/private.key";

  private static final FormatDateTimeFormatter formatDateTimeFormatter =
        Joda.forPattern("yyyy-MM-dd");
  private static final DateMathParser dateMathParser = formatDateTimeFormatter.toDateMathParser();
  private static final DateTimeFormatter dateTimeFormatter = formatDateTimeFormatter.printer();

  public static String dumpLicense(License license) throws Exception {
    XContentBuilder builder = XContentFactory.contentBuilder(XContentType.JSON);
    builder.startObject();
    builder.startObject("license");
    license.toInnerXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.endObject();
    builder.endObject();
    return Strings.toString(builder);
  }

  public static String dateMathString(String time, final long now) {
    return dateTimeFormatter.print(dateMathParser.parse(time, () -> now));
  }

  public static long dateMath(String time, final long now) {
    return dateMathParser.parse(time, () -> now);
  }

  public static LicenseSpec generateRandomLicenseSpec(int version) {
    boolean datesInMillis = randomBoolean();
    long now = System.currentTimeMillis();
    String uid = UUID.randomUUID().toString();
    String issuer = "issuer__"  + randomInt();
    String issuedTo = "issuedTo__" + randomInt();
    String type = version < License.VERSION_NO_FEATURE_TYPE ?
```

```
            randomFrom("subscription", "internal", "development") :
            randomFrom("basic", "silver", "dev", "gold", "platinum");
    final String subscriptionType;
    final String feature;
    if (version < License.VERSION_NO_FEATURE_TYPE) {
        subscriptionType = randomFrom("gold", "silver", "platinum");
        feature = "feature__" + randomInt();
    } else {
        subscriptionType = null;
        feature = null;
    }
    int maxNodes = randomIntBetween(5, 100);
    if (datesInMillis) {
        long issueDateInMillis = dateMath("now", now);
        long expiryDateInMillis = dateMath("now+10d/d", now);
        return new LicenseSpec(
                version,
                uid,
                feature,
                issueDateInMillis,
                expiryDateInMillis,
                type,
                subscriptionType,
                issuedTo,
                issuer,
                maxNodes);
    } else {
        String issueDate = dateMathString("now", now);
        String expiryDate = dateMathString("now+10d/d", now);
        return new LicenseSpec(
                version,
                uid,
                feature,
                issueDate,
                expiryDate, type,
                subscriptionType,
                issuedTo,
                issuer,
                maxNodes);
    }
}

public static String generateLicenseSpecString(LicenseSpec licenseSpec) throws IOException {
    XContentBuilder licenses = jsonBuilder();
    licenses.startObject();
    licenses.startObject("license")
            .field("uid", licenseSpec.uid)
            .field("type", licenseSpec.type)
```

```
            .field("subscription_type", licenseSpec.subscriptionType)
            .field("issued_to", licenseSpec.issuedTo)
            .field("issuer", licenseSpec.issuer)
            .field("feature", licenseSpec.feature)
            .field("max_nodes", licenseSpec.maxNodes);

    if (licenseSpec.issueDate != null) {
        licenses.field("issue_date", licenseSpec.issueDate);
    } else {
        licenses.field("issue_date_in_millis", licenseSpec.issueDateInMillis);
    }
    if (licenseSpec.expiryDate != null) {
        licenses.field("expiry_date", licenseSpec.expiryDate);
    } else {
        licenses.field("expiry_date_in_millis", licenseSpec.expiryDateInMillis);
    }
    licenses.field("version", licenseSpec.version);
    licenses.endObject();
    licenses.endObject();
    return Strings.toString(licenses);
}

public static void assertLicenseSpec(LicenseSpec spec, License license) {
    MatcherAssert.assertThat(license.uid(), equalTo(spec.uid));
    MatcherAssert.assertThat(license.issuedTo(), equalTo(spec.issuedTo));
    MatcherAssert.assertThat(license.issuer(), equalTo(spec.issuer));
    MatcherAssert.assertThat(license.type(), equalTo(spec.type));
    MatcherAssert.assertThat(license.maxNodes(), equalTo(spec.maxNodes));
    if (spec.issueDate != null) {
        MatcherAssert.assertThat(
                license.issueDate(),
                equalTo(DateUtils.beginningOfTheDay(spec.issueDate)));
    } else {
        MatcherAssert.assertThat(license.issueDate(), equalTo(spec.issueDateInMillis));
    }
    if (spec.expiryDate != null) {
        MatcherAssert.assertThat(
                license.expiryDate(),
                equalTo(DateUtils.endOfTheDay(spec.expiryDate)));
    } else {
        MatcherAssert.assertThat(license.expiryDate(), equalTo(spec.expiryDateInMillis));
    }
}

public static License generateSignedLicense(
        TimeValue expiryDuration, Path pubKeyPath, Path priKeyPath) throws Exception {
    long issue = System.currentTimeMillis();
    int version = ESTestCase.randomIntBetween(License.VERSION_START, License.VERSION_CURRENT);
```

```
    String type = version < License.VERSION_NO_FEATURE_TYPE ?
            randomFrom("subscription", "internal", "development") :
            randomFrom("trial", "basic", "silver", "dev", "gold", "platinum");
    final License.Builder builder = License.builder()
            .uid(UUID.randomUUID().toString())
            .expiryDate(issue + expiryDuration.getMillis())
            .issueDate(issue)
            .version(version)
            .type(type)
            .issuedTo("customer")
            .issuer("elasticsearch")
            .maxNodes(5);
    if (version == License.VERSION_START) {
        builder.subscriptionType(randomFrom("dev", "gold", "platinum", "silver"));
        builder.feature(ESTestCase.randomAlphaOfLength(10));
    }
    LicenseSigner signer = new LicenseSigner(priKeyPath, pubKeyPath);
    return signer.sign(builder.build());
}

public static class LicenseSpec {
    public final int version;
    public final String feature;
    public final String issueDate;
    public final long issueDateInMillis;
    public final String expiryDate;
    public final long expiryDateInMillis;
    public final String uid;
    public final String type;
    public final String subscriptionType;
    public final String issuedTo;
    public final String issuer;
    public final int maxNodes;

    public LicenseSpec(
            int version,
            String uid,
            String feature,
            long issueDateInMillis,
            long expiryDateInMillis,
            String type,
            String subscriptionType,
            String issuedTo,
            String issuer,
            int maxNodes) {
        this.version = version;
        this.feature = feature;
        this.issueDateInMillis = issueDateInMillis;
```

```
        this.issueDate = null;
        this.expiryDateInMillis = expiryDateInMillis;
        this.expiryDate = null;
        this.uid = uid;
        this.type = type;
        this.subscriptionType = subscriptionType;
        this.issuedTo = issuedTo;
        this.issuer = issuer;
        this.maxNodes = maxNodes;
    }

    public LicenseSpec(
            int version,
            String uid,
            String feature,
            String issueDate,
            String expiryDate,
            String type,
            String subscriptionType,
            String issuedTo,
            String issuer,
            int maxNodes) {
        this.version = version;
        this.feature = feature;
        this.issueDate = issueDate;
        this.issueDateInMillis = -1;
        this.expiryDate = expiryDate;
        this.expiryDateInMillis = -1;
        this.uid = uid;
        this.type = type;
        this.subscriptionType = subscriptionType;
        this.issuedTo = issuedTo;
        this.issuer = issuer;
        this.maxNodes = maxNodes;
    }
  }

}
```


0"0
 *H
0
+5Bj:h$Tfsr&avDgpi]2oR<Cizzz E(yhZ}TCjh~!P|499cq5b!Y{VX=,Vd ;g9hD`Xv8n|i) 34aYfsWBkuOo@
u@175/YmI4~
d&^w t<q^XQ^KG#|X?tV}f;5ENHgXyk9Mh/WM*D;D^]H`"
}W"o"'YY@9x"|`hNb>a3>8ZOSd}R4HxQU99dQ\nTxL?>*}QMB7fhIE952^hn=-m2-
pF8T_I^yT1=)dc=t*Ge^^f71cb~_^|5yZl&9LSdxR8XNUC1%^R;CiVIN~O}c%\'h|&UX;j{_#9/^/1H@_Wr$l,})v[#?
;m;`O'ZiB <Pu
OkH4jbr

WvF\w'#`)~Vx/|?{zDa
\| hGEi?#r~N
zm`nL|~1~'?/'>k?~gyk,b' _.41K??3d2kgb$7~QjlqN]j,Lr-9Vl34uNe+'4dpP.V}uvq4D}08LrTeg5Z+n@&+(P=-M"
[rw8baSjJaam+ +?b!u&2tW!IYi.fj@/ G#uZv_sgj2UXk.I`dbq7kT%h5'Nv5[D4!l@1
5}*`"DK[%N^(Af9l] 0X`M:c01uAm\jB/.;f
```
es.logger.level=INFO
log4j.rootLogger=${es.logger.level}, out

log4j.logger.org.apache.http=INFO, out
log4j.additivity.org.apache.http=false

log4j.logger.org.elasticsearch.license=TRACE

log4j.appender.out=org.apache.log4j.ConsoleAppender
log4j.appender.out.layout=org.apache.log4j.PatternLayout
log4j.appender.out.layout.conversionPattern=[%d{ISO8601}][%-5p][%-25c] %m%n
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license.licensor;

import org.apache.lucene.util.BytesRef;
import org.apache.lucene.util.BytesRefIterator;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.hash.MessageDigests;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.license.CryptUtils;
import org.elasticsearch.license.License;

import java.io.IOException;
import java.nio.ByteBuffer;
import java.nio.file.Files;
import java.nio.file.Path;
import java.security.InvalidKeyException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.security.PrivateKey;
import java.security.PublicKey;
import java.security.SecureRandom;
import java.security.Signature;
import java.security.SignatureException;
import java.util.Base64;
import java.util.Collections;
```

```
import java.util.Map;

/**
* Responsible for generating a license signature according to the signature spec and sign it with
* the provided encrypted private key
*/
public class LicenseSigner {

  private static final int MAGIC_LENGTH = 13;
  private final Path publicKeyPath;
  private final Path privateKeyPath;

  public LicenseSigner(final Path privateKeyPath, final Path publicKeyPath) {
     this.publicKeyPath = publicKeyPath;
     this.privateKeyPath = privateKeyPath;
  }

  /**
   * Generates a signature for the {@code licenseSpec}. Signature structure:
   * <code>
   * | VERSION | MAGIC | PUB_KEY_DIGEST | SIGNED_LICENSE_CONTENT |
   * </code>
   *
   * @return a signed License
   */
  public License sign(License licenseSpec) throws IOException {
     XContentBuilder contentBuilder = XContentFactory.contentBuilder(XContentType.JSON);
     final Map<String, String> licenseSpecViewMode =
           Collections.singletonMap(License.LICENSE_SPEC_VIEW_MODE, "true");
     licenseSpec.toXContent(contentBuilder, new ToXContent.MapParams(licenseSpecViewMode));
     final byte[] signedContent;
     final boolean preV4 = licenseSpec.version() < License.VERSION_CRYPTO_ALGORITHMS;
     try {
        final Signature rsa = Signature.getInstance("SHA512withRSA");
        PrivateKey decryptedPrivateKey =
CryptUtils.readEncryptedPrivateKey(Files.readAllBytes(privateKeyPath));
        rsa.initSign(decryptedPrivateKey);
        final BytesRefIterator iterator = BytesReference.bytes(contentBuilder).iterator();
        BytesRef ref;
        while((ref = iterator.next()) != null) {
           rsa.update(ref.bytes, ref.offset, ref.length);
        }
        signedContent = rsa.sign();
     } catch (InvalidKeyException
           | IOException
           | NoSuchAlgorithmException
           | SignatureException e) {
        throw new IllegalStateException(e);
```

```
        }
        final byte[] magic = new byte[MAGIC_LENGTH];
        SecureRandom random = new SecureRandom();
        random.nextBytes(magic);
        final byte[] publicKeyBytes = Files.readAllBytes(publicKeyPath);
        PublicKey publicKey = CryptUtils.readPublicKey(publicKeyBytes);
        final byte[] pubKeyFingerprint = preV4 ?
Base64.getEncoder().encode(CryptUtils.writeEncryptedPublicKey(publicKey)) :
                getPublicKeyFingerprint(publicKeyBytes);
        byte[] bytes = new byte[4 + 4 + MAGIC_LENGTH + 4 + pubKeyFingerprint.length + 4 +
signedContent.length];
        ByteBuffer byteBuffer = ByteBuffer.wrap(bytes);
        byteBuffer.putInt(licenseSpec.version())
                .putInt(magic.length)
                .put(magic)
                .putInt(pubKeyFingerprint.length)
                .put(pubKeyFingerprint)
                .putInt(signedContent.length)
                .put(signedContent);

        return License.builder()
                .fromLicenseSpec(licenseSpec, Base64.getEncoder().encodeToString(bytes))
                .build();
    }

    private byte[] getPublicKeyFingerprint(byte[] keyBytes) {
        MessageDigest sha256 = MessageDigests.sha256();
        sha256.update(keyBytes);
        return sha256.digest();
    }
}
apply plugin: 'elasticsearch.build'

dependencies {
 compile "org.elasticsearch.plugin:x-pack-core:${version}"
 compile "org.elasticsearch:elasticsearch:${version}"
 testCompile "org.elasticsearch.test:framework:${version}"
}

project.forbiddenPatterns {
 exclude '**/*.key'
}

dependencyLicenses.enabled = false

task buildZip(type: Zip, dependsOn: jar) {
 String parentDir = "license-tools-${version}"
 into(parentDir + '/lib') {
```

```
  from jar
  from configurations.runtime
 }
 into(parentDir + '/bin') {
  from 'bin'
 }
}

assemble.dependsOn buildZip
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.license.LicenseService;
import org.elasticsearch.license.XPackLicenseState;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.ssl.SSLService;
import org.elasticsearch.xpack.deprecation.Deprecation;
import org.elasticsearch.xpack.graph.Graph;
import org.elasticsearch.xpack.logstash.Logstash;
import org.elasticsearch.xpack.ml.MachineLearning;
import org.elasticsearch.xpack.monitoring.Monitoring;
import org.elasticsearch.xpack.security.Security;
import org.elasticsearch.xpack.watcher.Watcher;

import java.nio.file.Path;

public class CompositeTestingXPackPlugin extends LocalStateCompositeXPackPlugin {

  public CompositeTestingXPackPlugin(final Settings settings, final Path configPath) throws Exception {
    super(settings, configPath);
    CompositeTestingXPackPlugin thisVar = this;
    plugins.add(new Deprecation());
    plugins.add(new Graph(settings));
    plugins.add(new Logstash(settings));
    plugins.add(new MachineLearning(settings, configPath) {
      @Override
      protected XPackLicenseState getLicenseState() {
        return super.getLicenseState();
      }
    });
    plugins.add(new Monitoring(settings) {
      @Override
      protected SSLService getSslService() {
```

```
            return thisVar.getSslService();
        }

        @Override
        protected LicenseService getLicenseService() {
            return thisVar.getLicenseService();
        }

        @Override
        protected XPackLicenseState getLicenseState() {
            return thisVar.getLicenseState();
        }
    });
    plugins.add(new Watcher(settings) {
        @Override
        protected SSLService getSslService() {
            return thisVar.getSslService();
        }

        @Override
        protected XPackLicenseState getLicenseState() {
            return thisVar.getLicenseState();
        }
    });
    plugins.add(new Security(settings, configPath) {
        @Override
        protected SSLService getSslService() { return thisVar.getSslService(); }

        @Override
        protected XPackLicenseState getLicenseState() { return thisVar.getLicenseState(); }
    });
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.monitoring;

import org.elasticsearch.action.admin.cluster.node.info.NodeInfo;
import org.elasticsearch.action.admin.cluster.node.info.NodesInfoResponse;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.env.Environment;
import org.elasticsearch.node.MockNode;
import org.elasticsearch.node.Node;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.plugins.PluginInfo;
```

```
import org.elasticsearch.test.ESIntegTestCase.ClusterScope;
import org.elasticsearch.test.discovery.TestZenDiscovery;
import org.elasticsearch.tribe.TribePlugin;
import org.elasticsearch.xpack.core.XPackSettings;
import org.elasticsearch.xpack.monitoring.test.MonitoringIntegTestCase;

import java.nio.file.Path;
import java.util.ArrayList;
import java.util.Collection;
import java.util.function.Function;

import static org.elasticsearch.test.ESIntegTestCase.Scope.TEST;
import static org.hamcrest.Matchers.equalTo;

@ClusterScope(scope = TEST, transportClientRatio = 0, numClientNodes = 0, numDataNodes = 0)
public class MonitoringPluginTests extends MonitoringIntegTestCase {

  public MonitoringPluginTests() throws Exception {
    super();
  }

  @Override
  protected void startMonitoringService() {
    // do nothing as monitoring is sometime unbound
  }

  @Override
  protected void stopMonitoringService() {
    // do nothing as monitoring is sometime unbound
  }

  @Override
  protected boolean addTestZenDiscovery() {
    return false;
  }

  public static class TribeAwareTestZenDiscoveryPlugin extends TestZenDiscovery.TestPlugin {

    public TribeAwareTestZenDiscoveryPlugin(Settings settings) {
      super(settings);
    }

    @Override
    public Settings additionalSettings() {
      if (settings.getGroups("tribe", true).isEmpty()) {
        return super.additionalSettings();
      } else {
        return Settings.EMPTY;
```

```
        }
    }
}

public static class MockTribePlugin extends TribePlugin {

    public MockTribePlugin(Settings settings) {
        super(settings);
    }

    protected Function<Settings, Node> nodeBuilder(Path configPath) {
        return settings -> new MockNode(new Environment(settings, configPath), internalCluster().getPlugins());
    }

}

@Override
protected Collection<Class<? extends Plugin>> nodePlugins() {
    ArrayList<Class<? extends Plugin>> plugins = new ArrayList<>(super.nodePlugins());
    plugins.add(MockTribePlugin.class);
    plugins.add(TribeAwareTestZenDiscoveryPlugin.class);
    return plugins;
}

@Override
protected Settings nodeSettings(int nodeOrdinal) {
    return Settings.builder()
            .put(super.nodeSettings(nodeOrdinal))
            .put(MonitoringService.INTERVAL.getKey(), "-1")
            .put(XPackSettings.SECURITY_ENABLED.getKey(), false)
            .put(XPackSettings.WATCHER_ENABLED.getKey(), false)
            .build();
}

@Override
protected Settings transportClientSettings() {
    return Settings.builder()
            .put(super.transportClientSettings())
            .put(XPackSettings.SECURITY_ENABLED.getKey(), false)
            .build();
}

public void testMonitoringEnabled() {
    internalCluster().startNode(Settings.builder()
            .put(XPackSettings.MONITORING_ENABLED.getKey(), true)
            .build());
    assertPluginIsLoaded();
    assertServiceIsBound(MonitoringService.class);
```

```
}

public void testMonitoringDisabled() {
    internalCluster().startNode(Settings.builder()
            .put(XPackSettings.MONITORING_ENABLED.getKey(), false)
            .build());
    assertPluginIsLoaded();
    assertServiceIsNotBound(MonitoringService.class);
}

public void testMonitoringEnabledOnTribeNode() {
    internalCluster().startNode(Settings.builder()
            .put(XPackSettings.MONITORING_ENABLED.getKey(), true)
            .put("tribe.name", "t1")
            .build());
    assertPluginIsLoaded();
    assertServiceIsBound(MonitoringService.class);
}

public void testMonitoringDisabledOnTribeNode() {
    internalCluster().startNode(Settings.builder().put("tribe.name", "t1").build());
    assertPluginIsLoaded();
    assertServiceIsNotBound(MonitoringService.class);
}

private void assertPluginIsLoaded() {
    NodesInfoResponse response = client().admin().cluster().prepareNodesInfo().setPlugins(true).get();
    for (NodeInfo nodeInfo : response.getNodes()) {
        assertNotNull(nodeInfo.getPlugins());

        boolean found = false;
        for (PluginInfo plugin : nodeInfo.getPlugins().getPluginInfos()) {
            assertNotNull(plugin);

            if (LocalStateMonitoring.class.getName().equals(plugin.getName())) {
                found = true;
                break;
            }
        }
        assertThat("xpack plugin not found", found, equalTo(true));
    }
}

private void assertServiceIsBound(Class<?> klass) {
    try {
        Object binding = internalCluster().getDataNodeInstance(klass);
        assertNotNull(binding);
        assertTrue(klass.isInstance(binding));
```

```
        } catch (Exception e) {
            fail("no service bound for class " + klass.getSimpleName());
        }
    }

    private void assertServiceIsNotBound(Class<?> klass) {
        try {
            // it should be bound, but directly as null
            assertNull(internalCluster().getDataNodeInstance(klass));
        } catch (Exception ce) {
            assertThat("message contains error about missing implementation: " + ce.getMessage(),
                    ce.getMessage().contains("Could not find a suitable constructor"), equalTo(true));
        }
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.monitoring;

import org.elasticsearch.client.Client;
import org.elasticsearch.license.TribeTransportTestCase;
import org.elasticsearch.xpack.core.XPackField;
import org.elasticsearch.xpack.core.monitoring.action.MonitoringBulkAction;
import org.elasticsearch.xpack.core.monitoring.action.MonitoringBulkRequest;

import java.util.Collections;
import java.util.List;

public class MonitoringTribeTests extends TribeTransportTestCase {

    @Override
    protected List<String> enabledFeatures() {
        return Collections.singletonList(XPackField.MONITORING);
    }

    @Override
    protected void verifyActionOnClientNode(Client client) throws Exception {
        assertMonitoringTransportActionsWorks(client);
    }

    @Override
    protected void verifyActionOnMasterNode(Client masterClient) throws Exception {
        assertMonitoringTransportActionsWorks(masterClient);
    }
```

```
    @Override
    protected void verifyActionOnDataNode(Client dataNodeClient) throws Exception {
        assertMonitoringTransportActionsWorks(dataNodeClient);
    }

    private static void assertMonitoringTransportActionsWorks(Client client) throws Exception {
        client.execute(MonitoringBulkAction.INSTANCE, new MonitoringBulkRequest());
    }

    @Override
    protected void verifyActionOnTribeNode(Client tribeClient) {
        failAction(tribeClient, MonitoringBulkAction.INSTANCE);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.test;

import org.elasticsearch.Build;
import org.elasticsearch.common.bytes.BytesArray;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.transport.TransportAddress;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.license.GetLicenseResponse;
import org.elasticsearch.license.License;
import org.elasticsearch.license.LicensingClient;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.protocol.xpack.license.LicensesStatus;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackPlugin;
import org.elasticsearch.xpack.core.XPackSettings;
import org.junit.AfterClass;

import java.io.IOException;
import java.net.InetAddress;
import java.net.InetSocketAddress;
import java.net.URL;
import java.nio.charset.StandardCharsets;
import java.util.Collection;
import java.util.Collections;

import static org.hamcrest.CoreMatchers.equalTo;

public class LicensingTribeIT extends ESIntegTestCase {
```

```
private static TestCluster cluster2;
private static TestCluster tribeNode;

@Override
protected Collection<Class<? extends Plugin>> nodePlugins() {
    return Collections.singletonList(XPackPlugin.class);
}

@Override
protected Collection<Class<? extends Plugin>> transportClientPlugins() {
    return Collections.singletonList(LocalStateCompositeXPackPlugin.class);
}

@Override
public void setUp() throws Exception {
    super.setUp();
    if (cluster2 == null) {
        cluster2 = buildExternalCluster(System.getProperty("tests.cluster2"));
    }
    if (tribeNode == null) {
        tribeNode = buildExternalCluster(System.getProperty("tests.tribe"));
    }
}

@Override
protected Settings nodeSettings(int nodeOrdinal) {
    Settings.Builder builder = Settings.builder();
    builder.put(super.nodeSettings(nodeOrdinal));
    return builder.build();
}

@AfterClass
public static void tearDownExternalClusters() throws IOException {
    if (cluster2 != null) {
        try {
            cluster2.close();
        } finally {
            cluster2 = null;
        }
    }
    if (tribeNode != null) {
        try {
            tribeNode.close();
        } finally {
            tribeNode = null;
        }
    }
}
```

```
  @Override
  protected Settings externalClusterClientSettings() {
    Settings.Builder builder = Settings.builder();
    builder.put(XPackSettings.SECURITY_ENABLED.getKey(), false);
    builder.put(XPackSettings.MONITORING_ENABLED.getKey(), false);
    builder.put(XPackSettings.WATCHER_ENABLED.getKey(), false);
    builder.put(XPackSettings.GRAPH_ENABLED.getKey(), false);
    builder.put(XPackSettings.MACHINE_LEARNING_ENABLED.getKey(), false);
    return builder.build();
  }

  private ExternalTestCluster buildExternalCluster(String clusterAddresses) throws IOException {
    String[] stringAddresses = clusterAddresses.split(",");
    TransportAddress[] transportAddresses = new TransportAddress[stringAddresses.length];
    int i = 0;
    for (String stringAddress : stringAddresses) {
      URL url = new URL("http://" + stringAddress);
      InetAddress inetAddress = InetAddress.getByName(url.getHost());
      transportAddresses[i++] = new TransportAddress(new InetSocketAddress(inetAddress, url.getPort()));
    }
    return new ExternalTestCluster(createTempDir(), externalClusterClientSettings(), transportClientPlugins(),
transportAddresses);
  }

  public void testLicensePropagateToTribeNode() throws Exception {
    assumeTrue("License is only valid when tested against snapshot/test keys", Build.CURRENT.isSnapshot());
    // test that auto-generated basic license propagates to tribe
    assertBusy(() -> {
      GetLicenseResponse getLicenseResponse = new
LicensingClient(tribeNode.client()).prepareGetLicense().get();
      assertNotNull(getLicenseResponse.license());
      assertThat(getLicenseResponse.license().operationMode(), equalTo(License.OperationMode.BASIC));
    });

    // test that signed license with higher operation mode takes precedence
    LicensingClient cluster2Client = new LicensingClient(cluster2.client());
    PutLicenseResponse licenseResponse = cluster2Client
        .preparePutLicense(License.fromSource(new
BytesArray(PLATINUM_LICENSE.getBytes(StandardCharsets.UTF_8)), XContentType.JSON))
        .setAcknowledge(true).get();
    assertThat(licenseResponse.isAcknowledged(), equalTo(true));
    assertThat(licenseResponse.status(), equalTo(LicensesStatus.VALID));
    assertBusy(() -> {
      GetLicenseResponse getLicenseResponse = new
LicensingClient(tribeNode.client()).prepareGetLicense().get();
      assertNotNull(getLicenseResponse.license());
      assertThat(getLicenseResponse.license().operationMode(), equalTo(License.OperationMode.PLATINUM));
```

```
        });

        // test removing signed license falls back works
        assertTrue(cluster2Client.prepareDeleteLicense().get().isAcknowledged());
        assertBusy(() -> {
            GetLicenseResponse getLicenseResponse = new
LicensingClient(tribeNode.client()).prepareGetLicense().get();
            assertNotNull(getLicenseResponse.license());
            assertThat(getLicenseResponse.license().operationMode(), equalTo(License.OperationMode.BASIC));
        });
    }

    public void testDummy() throws Exception {
        // this test is here so that testLicensePropagateToTribeNode's assumption
        // doesn't result in this test suite to have no tests run and trigger a build failure
    }

    private static final String PLATINUM_LICENSE = "{\"license\":{\"uid\":\"1\",\"type\":\"platinum\"," +
            "\"issue_date_in_millis\":1411948800000,\"expiry_date_in_millis\":1914278399999,\"max_nodes\":1," +
            "\"issued_to\":\"issuedTo\",\"issuer\":\"issuer\"," +
"\"signature\":\"AAAAAwAAAA2hWlkvKcxQIpdVWdCtAAABmC9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxO
WpSbTVoMVZwUzRxVk1" +
"PSmkxakxZdW5IMlhlTHNoN1N2MXMvRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHBkWkprZWZ
SQi9iNmRQNkw1SlpLN0l" +
"DalZCS095MXRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0U0MXZ0dDJHSERiZTVLOHNzSDBy
WnpoZEphZHBEZjUrTVB" +
"xRENNSXNsWWJjZllaODdzVmEzUjNiWktNWGM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOEhQaVB2azhmU
k9QVzhFeTFoM1Q0RnJXSG5" +
"3MWk2K055c28zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2VE1qbVNQS2lxOHN5bUErZlNIWkZSVmZI
WEtaSU9wTTJENDVvT1NCYkla" +
"cUYyK2FwRW9xa0t6dldMbmMzSGtQc3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRWUzh6UG5pdEN
GWFRreXNKNkE9PQAAAQBWg" +
"u3yZp0KOBG//92X4YVmau3P5asvx0FAPDX2Ze734Tap/nc30X6Rt4yEEm+6bCQr/ibBOqWboJKRbbTZLBQfY
FmL1ZqvAY3bJJ1/Xs" +
"8NyDfxKGztlUt/IIOzHPzxs0f8Bv4OJeK48vjovWaDc1Vmo4n1SGyyL0JcEbOWC6A3U3mBsWn7wLUe+hW9+
akVAYOO5TIcm60ub7k" +
"H/LIZNOhvGglSVDbl3p8EBkNMy0CV7urQ0wdG1nLCnvf8/BiT15lC5nLrM9Dt5w3pzciPlASzw4iksW/CzvYy5
tjOoWKEnxi2EZOB" +
            "9dKyT4mTdvyBOrTHLdgr4lmHd3qYAEgcTCaQ\",\"start_date_in_millis\":-1}}";
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.client.Client;
```

```
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.protocol.xpack.license.GetLicenseRequest;

import static org.elasticsearch.license.TestUtils.generateSignedLicense;

public class LicenseTribeTests extends TribeTransportTestCase {

  @Override
  protected void verifyActionOnClientNode(Client client) throws Exception {
    assertLicenseTransportActionsWorks(client);
  }

  @Override
  protected void verifyActionOnMasterNode(Client masterClient) throws Exception {
    assertLicenseTransportActionsWorks(masterClient);
  }

  @Override
  protected void verifyActionOnDataNode(Client dataNodeClient) throws Exception {
    assertLicenseTransportActionsWorks(dataNodeClient);
  }

  private static void assertLicenseTransportActionsWorks(Client client) throws Exception {
    client.execute(GetLicenseAction.INSTANCE, new GetLicenseRequest()).get();
    client.execute(PutLicenseAction.INSTANCE, new PutLicenseRequest()
        .license(generateSignedLicense(TimeValue.timeValueHours(1))));
    client.execute(DeleteLicenseAction.INSTANCE, new DeleteLicenseRequest());
  }

  @Override
  protected void verifyActionOnTribeNode(Client tribeClient) throws Exception {
    // The get licence action should work, but everything else should fail
    tribeClient.execute(GetLicenseAction.INSTANCE, new GetLicenseRequest()).get();
    failAction(tribeClient, PutLicenseAction.INSTANCE);
    failAction(tribeClient, DeleteLicenseAction.INSTANCE);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.action.admin.cluster.health.ClusterHealthResponse;
import org.elasticsearch.action.admin.cluster.node.info.NodeInfo;
```

```
import org.elasticsearch.action.admin.cluster.node.info.NodesInfoResponse;
import org.elasticsearch.analysis.common.CommonAnalysisPlugin;
import org.elasticsearch.client.Client;
import org.elasticsearch.client.Requests;
import org.elasticsearch.cluster.health.ClusterHealthStatus;
import org.elasticsearch.cluster.node.DiscoveryNode;
import org.elasticsearch.cluster.node.DiscoveryNodes;
import org.elasticsearch.common.Priority;
import org.elasticsearch.common.UUIDs;
import org.elasticsearch.common.network.NetworkModule;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.transport.TransportAddress;
import org.elasticsearch.discovery.zen.SettingsBasedHostsProvider;
import org.elasticsearch.env.Environment;
import org.elasticsearch.node.MockNode;
import org.elasticsearch.node.Node;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.test.ESIntegTestCase;
import org.elasticsearch.test.ESIntegTestCase.ClusterScope;
import org.elasticsearch.test.ESIntegTestCase.Scope;
import org.elasticsearch.test.InternalTestCluster;
import org.elasticsearch.test.NodeConfigurationSource;
import org.elasticsearch.test.TestCluster;
import org.elasticsearch.test.discovery.TestZenDiscovery;
import org.elasticsearch.tribe.TribePlugin;
import org.elasticsearch.xpack.CompositeTestingXPackPlugin;
import org.elasticsearch.xpack.core.XPackClientPlugin;
import org.elasticsearch.xpack.core.XPackPlugin;
import org.elasticsearch.xpack.core.XPackSettings;
import org.elasticsearch.xpack.core.XPackField;
import org.elasticsearch.xpack.deprecation.Deprecation;
import org.elasticsearch.xpack.graph.Graph;
import org.elasticsearch.xpack.logstash.Logstash;
import org.elasticsearch.xpack.ml.MachineLearning;
import org.elasticsearch.xpack.core.ml.MachineLearningField;
import org.elasticsearch.xpack.monitoring.Monitoring;
import org.elasticsearch.xpack.security.Security;
import org.elasticsearch.xpack.upgrade.Upgrade;
import org.elasticsearch.xpack.watcher.Watcher;

import java.nio.file.Path;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collection;
import java.util.Collections;
import java.util.List;
import java.util.function.Function;
```

```
import static org.elasticsearch.test.hamcrest.ElasticsearchAssertions.assertAcked;
import static org.hamcrest.Matchers.anyOf;
import static org.hamcrest.Matchers.containsString;
import static org.hamcrest.Matchers.equalTo;

@ClusterScope(scope = Scope.TEST, transportClientRatio = 0, numClientNodes = 1, numDataNodes = 2)
public abstract class TribeTransportTestCase extends ESIntegTestCase {

  protected List<String> enabledFeatures() {
    return Collections.emptyList();
  }

  @Override
  protected final Settings nodeSettings(int nodeOrdinal) {
    final Settings.Builder builder = Settings.builder()
        .put(NetworkModule.HTTP_ENABLED.getKey(), false)
        .put("transport.type", getTestTransportType());
    List<String> enabledFeatures = enabledFeatures();
    builder.put(XPackSettings.SECURITY_ENABLED.getKey(),
enabledFeatures.contains(XPackField.SECURITY));
    builder.put(XPackSettings.MONITORING_ENABLED.getKey(),
enabledFeatures.contains(XPackField.MONITORING));
    builder.put(XPackSettings.WATCHER_ENABLED.getKey(),
enabledFeatures.contains(XPackField.WATCHER));
    builder.put(XPackSettings.GRAPH_ENABLED.getKey(), enabledFeatures.contains(XPackField.GRAPH));
    builder.put(XPackSettings.MACHINE_LEARNING_ENABLED.getKey(),
enabledFeatures.contains(XPackField.MACHINE_LEARNING));
    builder.put(MachineLearningField.AUTODETECT_PROCESS.getKey(), false);
    return builder.build();
  }

  @Override
  protected boolean ignoreExternalCluster() {
    return true;
  }

  @Override
  protected boolean addTestZenDiscovery() {
    return false;
  }

  public static class TribeAwareTestZenDiscoveryPlugin extends TestZenDiscovery.TestPlugin {

    public TribeAwareTestZenDiscoveryPlugin(Settings settings) {
      super(settings);
    }

    @Override
```

```
    public Settings additionalSettings() {
        if (settings.getGroups("tribe", true).isEmpty()) {
            return super.additionalSettings();
        } else {
            return Settings.EMPTY;
        }
    }
}

public static class MockTribePlugin extends TribePlugin {

    public MockTribePlugin(Settings settings) {
        super(settings);
    }

    protected Function<Settings, Node> nodeBuilder(Path configPath) {
        return settings -> new MockNode(new Environment(settings, configPath), internalCluster().getPlugins());
    }

}

@Override
protected Collection<Class<? extends Plugin>> nodePlugins() {
    ArrayList<Class<? extends Plugin>> plugins = new ArrayList<>();
    plugins.add(MockTribePlugin.class);
    plugins.add(TribeAwareTestZenDiscoveryPlugin.class);
    plugins.add(CompositeTestingXPackPlugin.class);
    plugins.add(CommonAnalysisPlugin.class);
    return plugins;
}

@Override
protected final Collection<Class<? extends Plugin>> transportClientPlugins() {
    ArrayList<Class<? extends Plugin>> plugins = new ArrayList<>();
    plugins.add(MockTribePlugin.class);
    plugins.add(TribeAwareTestZenDiscoveryPlugin.class);
    plugins.add(XPackClientPlugin.class);
    plugins.add(CommonAnalysisPlugin.class);
    return plugins;
}
public void testTribeSetup() throws Exception {
    NodeConfigurationSource nodeConfigurationSource = new NodeConfigurationSource() {
        @Override
        public Settings nodeSettings(int nodeOrdinal) {
            return TribeTransportTestCase.this.nodeSettings(nodeOrdinal);
        }

        @Override
```

```
            public Path nodeConfigPath(int nodeOrdinal) {
                return null;
            }

            @Override
            public Collection<Class<? extends Plugin>> nodePlugins() {
                return TribeTransportTestCase.this.nodePlugins();
            }

            @Override
            public Settings transportClientSettings() {
                return TribeTransportTestCase.this.transportClientSettings();
            }

            @Override
            public Collection<Class<? extends Plugin>> transportClientPlugins() {
                return TribeTransportTestCase.this.transportClientPlugins();
            }
        };
        final InternalTestCluster cluster2 = new InternalTestCluster(
                randomLong(), createTempDir(), true, true, 2, 2,
                UUIDs.randomBase64UUID(random()), nodeConfigurationSource, 1, false, "tribe_node2",
                getMockPlugins(), getClientWrapper());

        cluster2.beforeTest(random(), 0.0);

        logger.info("create 2 indices, test1 on t1, and test2 on t2");
        assertAcked(internalCluster().client().admin().indices().prepareCreate("test1").get());
        assertAcked(cluster2.client().admin().indices().prepareCreate("test2").get());
        ensureYellow(internalCluster());
        ensureYellow(cluster2);
//      Map<String,String> asMap = internalCluster().getDefaultSettings().getAsMap();
        Settings.Builder tribe1Defaults = Settings.builder();
        Settings.Builder tribe2Defaults = Settings.builder();
        internalCluster().getDefaultSettings().keySet().forEach(k -> {
            if (k.startsWith("path.") == false) {
                tribe1Defaults.copy(k, internalCluster().getDefaultSettings());
                tribe2Defaults.copy(k, internalCluster().getDefaultSettings());
            }
        });
        tribe1Defaults.normalizePrefix("tribe.t1.");
        tribe2Defaults.normalizePrefix("tribe.t2.");
        // give each tribe it's unicast hosts to connect to
        tribe1Defaults.putList("tribe.t1." +
SettingsBasedHostsProvider.DISCOVERY_ZEN_PING_UNICAST_HOSTS_SETTING.getKey(),
                getUnicastHosts(internalCluster().client()));
        tribe1Defaults.putList("tribe.t2." +
SettingsBasedHostsProvider.DISCOVERY_ZEN_PING_UNICAST_HOSTS_SETTING.getKey(),
```

```
            getUnicastHosts(cluster2.client()));

        Settings merged = Settings.builder()
            .put("tribe.t1.cluster.name", internalCluster().getClusterName())
            .put("tribe.t2.cluster.name", cluster2.getClusterName())
            .put("tribe.t1.transport.tcp.port", 0)
            .put("tribe.t2.transport.tcp.port", 0)
            .put("tribe.t1.transport.type", getTestTransportType())
            .put("tribe.t2.transport.type", getTestTransportType())
            .put("tribe.blocks.write", false)
            .put(tribe1Defaults.build())
            .put(tribe2Defaults.build())
            .put(NetworkModule.HTTP_ENABLED.getKey(), false)
            .put(internalCluster().getDefaultSettings())
            .put(XPackSettings.SECURITY_ENABLED.getKey(), false) // otherwise it conflicts with mock transport
            .put(XPackSettings.MACHINE_LEARNING_ENABLED.getKey(), false)
            .put(MachineLearningField.AUTODETECT_PROCESS.getKey(), false)
            .put("tribe.t1." + XPackSettings.SECURITY_ENABLED.getKey(), false)
            .put("tribe.t2." + XPackSettings.SECURITY_ENABLED.getKey(), false)
            .put("tribe.t1." + XPackSettings.WATCHER_ENABLED.getKey(), false)
            .put("tribe.t2." + XPackSettings.WATCHER_ENABLED.getKey(), false)
            .put("tribe.t1." + XPackSettings.MACHINE_LEARNING_ENABLED.getKey(), false)
            .put("tribe.t2." + XPackSettings.MACHINE_LEARNING_ENABLED.getKey(), false)
            .put("tribe.t1." + MachineLearningField.AUTODETECT_PROCESS.getKey(), false)
            .put("tribe.t2." + MachineLearningField.AUTODETECT_PROCESS.getKey(), false)
            .put("node.name", "tribe_node") // make sure we can identify threads from this node
            .put("transport.type", getTestTransportType())
            .build();

        final List<Class<? extends Plugin>> mockPlugins = Arrays.asList(MockTribePlugin.class,
TribeAwareTestZenDiscoveryPlugin.class,
            getTestTransportPlugin(), Deprecation.class, Graph.class, Logstash.class, MachineLearning.class,
Monitoring.class,
            Security.class, Upgrade.class, Watcher.class, XPackPlugin.class);
        final Node tribeNode = new MockNode(merged, mockPlugins).start();
        Client tribeClient = tribeNode.client();

        logger.info("wait till tribe has the same nodes as the 2 clusters");
        assertBusy(() -> {
            DiscoveryNodes tribeNodes = tribeNode.client().admin().cluster().prepareState().get().getState().getNodes();
            assertThat(countDataNodesForTribe("t1", tribeNodes),
                equalTo(internalCluster().client().admin().cluster().prepareState().get().getState()
                    .getNodes().getDataNodes().size()));
            assertThat(countDataNodesForTribe("t2", tribeNodes),
                equalTo(cluster2.client().admin().cluster().prepareState().get().getState()
                    .getNodes().getDataNodes().size()));
        });
        logger.info(" --> verify transport actions for tribe node");
```

```
    verifyActionOnTribeNode(tribeClient);
    logger.info(" --> verify transport actions for data node");
    verifyActionOnDataNode((randomBoolean() ? internalCluster() : cluster2).dataNodeClient());
    logger.info(" --> verify transport actions for master node");
    verifyActionOnMasterNode((randomBoolean() ? internalCluster() : cluster2).masterClient());
    logger.info(" --> verify transport actions for client node");
    verifyActionOnClientNode((randomBoolean() ? internalCluster() : cluster2).coordOnlyNodeClient());
    try {
        cluster2.wipe(Collections.<String>emptySet());
    } finally {
        cluster2.afterTest();
    }
    tribeNode.close();
    cluster2.close();
}

/**
 * Verify transport action behaviour on client node
 */
protected abstract void verifyActionOnClientNode(Client client) throws Exception;

/**
 * Verify transport action behaviour on master node
 */
protected abstract void verifyActionOnMasterNode(Client masterClient) throws Exception;

/**
 * Verify transport action behaviour on data node
 */
protected abstract void verifyActionOnDataNode(Client dataNodeClient) throws Exception;

/**
 * Verify transport action behaviour on tribe node
 */
protected abstract void verifyActionOnTribeNode(Client tribeClient) throws Exception;

protected void failAction(Client client, Action action) {
    try {
        client.execute(action, action.newRequestBuilder(client).request());
        fail("expected [" + action.name() + "] to fail");
    } catch (IllegalStateException e) {
        assertThat(e.getMessage(), containsString("failed to find action"));
    }
}

private void ensureYellow(TestCluster testCluster) {
    ClusterHealthResponse actionGet = testCluster.client().admin().cluster()
                .health(Requests.clusterHealthRequest().waitForYellowStatus()
```

```
                        .waitForEvents(Priority.LANGUID).waitForNoRelocatingShards(true)).actionGet();
        if (actionGet.isTimedOut()) {
            logger.info("ensureGreen timed out, cluster state:\n{}\n{}", testCluster.client().admin().cluster()
                    .prepareState().get().getState(),
                    testCluster.client().admin().cluster().preparePendingClusterTasks().get());
            assertThat("timed out waiting for yellow state", actionGet.isTimedOut(), equalTo(false));
        }
        assertThat(actionGet.getStatus(), anyOf(equalTo(ClusterHealthStatus.YELLOW),
equalTo(ClusterHealthStatus.GREEN)));
    }

    private int countDataNodesForTribe(String tribeName, DiscoveryNodes nodes) {
        int count = 0;
        for (DiscoveryNode node : nodes) {
            if (!node.isDataNode()) {
                continue;
            }
            if (tribeName.equals(node.getAttributes().get("tribe.name"))) {
                count++;
            }
        }
        return count;
    }

    private static String[] getUnicastHosts(Client client) {
        ArrayList<String> unicastHosts = new ArrayList<>();
        NodesInfoResponse nodeInfos = client.admin().cluster().prepareNodesInfo().clear().setTransport(true).get();
        for (NodeInfo info : nodeInfos.getNodes()) {
            TransportAddress address = info.getTransport().getAddress().publishAddress();
            unicastHosts.add(address.getAddress() + ":" + address.getPort());
        }
        return unicastHosts.toArray(new String[unicastHosts.size()]);
    }

}
import org.elasticsearch.gradle.test.ClusterConfiguration
import org.elasticsearch.gradle.test.ClusterFormationTasks
import org.elasticsearch.gradle.test.NodeInfo

apply plugin: 'elasticsearch.standalone-test'
apply plugin: 'elasticsearch.standalone-rest-test'
apply plugin: 'elasticsearch.rest-test'

dependencies {
    testCompile project(path: ':modules:tribe', configuration: 'runtime')
    testCompile project(path: xpackProject('plugin').path, configuration: 'testArtifacts')

    // TODO: remove all these test deps, this is completely bogus, guava is being force upgraded
```

```
  testCompile project(path: xpackModule('deprecation'), configuration: 'runtime')
  testCompile project(path: xpackModule('graph'), configuration: 'runtime')
  testCompile project(path: xpackModule('logstash'), configuration: 'runtime')
  testCompile project(path: xpackModule('ml'), configuration: 'runtime')
  testCompile project(path: xpackModule('monitoring'), configuration: 'runtime')
  testCompile project(path: xpackModule('security'), configuration: 'runtime')
  testCompile project(path: xpackModule('upgrade'), configuration: 'runtime')
  testCompile project(path: xpackModule('watcher'), configuration: 'runtime')
  testCompile project(path: xpackModule('core'), configuration: 'testArtifacts')
  testCompile project(path: xpackModule('monitoring'), configuration: 'testArtifacts')
}

compileTestJava.options.compilerArgs << "-Xlint:-rawtypes,-unchecked"
namingConventions.skipIntegTestInDisguise = true

test {
 /*
  * We have to disable setting the number of available processors as tests in the same JVM randomize processors and
will step on each
  * other if we allow them to set the number of available processors as it's set-once in Netty.
  */
 systemProperty 'es.set.netty.runtime.available.processors', 'false'
 include '**/*Tests.class'
}

String licensePath = xpackProject('license-tools').projectDir.toPath().resolve('src/test/resources').toString()
sourceSets {
 test {
  resources {
   srcDirs += [licensePath]
  }
 }
}

project.forbiddenPatterns {
 exclude '**/*.key'
}

task setupClusterOne {}
ClusterConfiguration cluster1Config = new ClusterConfiguration(project)
cluster1Config.clusterName = 'cluster1'
cluster1Config.setting('node.name', 'cluster1-node1')
// x-pack
cluster1Config.setting('xpack.monitoring.enabled', false)
cluster1Config.setting('xpack.security.enabled', false)
cluster1Config.setting('xpack.watcher.enabled', false)
cluster1Config.setting('xpack.graph.enabled', false)
cluster1Config.setting('xpack.ml.enabled', false)
```

```
List<NodeInfo> cluster1Nodes = ClusterFormationTasks.setup(project, 'clusterOne', setupClusterOne,
cluster1Config)

task setupClusterTwo {}
ClusterConfiguration cluster2Config = new ClusterConfiguration(project)
cluster2Config.clusterName = 'cluster2'
cluster2Config.setting('node.name', 'cluster2-node1')
// x-pack
cluster2Config.setting('xpack.monitoring.enabled', false)
cluster2Config.setting('xpack.monitoring.enabled', false)
cluster2Config.setting('xpack.security.enabled', false)
cluster2Config.setting('xpack.watcher.enabled', false)
cluster2Config.setting('xpack.graph.enabled', false)
cluster2Config.setting('xpack.ml.enabled', false)

List<NodeInfo> cluster2Nodes = ClusterFormationTasks.setup(project, 'clusterTwo', setupClusterTwo,
cluster2Config)

integTestCluster {
 dependsOn setupClusterOne, setupClusterTwo
 setting 'node.name', 'tribe-node'
 setting 'tribe.on_conflict', 'prefer_cluster1'
 setting 'tribe.cluster1.cluster.name', 'cluster1'
 setting 'tribe.cluster1.discovery.zen.ping.unicast.hosts', '"${-> cluster1Nodes.get(0).transportUri()}"'
 setting 'tribe.cluster1.http.enabled', 'true'
 setting 'tribe.cluster1.http.port', '0'
 setting 'tribe.cluster1.transport.tcp.port', '0'
 setting 'tribe.cluster1.xpack.monitoring.enabled', false
 setting 'tribe.cluster1.xpack.monitoring.enabled', false
 setting 'tribe.cluster1.xpack.security.enabled', false
 setting 'tribe.cluster1.xpack.watcher.enabled', false
 setting 'tribe.cluster1.xpack.graph.enabled', false
 setting 'tribe.cluster1.xpack.ml.enabled', false
 setting 'tribe.cluster1.xpack.license.self_generated.type', 'trial'
 setting 'tribe.cluster2.cluster.name', 'cluster2'
 setting 'tribe.cluster2.discovery.zen.ping.unicast.hosts', '"${-> cluster2Nodes.get(0).transportUri()}"'
 setting 'tribe.cluster2.http.enabled', 'true'
 setting 'tribe.cluster1.http.port', '0'
 setting 'tribe.cluster1.transport.tcp.port', '0'
 setting 'tribe.cluster2.xpack.monitoring.enabled', false
 setting 'tribe.cluster2.xpack.monitoring.enabled', false
 setting 'tribe.cluster2.xpack.security.enabled', false
 setting 'tribe.cluster2.xpack.watcher.enabled', false
 setting 'tribe.cluster2.xpack.graph.enabled', false
 setting 'tribe.cluster2.xpack.ml.enabled', false
 setting 'tribe.cluster2.xpack.license.self_generated.type', 'trial'
 // x-pack
```

```
 setting 'xpack.monitoring.enabled', false
 setting 'xpack.monitoring.enabled', false
 setting 'xpack.security.enabled', false
 setting 'xpack.watcher.enabled', false
 setting 'xpack.graph.enabled', false
 setting 'xpack.ml.enabled', false
 setting 'xpack.license.self_generated.type', 'trial'
 waitCondition = { node, ant ->
   File tmpFile = new File(node.cwd, 'wait.success')
   // 5 nodes: tribe + clusterOne (1 node + tribe internal node) + clusterTwo (1 node + tribe internal node)
   ant.get(src: "http://${node.httpUri()}/_cluster/health?wait_for_nodes=>=5&wait_for_status=yellow",
           dest: tmpFile.toString(),
           ignoreerrors: true,
           retries: 10)
   return tmpFile.exists()
 }
}

integTestRunner {
 /*
  * We have to disable setting the number of available processors as tests in the same JVM randomize processors and
will step on each
  * other if we allow them to set the number of available processors as it's set-once in Netty.
  */
 systemProperty 'es.set.netty.runtime.available.processors', 'false'
 systemProperty 'tests.cluster', "${-> cluster1Nodes.get(0).transportUri()}"
 systemProperty 'tests.cluster2', "${-> cluster2Nodes.get(0).transportUri()}"
 systemProperty 'tests.tribe', "${-> integTest.nodes.get(0).transportUri()}"
 finalizedBy 'clusterOne#stop'
 finalizedBy 'clusterTwo#stop'
}
---
teardown:
 - do:
     xpack.license.post:
       acknowledge: true
       body: |
         {"licenses":[{"uid":"3aa62ffe-36e1-4fad-bfdc-
9dff8301eb22","type":"trial","issue_date_in_millis":1523456691721,"expiry_date_in_millis":1838816691721,"max
_nodes":5,"issued_to":"customer","issuer":"elasticsearch","signature":"AAAABAAAAA2kWNcuc+DT0lrlmYZKA
AAAIAo5/x6hrsGh1GqqrJmy4qgmEC7gK0U4zQ6q5ZEMhm4jAAABAEn6fG9y2VxKBu2T3D5hffh56kzOQOD
COdhr0y2d17ZSIJMZRqO7ZywPCWNS1aR33GhfIHkTER0ysML0xMH/gXavhyRvMBndJj0UBKzuwpTawSlnx
YtcqN8mSBIvJC7Ki+uJ1SpAILC2ZP9fnkRlqwXqBlTwfYn7xnZgu9DKrOWru/ipTPObo7jcePl8VTK6nWFen7/h
CFDQTUFZ0jQvd+nq7A1PAcHGNxGfdbMVmAXCXgGWkRfT3clo9/vadgo+isNyh1sPq9mN7gwsvBAKtA1Frp
H2EXYYbfOsSpBvUmhYMgErLg1k3/CbS0pCWLKOaX1xTMayosdZOjagU3auZXY=","start_date_in_millis":-
1}]}
---
"Installing and getting license works":
```

```
## current license version
- do:
    xpack.license.post:
      acknowledge: true
      body: |
        {"licenses":[{"uid":"894371dc-9t49-4997-93cb-
8o2e3r7fa6a8","type":"trial","issue_date_in_millis":1411948800000,"expiry_date_in_millis":1916956799999,"max
_nodes":1,"issued_to":"issuedTo","issuer":"issuer","signature":"AAAAAgAAAA0FWh0T9njItjQ2qammAAABmC
9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVoMVZwUzRxVk1PSmkxakxZdW5IMlhlTHNoN1N2MXM
vRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHBkWkprZWZSQi9iNmRQNkw1SlpLN0lDalZCS095M
XRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0U0MXZ0dDJHSERiZTVLOHNzSDByWnpoZEphZH
BEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNiWktNWGM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOEhQa
VB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K055c28zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2VE1q
bVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9wTTJENDVvT1NCYklacUYyK2FwRW9xa0t6dldMbmMzSGtQ
c3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRWUzh6UG5pdENGWFRreXNKNkE9PQAAAQBZhvozA0
trrxhUZ1QbaTsKTna9C5KVQ6pv8yg1pnsBpZXCl8kX1SrgoFn1bXq61IvJwfw5qnmYNiH3hRhTO9EyaCBqaLk8
NXZQ6TrRkQSpEnnBwAYUkZeKXsIuBoOk4B4mzwC/r8aMAkzrTiEBtBbog+57cSaU9y37Gkdd+1jXCQrxP+jO
EUf7gnXWZvE6oeRroLvCt1fYn09k0CF8kKTbrPTSjC6igZR3uvTHyee74XQ9PRavvHax73T4UOEdQZX/P1ibS
QIWKbBRD5YQ1POYVjTayoltTnWLMxfEcAkkATJZLhpBEHST7kZWjrTS6J1dCReJc7a8Vsj/78HXvOIy"}]}

- match: { license_status:  "valid" }

- do:
    xpack.license.get: {}

## a license object has 11 attributes
- length: { license: 11 }

## bwc for licenses format
- do:
    xpack.license.post:
      acknowledge: true
      body: |
        {"licenses":[{"uid":"893361dc-9749-4997-93cb-
802e3d7fa4a8","type":"gold","issue_date_in_millis":1411948800000,"expiry_date_in_millis":1914278399999,"ma
x_nodes":1,"issued_to":"issued_to","issuer":"issuer","signature":"AAAAAwAAAA2T3vqdBBetKQaBgxipAAAB
mC9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVoMVZwUzRxVk1PSmkxakxZdW5IMlhlTHNoN1N2M
XMvRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHBkWkprZWZSQi9iNmRQNkw1SlpLN0lDalZCS09
5MXRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0U0MXZ0dDJHSERiZTVLOHNzSDByWnpoZEph
ZHBEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNiWktNWGM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOE
hQaVB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K055c28zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2V
E1qbVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9wTTJENDVvT1NCYklacUYyK2FwRW9xa0t6dldMbmMzS
GtQc3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRWUzh6UG5pdENGWFRreXNKNkE9PQAAAQB7pG
FYgawfLm9zzT80LvcLHjy1t/v2uSzCQWKdXXhrwSy4WrAH2uK/+PEiQ7aEpW5erLsyJ5KLA6OEZJDaP7r+mj
OPuLt0++l5j4DMn7ybMzOPHXWBc6LETE3+pp0GZPyOmwsDkZSRUegTtciR2R6z+mdnGrhOYM80y08KVW
whdU/DHw41MK7ePo6tq73Nz49y9lDgt9fxA0t4ggEBPbnTDDBVQ25AjauY8sa0M5eg9rDDRayw1KamYWrara8
PIGX+2YjhtUeQhmlCPdlxc9wECJ7/knPss5bI3ZoXQR3fyXhjcXNnHEIsblqLrMCal3pLxs7lI+KPYMa2ZYL/am4
```

P"}]}

```
- match: { license_status:  "valid" }

- do:
    xpack.license.get: {}

- length: { license: 11 }

## license version: 1.x
- do:
    xpack.license.post:
      acknowledge: true
      body: |
        {"licenses":[{"uid":"893361dc-9749-4997-93cb-
```

802e3d7fa4a8","type":"subscription","subscription_type":"gold","issue_date_in_millis":1411948800000,"feature":"
shield","expiry_date_in_millis":1914278399999,"max_nodes":1,"issued_to":"issuedTo","issuer":"issuer","signature
":"AAAAAQAAAA0LVAywwpSH94cyXr4zAAABmC9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVo
MVZwUzRxVk1PSmkxakxZdW5IMlhlTHNoN1N2MXMvRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0pp
THBkWkprZWZSQi9iNmRQNkw1SlpLN0lDalZCS095MXRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeG
wxb0U0MXZ0dDJHSERiZTVLOHNzSDByWnpoZEphZHBEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNi
WktNWGM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOEhQaVB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K
055c28zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2VE1qbVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9w
TTJENDVvT1NCYklacUYyK2FwRW9xa0t6dldMbmMzSGtQc3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2
bDRWUzh6UG5pdENGWFRreXNKNkE9PQAAAQA4qscc/URRZVdFoLwgy9dqybYEQLW8YLkiAyPV5XHHH
dtk+dtZIepiNEDkUXhSX2waVJlsNRF8/4kqplDfwNoD2TUM8fTgiIfiSiZYGDTGST+yW/5eAveEU5J5v1liBN27b
wkqL+V4YAa0Tcm7NKKwjScWKAHiTU3vF8chPkGfCHE0kQgVwPC9RE82pTw0s6/uR4PfLGNFfqPM0uiE5n
ucfVrtj89JQiO/KA/7ZyFbo7VTNXxZQt7T7rZWBCP9KIjptXzcWuk08Q5S+rSoJNYbFo3HGKtrCVsRz/55rceNtd
wKKXu1IwnSeir4I1/KLduQTtFLy0+1th87VS8T88UT"}]}
```

```
- match: { license_status:  "valid" }

- do:
    xpack.license.get: {}

- length: { license: 11 }

## multiple licenses version: 1.x
- do:
    xpack.license.post:
      acknowledge: true
      body: |
        {"licenses":[{"uid":"893361dc-9749-4997-93cb-
```

802e3d7fa4a8","type":"internal","subscription_type":"none","issue_date_in_millis":1411948800000,"feature":"shiel
d","expiry_date_in_millis":1440892799999,"max_nodes":1,"issued_to":"issuedTo","issuer":"issuer","signature":"A
AAAAQAAAA04Q4ky3rFyyWLFkytEAAABmC9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVoMVZw
UzRxVk1PSmkxakxZdW5IMlhlTHNoN1N2MXMvRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHBkW
kprZWZSQi9iNmRQNkw1SlpLN0lDalZCS095MXRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0U0
```

MXZ0dDJHSERiZTVLOHNzSDByWnpoZEphZHBEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNiWktNW
GM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOEhQaVB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K055c28z
SmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2VE1qbVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9wTTJEND
VvT1NCYklacUYyK2FwRW9xa0t6dldMbmMzSGtQc3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRWUz
h6UG5pdENGWFRreXNKNkE9PQAAAQBxMvUMn4h2E4R4TQMijahTxQj4LPQO4f1M79UxX/XkDlGcH+J5p
RHx08OtTRPsFL1lED+h+PIXx307Vo+PNDsOxrWvoYZeYBkOLAO3ny9vhQga+52jYhMxIuFrT9xbcSCSNpMh
GojgOIPU2WgiopVdVcimo1+Gk8VtklPB1wPwFzfOjOnPgp/Icx3WYpfkeAUUOyWUYiFIBAe4bnz84iF+xwLKb
gYk6aHF25ECBtdb/Uruhcm9+jEFpoIEUtCouvvk9C+NJZ4OickV4xpRgaRG2x9PONH8ZN0QGhGYhJGbisoCxu
DmlLsyVxqxfMu3n/r7/jdsEJScjAlSrsLDOu6H"},{"uid":"893361dc-9749-4997-93cb-
802e3dofh7aa","type":"internal","subscription_type":"none","issue_date_in_millis":1443484800000,"feature":"watc
her","expiry_date_in_millis":1914278399999,"max_nodes":1,"issued_to":"issuedTo","issuer":"issuer","signature":"
AAAAAQAAAA0Sc90guRIaQEmgLvMnAAABmC9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVoMV
ZwUzRxVk1PSmkxakxZdW5IMlhlTHNoN1N2MXMvRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHB
kWkprZWZSQi9iNmRQNkw1SlpLN0lDalZCS095MXRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0
U0MXZ0dDJHSERiZTVLOHNzSDByWnpoZEphZHBEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNiWktN
WGM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOEhQaVB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K055c2
8zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2VE1qbVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9wTTJEN
DVvT1NCYklacUYyK2FwRW9xa0t6dldMbmMzSGtQc3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRW
Uzh6UG5pdENGWFRreXNKNkE9PQAAAQCQ94dju0pnDZR3Uuypi0ic3aQJ+nvVqe+U8u79Dga5n1qIjcHDh7H
vIBJEkF+tnVPlo/PXV/x7BZSwVY1PVErit+6rYix1yuHEgqwxmx/VdRICjCaZM6tk0Ob4dZCPv6Ebn2Mmk89KH
C/PwiLPqF6QfwV/Pkpa8k2A3ORJmvYSDvXhe6tCs8dqc4ebrsFxqrZjwWh5CZSpzqqZBFXlngDv2N0hHhpGlueR
szD0JJ5dfEL5ZA1DDOrgO9OJVejSHyRqe1L5QRUNdXPVfS+EAG0Dd1cNdJ/sMpYCPnVjbw6iq2/YgM3cuztsX
VBY7ij4WnoP3ce7Zjs9TwHn+IqzftC6"}]}

```
 - match: { license_status:  "valid" }

 - do:
     xpack.license.get: {}

 - length: { license: 11 }
 - match: { license.uid: "893361dc-9749-4997-93cb-802e3dofh7aa" }
---
"Should throw 404 after license deletion":
 - do:
     xpack.license.delete: {}

 - match: { acknowledged: true }

 - do:
     xpack.license.get: {}
     catch: missing

---
"Should install a feature type license":

   # VERSION_NO_FEATURE_TYPE license version
 - do:
     xpack.license.post:
```

acknowledge: true
body: |
{"license": {"uid":"893361dc-9749-4997-93cb-
802e3d7fa4a8","type":"gold","issue_date_in_millis":1411948800000,"expiry_date_in_millis":1914278399999,"ma
x_nodes":1,"issued_to":"issued_to","issuer":"issuer","signature":"AAAAAgAAAA3U8+YmnvwC+CWsV/mRAA
ABmC9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVoMVZwUzRxVk1PSmkxakxZdW5IMlhlTHNoN1N
2MXMvRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHBkWkprZWZSQi9iNmRQNkw1SlpLN0lDalZC
S095MXRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0U0MXZ0dDJHSERiZTVLOHNzSDByWnpoZ
EphZHBEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNiWktNWGM5TUhQV2plaUo4Q1JOUml4MXNuL0pS
OEhQaVB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K055c28zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3
o2VE1qbVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9wTTJENDVvT1NCYklacUYyK2FwRW9xa0t6dldMbm
MzSGtQc3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRWUzh6UG5pdENGWFRreXNKNkE9PQAAAQB
e8GfzDm6T537Iuuvjetb3xK5dvg0K5NQapv+rczWcQFxgCuzbF8plkgetP1aAGZP4uRESDQPMlOCsx4d0UqqAm
9f7GbBQ3l93P+PogInPFeEH9NvOmaAQovmxVM9SE6DsDqlX4cXSO+bgWpXPTd2LmpoQc1fXd6BZ8GeuyYp
VHVKp9hVU0tAYjw6HzYOE7+zuO1oJYOxElqy66AnIfkvHrvni+flym3tE7tDTgsDRaz7W3iBhaqiSntEqabEkvH
dPHQdSR99XGaEvnHO1paK01/35iZF6OXHsF7CCj+558GRXiVxzueOe7TsGSSt8g7YjZwV9bRCyU7oB4B/nidg
I"}}

- match: { license_status: "valid" }

- do:
xpack.license.get: {}

- length: { license: 11 }

"Cannot start basic":

- do:
catch: bad_request
xpack.license.post:
acknowledge: true
body: |
{"license":{"uid":"893361dc-9749-4997-93cb-
802e3d7fa4a8","type":"basic","issue_date_in_millis":1411948800000,"expiry_date_in_millis":1914278399999,"ma
x_nodes":1,"issued_to":"issuedTo","issuer":"issuer","signature":"AAAAAgAAAA0lKPZ0a7aZquUltho/AAABmC
9ZN0hjZDBGYnVyRXpCOW5Bb3FjZDAxOWpSbTVoMVZwUzRxVk1PSmkxakxZdW5IMlhlTHNoN1N2MXM
vRFk4d3JTZEx3R3RRZ0pzU3lobWJKZnQvSEFva0ppTHBkWkprZWZSQi9iNmRQNkw1SlpLN0lDalZCS095M
XRGN1lIZlpYcVVTTnFrcTE2dzhJZmZrdFQrN3JQeGwxb0U0MXZ0dDJHSERiZTVLOHNzSDByWnpoZEphZH
BEZjUrTVBxRENNSXNsWWJjZllaODdzVmEzUjNiWktNWGM5TUhQV2plaUo4Q1JOUml4MXNuL0pSOEhQa
VB2azhmUk9QVzhFeTFoM1Q0RnJXSG53MWk2K055c28zSmRnVkF1b2JSQkFLV2VXUmVHNDZ2R3o2VE1q
bVNQS2lxOHN5bUErZlNIWkZSVmZIWEtaSU9wTTJENDVvT1NCYklacUYyK2FwRW9xa0t6dldMbmMzSGtQ
c3FWOTgzZ3ZUcXMvQkt2RUZwMFJnZzlvL2d2bDRWUzh6UG5pdENGWFRreXNKNkE9PQAAAQAALuQ44
S3IG6SzolcXVJ6Z4CIXORDrYQ+wdLCeey0XdujTslAOj+k+vNgo6wauc7Uswi01esHu4lb5IgpvKy7RRCbh5bj/z2
ubu2qMJqopp9BQyD7VQjVfqmG6seUMJwJ1a5Avvm9r41YPSPcrii3bKK2e1l6jK6N8ibCvnTyY/XkYGCJrBWT
SJePDbg6ErbyodrZ37x1StLbPWcNAkmweyHjDJnvYnbeZZO7A3NmubXZjW7Ttf8/YwQyE00PqMcl7fVPY3hk
KpAeHf8aaJbqkKYbqZuER3EWJX7ZvLVb1dNdNg8aXRn7YrkQcYwWgptYQpfV+D7yEJ4j5muAEoler"}}

- match: { error.root_cause.0.reason: 'Installing basic licenses is no longer allowed. Use the POST

```
/_xpack/license/start_basic API to install a basic license that does not expire.' }
---
"Should fail gracefully when body content is not provided":

 - do:
     catch: bad_request
     xpack.license.post:
       acknowledge: true

 - match: { error.root_cause.0.reason: 'The license must be provided in the request body' }
---
"Current license is trial means not eligle to start trial":

 - do:
     xpack.license.get_trial_status: {}

 - match: { eligible_to_start_trial: false }

 - do:
     xpack.license.post_start_basic:
       acknowledge: true

 - match: { basic_was_started: true }

 - do:
     xpack.license.get_trial_status: {}

 - match: { eligible_to_start_trial: false }

 - do:
     catch: forbidden
     xpack.license.post_start_trial:
       acknowledge: true

 - match: { trial_was_started: false }
 - match: { error_message: "Operation failed: Trial was already activated." }
---
"Trial license cannot be basic":
 - do:
     catch: bad_request
     xpack.license.post_start_trial:
       type: "basic"
       acknowledge: true
---
"Can start basic license if do not already have basic":
 - do:
     xpack.license.get_basic_status: {}
```

```
 - match: { eligible_to_start_basic: true }

 - do:
     xpack.license.post_start_basic:
       acknowledge: true

 - match: { basic_was_started: true }
 - match: { acknowledged: true }

 - do:
       xpack.license.get_basic_status: {}

 - match: { eligible_to_start_basic: false }

 - do:
     catch: forbidden
     xpack.license.post_start_basic: {}

 - match: { basic_was_started: false }
 - match: { acknowledged: true }
 - match: { error_message: "Operation failed: Current license is basic." }
---
"Must acknowledge to start basic":
 - do:
     xpack.license.post_start_basic: {}

 - match: { basic_was_started: false }
 - match: { acknowledged: false }
 - match: { error_message: "Operation failed: Needs acknowledgement." }
{
 "xpack.license.post_start_trial": {
   "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
   "methods": ["POST"],
   "url": {
     "path": "/_xpack/license/start_trial",
     "paths": ["/_xpack/license/start_trial"],
     "parts" : {
     },
     "params": {
       "type": {
         "type" : "string",
         "description" : "The type of trial license to generate (default: \"trial\")"
       },
       "acknowledge": {
         "type" : "boolean",
         "description" : "whether the user has acknowledged acknowledge messages (default: false)"
       }
     }
```

```
  },
  "body": null
 }
}
{
 "xpack.license.get": {
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
  "methods": ["GET"],
  "url": {
    "path": "/_xpack/license",
    "paths": ["/_xpack/license"],
    "parts" : {
    },
    "params": {
      "local": {
        "type" : "boolean",
        "description" : "Return local information, do not retrieve the state from master node (default: false)"
      }
    }
  },
  "body": null
 }
}
{
 "xpack.license.delete": {
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
  "methods": ["DELETE"],
  "url": {
    "path": "/_xpack/license",
    "paths": ["/_xpack/license"],
    "parts" : {}
  },
  "body": null
 }
}
{
 "xpack.license.get_basic_status": {
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
  "methods": ["GET"],
  "url": {
    "path": "/_xpack/license/basic_status",
    "paths": ["/_xpack/license/basic_status"],
    "parts" : {
    },
    "params": {
    }
  },
  "body": null
```

```
 }
}
{
 "xpack.license.post": {
   "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
   "methods": ["PUT", "POST"],
   "url": {
     "path": "/_xpack/license",
     "paths": ["/_xpack/license"],
     "parts" : {
     },
     "params": {
       "acknowledge": {
         "type" : "boolean",
         "description" : "whether the user has acknowledged acknowledge messages (default: false)"
       }
     }
   },
   "body": {
     "description" : "licenses to be installed"
   }
 }
}
{
 "xpack.license.post_start_basic": {
   "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
   "methods": ["POST"],
   "url": {
     "path": "/_xpack/license/start_basic",
     "paths": ["/_xpack/license/start_basic"],
     "parts" : {
     },
     "params": {
       "acknowledge": {
         "type" : "boolean",
         "description" : "whether the user has acknowledged acknowledge messages (default: false)"
       }
     }
   },
   "body": null
 }
}
{
 "xpack.license.get_trial_status": {
   "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html",
   "methods": ["GET"],
   "url": {
     "path": "/_xpack/license/trial_status",
```

"paths": ["/_xpack/license/trial_status"],
"parts" : {
},
"params": {
}
},
"body": null
}
}
You may use under either the Apache License Version 2.0 or the BSD
3-Clause License.

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

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

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

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

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

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

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

5. Submission of Contributions. Unless You explicitly state otherwise,

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

Copyright (c) 2011, Mike Samuel
All rights reserved.

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

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

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

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a)the power, direct or indirect, to cause

the direction or management of such entity, whether by contract or otherwise, or (b)ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section2.1(b) above, no patent license is granted: (1)for code that You delete from the Original Software, or (2)for infringements caused by: (i)the modification of the Original Software, or (ii)the combination of the Original Software with other software or devices.

2.2. Contributor Grant.
Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1)Modifications made by that Contributor (or portions thereof); and (2)the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section2.2(b) above, no patent license is granted: (1)for any code that Contributor has deleted from the Contributor Version; (2)for infringements caused by: (i)third party modifications of Contributor Version, or (ii)the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3)under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a

medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.
You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.
You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.
You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.
You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.
Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a

notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a)rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b)otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER

CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The GlassFish code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal

Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this

License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

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

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software

with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such

Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial

Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED,

INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for

the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

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

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and

to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

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

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

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

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

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

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

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

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is

to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

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

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

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

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

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

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

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

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and

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

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

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

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

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

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

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

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

executable.

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

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

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

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

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

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

circumstances.

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

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

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

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

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

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

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

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

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

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

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

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

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

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

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

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

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

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

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

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

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

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

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

#

Certain source files distributed by Oracle America, Inc. and/or its
affiliates are subject to the following clarification and special
exception to the GPLv2, based on the GNU Project exception for its
Classpath libraries, known as the GNU Classpath Exception, but only
where Oracle has expressly included in the particular source file's
header the words "Oracle designates this particular file as subject to
the "Classpath" exception as provided by Oracle in the LICENSE file
that accompanied this code."

You should also note that Oracle includes multiple, independent
programs in this software package. Some of those programs are provided
under licenses deemed incompatible with the GPLv2 by the Free Software
Foundation and others. For example, the package includes programs
licensed under the Apache License, Version 2.0. Such programs are

licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result. Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

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

CLASSPATH EXCEPTION
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.
{
 "metadata": {
  "name": "X-Pack Monitoring: License Expiration",
  "xpack": {
   "link": "license",
   "expires_days": [60, 30, 14, 7],
   "severity": 0,

```
    "alert_index": ".monitoring-alerts-6",
    "cluster_uuid": "${monitoring.watch.cluster_uuid}",
    "type": "monitoring",
    "version_created": 6040099,
    "watch": "${monitoring.watch.id}"
  }
},
"trigger": {
  "schedule": {
    "interval": "1m"
  }
},
"input": {
  "chain": {
    "inputs": [
      {
        "check": {
          "search": {
            "request": {
              "indices": [
                ".monitoring-es-*"
              ],
              "body": {
                "size": 1,
                "sort": [
                  {
                    "timestamp": {
                      "order": "desc"
                    }
                  }
                ],
                "_source": [
                  "license.*"
                ],
                "query": {
                  "bool": {
                    "filter": [
                      {
                        "term": {
                          "cluster_uuid": "{{ctx.metadata.xpack.cluster_uuid}}"
                        }
                      },
                      {
                        "term": {
                          "type": "cluster_stats"
                        }
                      },
                      {
```

```
                "range": {
                  "timestamp": {
                    "gte": "now-2m"
                  }
                }
              }
            ]
          }
        }
      }
    }
  }
},
{
  "alert": {
    "search": {
      "request": {
        "indices": [
          ".monitoring-alerts-6"
        ],
        "body": {
          "size": 1,
          "terminate_after": 1,
          "query": {
            "bool": {
              "filter": {
                "term": {
                  "_id": "{{ctx.watch_id}}"
                }
              }
            }
          },
          "sort": [
            { "timestamp": { "order": "desc" } }
          ]
        }
      }
    }
  }
},
{
  "kibana_settings": {
    "search": {
      "request": {
        "indices": [
          ".monitoring-kibana-6-*"
        ],
```

```
            "body": {
              "size": 1,
              "query": {
                "bool": {
                  "filter": {
                    "term": {
                      "type": "kibana_settings"
                    }
                  }
                }
              },
              "sort": [
                {
                  "timestamp": {
                    "order": "desc"
                  }
                }
              ]
            }
          }
        }
      }
    }
  ]
}
},
"condition": {
  "script": {
    "source": "if (ctx.payload.check.hits.total == 0) {return false;}def license =
ctx.payload.check.hits.hits[0]._source.license;if (license == null) {return false;}ctx.vars.fails_check = false;Instant
expiry = Instant.ofEpochMilli(license.expiry_date_in_millis);ctx.vars.expiry = expiry;if (license.status != 'active')
{ctx.vars.expired = true;ctx.vars.fails_check = true;ctx.metadata.xpack.severity = 2001;} else {Instant now =
Instant.ofEpochMilli(new Date().getTime());ctx.vars.now = now;for (int i = ctx.metadata.xpack.expires_days.length
- 1;i > -1;--i) {if (license.type == 'trial' && i < 2) {break;}Instant fromNow =
now.plusSeconds(ctx.metadata.xpack.expires_days[i] * 24 * 60 * 60);if (fromNow.isAfter(expiry))
{ctx.vars.fails_check = true;ctx.metadata.xpack.severity = i * 1000;break;}}}ctx.vars.not_resolved =
(ctx.payload.alert.hits.total == 1 && ctx.payload.alert.hits.hits[0]._source.resolved_timestamp ==
null);ctx.vars.update = ctx.vars.fails_check || ctx.vars.not_resolved;"
  }
},
"transform": {
  "script": {
    "source": "ctx.vars.email_recipient = (ctx.payload.kibana_settings.hits.total > 0 &&
ctx.payload.kibana_settings.hits.hits[0]._source.kibana_settings.xpack != null) ?
ctx.payload.kibana_settings.hits.hits[0]._source.kibana_settings.xpack.default_admin_email : null;ctx.vars.is_new =
ctx.vars.fails_check && !ctx.vars.not_resolved;ctx.vars.is_resolved = !ctx.vars.fails_check &&
ctx.vars.not_resolved;def alertMessage = null;if (ctx.vars.fails_check) { alertMessage = 'Update your license.';} if
(ctx.vars.not_resolved) { ctx.payload = ctx.payload.alert.hits.hits[0]._source;ctx.payload.metadata =
```

```
ctx.metadata.xpack;if (ctx.vars.fails_check == false) { ctx.payload.resolved_timestamp = ctx.execution_time;} }
else { ctx.payload = [ 'timestamp': ctx.execution_time, 'prefix': 'This cluster\\'s license is going to expire in
{{#relativeTime}}metadata.time{{/relativeTime}} at {{#absoluteTime}}metadata.time{{/absoluteTime}}.',
'message': alertMessage, 'metadata': ctx.metadata.xpack ];} if (ctx.vars.fails_check) { ctx.payload.metadata.time =
ctx.vars.expiry.toString();} ctx.payload.update_timestamp = ctx.execution_time;return ctx.payload;"
   }
 },
 "actions": {
   "add_to_alerts_index": {
     "index": {
       "index": ".monitoring-alerts-6",
       "doc_type": "doc",
       "doc_id": "${monitoring.watch.unique_id}"
     }
   },
   "send_email_to_admin": {
     "condition": {
       "script": "return ctx.vars.email_recipient != null && (ctx.vars.is_new || ctx.vars.is_resolved)"
     },
     "email": {
       "to": "X-Pack Admin <{{ctx.vars.email_recipient}}>",
       "from": "X-Pack Admin <{{ctx.vars.email_recipient}}>",
       "subject":
"[{{#ctx.vars.is_new}}NEW{{/ctx.vars.is_new}}{{#ctx.vars.is_resolved}}RESOLVED{{/ctx.vars.is_resolved}}]
{{ctx.metadata.name}}",
       "body": {
         "text": "{{#ctx.vars.is_resolved}}This cluster alert has been resolved: {{/ctx.vars.is_resolved}} This cluster's
license {{#ctx.vars.is_new}}is going to expire on
{{ctx.payload.metadata.time}}{{/ctx.vars.is_new}}{{#ctx.vars.is_resolved}}was going to
expire{{ctx.payload.metadata.time}}{{/ctx.vars.is_resolved}}.
{{#ctx.vars.is_new}}{{ctx.payload.message}}{{/ctx.vars.is_new}}"
       }
     }
   }
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchSecurityException;
import org.elasticsearch.action.support.PlainActionFuture;
import org.elasticsearch.action.support.master.AcknowledgedResponse;
import org.elasticsearch.client.transport.TransportClient;
import org.elasticsearch.cluster.ClusterState;
```

```
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.license.License.OperationMode;
import org.elasticsearch.persistent.PersistentTasksCustomMetaData;
import org.elasticsearch.rest.RestStatus;
import org.elasticsearch.test.junit.annotations.TestLogging;
import org.elasticsearch.transport.Transport;
import org.elasticsearch.xpack.core.TestXPackTransportClient;
import org.elasticsearch.xpack.core.XPackField;
import org.elasticsearch.xpack.core.ml.action.CloseJobAction;
import org.elasticsearch.xpack.core.ml.action.DeleteDatafeedAction;
import org.elasticsearch.xpack.core.ml.action.DeleteJobAction;
import org.elasticsearch.xpack.core.ml.action.GetDatafeedsStatsAction;
import org.elasticsearch.xpack.core.ml.action.GetJobsStatsAction;
import org.elasticsearch.xpack.core.ml.action.OpenJobAction;
import org.elasticsearch.xpack.core.ml.action.PutDatafeedAction;
import org.elasticsearch.xpack.core.ml.action.PutJobAction;
import org.elasticsearch.xpack.core.ml.action.StartDatafeedAction;
import org.elasticsearch.xpack.core.ml.action.StopDatafeedAction;
import org.elasticsearch.xpack.core.ml.client.MachineLearningClient;
import org.elasticsearch.xpack.core.ml.datafeed.DatafeedState;
import org.elasticsearch.xpack.core.ml.job.config.JobState;
import org.elasticsearch.xpack.ml.LocalStateMachineLearning;
import org.elasticsearch.xpack.ml.support.BaseMlIntegTestCase;
import org.junit.Before;

import java.util.Collections;

import static org.hamcrest.Matchers.containsString;
import static org.hamcrest.Matchers.hasItem;
import static org.hamcrest.Matchers.is;

@TestLogging("org.elasticsearch.xpack.ml.action:DEBUG")
public class MachineLearningLicensingTests extends BaseMlIntegTestCase {

  @Before
  public void resetLicensing() {
    enableLicensing();

    ensureStableCluster(1);
    ensureYellow();
  }

  public void testMachineLearningPutJobActionRestricted() throws Exception {
    String jobId = "testmachinelearningputjobactionrestricted";
    // Pick a license that does not allow machine learning
    License.OperationMode mode = randomInvalidLicenseType();
```

```
        enableLicensing(mode);
        assertMLAllowed(false);
        // test that license restricted apis do not work
        Settings settings = internalCluster().transportClient().settings();
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            PlainActionFuture<PutJobAction.Response> listener = PlainActionFuture. newFuture();
            new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), listener);
            listener.actionGet();
            fail("put job action should not be enabled!");
        } catch (ElasticsearchSecurityException e) {
            assertThat(e.status(), is(RestStatus.FORBIDDEN));
            assertThat(e.getMessage(), containsString("non-compliant"));
            assertThat(e.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA),
hasItem(XPackField.MACHINE_LEARNING));
        }

        // Pick a license that does allow machine learning
        mode = randomValidLicenseType();
        enableLicensing(mode);
        assertMLAllowed(true);
        // test that license restricted apis do now work
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            PlainActionFuture<PutJobAction.Response> listener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), listener);
            PutJobAction.Response response = listener.actionGet();
            assertNotNull(response);
        }
    }

    public void testMachineLearningOpenJobActionRestricted() throws Exception {
        String jobId = "testmachinelearningopenjobactionrestricted";
        assertMLAllowed(true);
        // test that license restricted apis do now work
        Settings settings = internalCluster().transportClient().settings();
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
            PutJobAction.Response response = putJobListener.actionGet();
            assertNotNull(response);
        }

        // Pick a license that does not allow machine learning
```

```
        License.OperationMode mode = randomInvalidLicenseType();
        enableLicensing(mode);
        assertMLAllowed(false);
        // test that license restricted apis do not work
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), listener);
            listener.actionGet();
            fail("open job action should not be enabled!");
        } catch (ElasticsearchSecurityException e) {
            assertThat(e.status(), is(RestStatus.FORBIDDEN));
            assertThat(e.getMessage(), containsString("non-compliant"));
            assertThat(e.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA),
hasItem(XPackField.MACHINE_LEARNING));
        }

        // Pick a license that does allow machine learning
        mode = randomValidLicenseType();
        enableLicensing(mode);
        assertMLAllowed(true);

        // now that the license is invalid, the job should get closed:
        assertBusy(() -> {
            JobState jobState = getJobStats(jobId).getState();
            assertEquals(JobState.CLOSED, jobState);
        });

        // test that license restricted apis do now work
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), listener);
            AcknowledgedResponse response = listener.actionGet();
            assertNotNull(response);
        }
    }

    public void testMachineLearningPutDatafeedActionRestricted() throws Exception {
        String jobId = "testmachinelearningputdatafeedactionrestricted";
        String datafeedId = jobId + "-datafeed";
        assertMLAllowed(true);
        // test that license restricted apis do now work
        Settings settings = internalCluster().transportClient().settings();
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
```

```
());
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
    }

    // Pick a license that does not allow machine learning
    License.OperationMode mode = randomInvalidLicenseType();
    enableLicensing(mode);
    assertMLAllowed(false);
    // test that license restricted apis do not work
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutDatafeedAction.Response> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putDatafeed(
                new PutDatafeedAction.Request(createDatafeed(datafeedId, jobId, Collections.singletonList(jobId))),
listener);
        listener.actionGet();
        fail("put datafeed action should not be enabled!");
    } catch (ElasticsearchSecurityException e) {
        assertThat(e.status(), is(RestStatus.FORBIDDEN));
        assertThat(e.getMessage(), containsString("non-compliant"));
        assertThat(e.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA),
hasItem(XPackField.MACHINE_LEARNING));
    }

    // Pick a license that does allow machine learning
    mode = randomValidLicenseType();
    enableLicensing(mode);
    assertMLAllowed(true);
    // test that license restricted apis do now work
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutDatafeedAction.Response> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putDatafeed(
                new PutDatafeedAction.Request(createDatafeed(datafeedId, jobId, Collections.singletonList(jobId))),
listener);
        PutDatafeedAction.Response response = listener.actionGet();
        assertNotNull(response);
    }
  }

  public void testAutoCloseJobWithDatafeed() throws Exception {
    String jobId = "testautoclosejobwithdatafeed";
    String datafeedId = jobId + "-datafeed";
```

```
    assertMLAllowed(true);
    String datafeedIndex = jobId + "-data";
    prepareCreate(datafeedIndex).addMapping("type", "{\"type\":{\"properties\":{\"time\":{\"type\":\"date\"}}}}",
        XContentType.JSON).get();
    Settings settings = internalCluster().transportClient().settings();
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        // put job
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
        // put datafeed
        PlainActionFuture<PutDatafeedAction.Response> putDatafeedListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putDatafeed(
                new PutDatafeedAction.Request(createDatafeed(datafeedId, jobId,
                        Collections.singletonList(datafeedIndex))), putDatafeedListener);
        PutDatafeedAction.Response putDatafeedResponse = putDatafeedListener.actionGet();
        assertNotNull(putDatafeedResponse);
        // open job
        PlainActionFuture<AcknowledgedResponse> openJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), openJobListener);
        AcknowledgedResponse openJobResponse = openJobListener.actionGet();
        assertNotNull(openJobResponse);
        // start datafeed
        PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).startDatafeed(new StartDatafeedAction.Request(datafeedId, 0L),
listener);
        listener.actionGet();
    }

    if (randomBoolean()) {
        enableLicensing(randomInvalidLicenseType());
    } else {
        disableLicensing();
    }
    assertMLAllowed(false);

    // now that the license is invalid, the job should be closed and datafeed stopped:
    assertBusy(() -> {
        JobState jobState = getJobStats(jobId).getState();
        assertEquals(JobState.CLOSED, jobState);

        DatafeedState datafeedState = getDatafeedStats(datafeedId).getDatafeedState();
        assertEquals(DatafeedState.STOPPED, datafeedState);

        ClusterState state = client().admin().cluster().prepareState().get().getState();
```

```
        PersistentTasksCustomMetaData tasks = state.metaData().custom(PersistentTasksCustomMetaData.TYPE);
        assertEquals(0, tasks.taskMap().size());
    });

    enableLicensing(randomValidLicenseType());
    assertMLAllowed(true);

    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        // open job
        PlainActionFuture<AcknowledgedResponse> openJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), openJobListener);
        AcknowledgedResponse openJobResponse = openJobListener.actionGet();
        assertNotNull(openJobResponse);
        // start datafeed
        PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).startDatafeed(new StartDatafeedAction.Request(datafeedId, 0L),
listener);
        listener.actionGet();
    }

    assertBusy(() -> {
        JobState jobState = getJobStats(jobId).getState();
        assertEquals(JobState.OPENED, jobState);

        DatafeedState datafeedState = getDatafeedStats(datafeedId).getDatafeedState();
        assertEquals(DatafeedState.STARTED, datafeedState);

        ClusterState state = client().admin().cluster().prepareState().get().getState();
        PersistentTasksCustomMetaData tasks = state.metaData().custom(PersistentTasksCustomMetaData.TYPE);
        assertEquals(2, tasks.taskMap().size());
    });

    if (randomBoolean()) {
        enableLicensing(randomInvalidLicenseType());
    } else {
        disableLicensing();
    }
    assertMLAllowed(false);

    // now that the license is invalid, the job should be closed and datafeed stopped:
    assertBusy(() -> {
        JobState jobState = getJobStats(jobId).getState();
        assertEquals(JobState.CLOSED, jobState);

        DatafeedState datafeedState = getDatafeedStats(datafeedId).getDatafeedState();
        assertEquals(DatafeedState.STOPPED, datafeedState);
```

```
        ClusterState state = client().admin().cluster().prepareState().get().getState();
        PersistentTasksCustomMetaData tasks = state.metaData().custom(PersistentTasksCustomMetaData.TYPE);
        assertEquals(0, tasks.taskMap().size());
    });
}

public void testMachineLearningStartDatafeedActionRestricted() throws Exception {
    String jobId = "testmachinelearningstartdatafeedactionrestricted";
    String datafeedId = jobId + "-datafeed";
    assertMLAllowed(true);
    String datafeedIndex = jobId + "-data";
    prepareCreate(datafeedIndex).addMapping("type", "{\"type\":{\"properties\":{\"time\":{\"type\":\"date\"}}}}",
        XContentType.JSON).get();
    // test that license restricted apis do now work
    Settings settings = internalCluster().transportClient().settings();
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
        PlainActionFuture<PutDatafeedAction.Response> putDatafeedListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putDatafeed(
                new PutDatafeedAction.Request(createDatafeed(datafeedId, jobId,
                        Collections.singletonList(datafeedIndex))), putDatafeedListener);
        PutDatafeedAction.Response putDatafeedResponse = putDatafeedListener.actionGet();
        assertNotNull(putDatafeedResponse);
        PlainActionFuture<AcknowledgedResponse> openJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), openJobListener);
        AcknowledgedResponse openJobResponse = openJobListener.actionGet();
        assertNotNull(openJobResponse);
    }

    // Pick a license that does not allow machine learning
    License.OperationMode mode = randomInvalidLicenseType();
    enableLicensing(mode);
    assertMLAllowed(false);

    // now that the license is invalid, the job should get closed:
    assertBusy(() -> {
        JobState jobState = getJobStats(jobId).getState();
        assertEquals(JobState.CLOSED, jobState);
        ClusterState state = client().admin().cluster().prepareState().get().getState();
        PersistentTasksCustomMetaData tasks = state.metaData().custom(PersistentTasksCustomMetaData.TYPE);
        assertEquals(0, tasks.taskMap().size());
    });
```

```
        // test that license restricted apis do not work
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).startDatafeed(new StartDatafeedAction.Request(datafeedId, 0L),
listener);
            listener.actionGet();
            fail("start datafeed action should not be enabled!");
        } catch (ElasticsearchSecurityException e) {
            assertThat(e.status(), is(RestStatus.FORBIDDEN));
            assertThat(e.getMessage(), containsString("non-compliant"));
            assertThat(e.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA),
hasItem(XPackField.MACHINE_LEARNING));
        }

        // Pick a license that does allow machine learning
        mode = randomValidLicenseType();
        enableLicensing(mode);
        assertMLAllowed(true);
        // test that license restricted apis do now work
        try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
            // re-open job now that the license is valid again
            PlainActionFuture<AcknowledgedResponse> openJobListener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), openJobListener);
            AcknowledgedResponse openJobResponse = openJobListener.actionGet();
            assertNotNull(openJobResponse);

            PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
            new MachineLearningClient(client).startDatafeed(new StartDatafeedAction.Request(datafeedId, 0L),
listener);
            AcknowledgedResponse response = listener.actionGet();
            assertNotNull(response);
        }
    }

    public void testMachineLearningStopDatafeedActionNotRestricted() throws Exception {
        String jobId = "testmachinelearningstopdatafeedactionnotrestricted";
        String datafeedId = jobId + "-datafeed";
        assertMLAllowed(true);
        String datafeedIndex = jobId + "-data";
        prepareCreate(datafeedIndex).addMapping("type", "{\"type\":{\"properties\":{\"time\":{\"type\":\"date\"}}}}",
            XContentType.JSON).get();
        // test that license restricted apis do now work
        Settings settings = internalCluster().transportClient().settings();
```

```
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
        PlainActionFuture<PutDatafeedAction.Response> putDatafeedListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putDatafeed(
                new PutDatafeedAction.Request(createDatafeed(datafeedId, jobId,
                        Collections.singletonList(datafeedIndex))), putDatafeedListener);
        PutDatafeedAction.Response putDatafeedResponse = putDatafeedListener.actionGet();
        assertNotNull(putDatafeedResponse);
        PlainActionFuture<AcknowledgedResponse> openJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), openJobListener);
        AcknowledgedResponse openJobResponse = openJobListener.actionGet();
        assertNotNull(openJobResponse);
        PlainActionFuture<AcknowledgedResponse> startDatafeedListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).startDatafeed(
            new StartDatafeedAction.Request(datafeedId, 0L), startDatafeedListener);
        AcknowledgedResponse startDatafeedResponse = startDatafeedListener.actionGet();
        assertNotNull(startDatafeedResponse);
    }

    boolean invalidLicense = randomBoolean();
    if (invalidLicense) {
        enableLicensing(randomInvalidLicenseType());
    } else {
        enableLicensing(randomValidLicenseType());
    }

    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<StopDatafeedAction.Response> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).stopDatafeed(new StopDatafeedAction.Request(datafeedId), listener);
        if (invalidLicense) {
            // the stop datafeed due to invalid license happens async, so check if the datafeed turns into stopped state:
            assertBusy(() -> {
                GetDatafeedsStatsAction.Response response =
                        new MachineLearningClient(client)
                                .getDatafeedsStats(new GetDatafeedsStatsAction.Request(datafeedId)).actionGet();
                assertEquals(DatafeedState.STOPPED, response.getResponse().results().get(0).getDatafeedState());
            });
        } else {
            listener.actionGet();
        }
```

```
        if (invalidLicense) {
            // the close due to invalid license happens async, so check if the job turns into closed state:
            assertBusy(() -> {
                GetJobsStatsAction.Response response =
                        new MachineLearningClient(client).getJobsStats(new
GetJobsStatsAction.Request(jobId)).actionGet();
                assertEquals(JobState.CLOSED, response.getResponse().results().get(0).getState());
            });
        }
    }
  }

  public void testMachineLearningCloseJobActionNotRestricted() throws Exception {
    String jobId = "testmachinelearningclosejobactionnotrestricted";
    assertMLAllowed(true);
    // test that license restricted apis do now work
    Settings settings = internalCluster().transportClient().settings();
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
        PlainActionFuture<AcknowledgedResponse> openJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).openJob(new OpenJobAction.Request(jobId), openJobListener);
        AcknowledgedResponse openJobResponse = openJobListener.actionGet();
        assertNotNull(openJobResponse);
    }

    boolean invalidLicense = randomBoolean();
    if (invalidLicense) {
        enableLicensing(randomInvalidLicenseType());
    } else {
        enableLicensing(randomValidLicenseType());
    }

    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<CloseJobAction.Response> listener = PlainActionFuture.newFuture();
        CloseJobAction.Request request = new CloseJobAction.Request(jobId);
        request.setCloseTimeout(TimeValue.timeValueSeconds(20));
        if (invalidLicense) {
            // the close due to invalid license happens async, so check if the job turns into closed state:
            assertBusy(() -> {
                GetJobsStatsAction.Response response =
                        new MachineLearningClient(client).getJobsStats(new
```

```
GetJobsStatsAction.Request(jobId)).actionGet();
                assertEquals(JobState.CLOSED, response.getResponse().results().get(0).getState());
            });
        } else {
            new MachineLearningClient(client).closeJob(request, listener);
            listener.actionGet();
        }
    }
}

public void testMachineLearningDeleteJobActionNotRestricted() throws Exception {
    String jobId = "testmachinelearningclosejobactionnotrestricted";
    assertMLAllowed(true);
    // test that license restricted apis do now work
    Settings settings = internalCluster().transportClient().settings();
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
    }

    // Pick a random license
    License.OperationMode mode = randomLicenseType();
    enableLicensing(mode);

    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).deleteJob(new DeleteJobAction.Request(jobId), listener);
        listener.actionGet();
    }
}

public void testMachineLearningDeleteDatafeedActionNotRestricted() throws Exception {
    String jobId = "testmachinelearningdeletedatafeedactionnotrestricted";
    String datafeedId = jobId + "-datafeed";
    assertMLAllowed(true);
    // test that license restricted apis do now work
    Settings settings = internalCluster().transportClient().settings();
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<PutJobAction.Response> putJobListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putJob(new PutJobAction.Request(createJob(jobId)), putJobListener);
```

```
        PutJobAction.Response putJobResponse = putJobListener.actionGet();
        assertNotNull(putJobResponse);
        PlainActionFuture<PutDatafeedAction.Response> putDatafeedListener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).putDatafeed(
                new PutDatafeedAction.Request(createDatafeed(datafeedId, jobId,
                        Collections.singletonList(jobId))), putDatafeedListener);
        PutDatafeedAction.Response putDatafeedResponse = putDatafeedListener.actionGet();
        assertNotNull(putDatafeedResponse);
    }

    // Pick a random license
    License.OperationMode mode = randomLicenseType();
    enableLicensing(mode);

    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateMachineLearning.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PlainActionFuture<AcknowledgedResponse> listener = PlainActionFuture.newFuture();
        new MachineLearningClient(client).deleteDatafeed(new DeleteDatafeedAction.Request(datafeedId),
listener);
        listener.actionGet();
    }
  }

  private static OperationMode randomInvalidLicenseType() {
    return randomFrom(License.OperationMode.GOLD, License.OperationMode.STANDARD,
License.OperationMode.BASIC);
  }

  private static OperationMode randomValidLicenseType() {
    return randomFrom(License.OperationMode.TRIAL, License.OperationMode.PLATINUM);
  }

  private static OperationMode randomLicenseType() {
    return randomFrom(License.OperationMode.values());
  }

  private static void assertMLAllowed(boolean expected) {
    for (XPackLicenseState licenseState : internalCluster().getInstances(XPackLicenseState.class)) {
        assertEquals(licenseState.isMachineLearningAllowed(), expected);
    }
  }

  public static void disableLicensing() {
    disableLicensing(randomValidLicenseType());
  }

  public static void disableLicensing(License.OperationMode operationMode) {
```

```
        for (XPackLicenseState licenseState : internalCluster().getInstances(XPackLicenseState.class)) {
            licenseState.update(operationMode, false, null);
        }
    }

    public static void enableLicensing() {
        enableLicensing(randomValidLicenseType());
    }

    public static void enableLicensing(License.OperationMode operationMode) {
        for (XPackLicenseState licenseState : internalCluster().getInstances(XPackLicenseState.class)) {
            licenseState.update(operationMode, true, null);
        }
    }
}
```

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2012 International Business Machines Corporation and others

All rights reserved.

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

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

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

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

```
/*
*                         Apache License
*                   Version 2.0, January 2004
*                http://www.apache.org/licenses/
*
*  TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
*
*  1. Definitions.
*
*    "License" shall mean the terms and conditions for use, reproduction,
*    and distribution as defined by Sections 1 through 9 of this document.
*
*    "Licensor" shall mean the copyright owner or entity authorized by
*    the copyright owner that is granting the License.
*
*    "Legal Entity" shall mean the union of the acting entity and all
*    other entities that control, are controlled by, or are under common
*    control with that entity. For the purposes of this definition,
*    "control" means (i) the power, direct or indirect, to cause the
*    direction or management of such entity, whether by contract or
*    otherwise, or (ii) ownership of fifty percent (50%) or more of the
*    outstanding shares, or (iii) beneficial ownership of such entity.
*
*    "You" (or "Your") shall mean an individual or Legal Entity
*    exercising permissions granted by this License.
*
*    "Source" form shall mean the preferred form for making modifications,
*    including but not limited to software source code, documentation
*    source, and configuration files.
*
*    "Object" form shall mean any form resulting from mechanical
*    transformation or translation of a Source form, including but
*    not limited to compiled object code, generated documentation,
*    and conversions to other media types.
*
*    "Work" shall mean the work of authorship, whether in Source or
*    Object form, made available under the License, as indicated by a
*    copyright notice that is included in or attached to the work
*    (an example is provided in the Appendix below).
*
*    "Derivative Works" shall mean any work, whether in Source or Object
*    form, that is based on (or derived from) the Work and for which the
*    editorial revisions, annotations, elaborations, or other modifications
*    represent, as a whole, an original work of authorship. For the purposes
*    of this License, Derivative Works shall not include works that remain
*    separable from, or merely link (or bind by name) to the interfaces of,
*    the Work and Derivative Works thereof.
*
```

```
*     "Contribution" shall mean any work of authorship, including
*     the original version of the Work and any modifications or additions
*     to that Work or Derivative Works thereof, that is intentionally
*     submitted to Licensor for inclusion in the Work by the copyright owner
*     or by an individual or Legal Entity authorized to submit on behalf of
*     the copyright owner. For the purposes of this definition, "submitted"
*     means any form of electronic, verbal, or written communication sent
*     to the Licensor or its representatives, including but not limited to
*     communication on electronic mailing lists, source code control systems,
*     and issue tracking systems that are managed by, or on behalf of, the
*     Licensor for the purpose of discussing and improving the Work, but
*     excluding communication that is conspicuously marked or otherwise
*     designated in writing by the copyright owner as "Not a Contribution."
*
*     "Contributor" shall mean Licensor and any individual or Legal Entity
*     on behalf of whom a Contribution has been received by Licensor and
*     subsequently incorporated within the Work.
*
*   2. Grant of Copyright License. Subject to the terms and conditions of
*     this License, each Contributor hereby grants to You a perpetual,
*     worldwide, non-exclusive, no-charge, royalty-free, irrevocable
*     copyright license to reproduce, prepare Derivative Works of,
*     publicly display, publicly perform, sublicense, and distribute the
*     Work and such Derivative Works in Source or Object form.
*
*   3. Grant of Patent License. Subject to the terms and conditions of
*     this License, each Contributor hereby grants to You a perpetual,
*     worldwide, non-exclusive, no-charge, royalty-free, irrevocable
*     (except as stated in this section) patent license to make, have made,
*     use, offer to sell, sell, import, and otherwise transfer the Work,
*     where such license applies only to those patent claims licensable
*     by such Contributor that are necessarily infringed by their
*     Contribution(s) alone or by combination of their Contribution(s)
*     with the Work to which such Contribution(s) was submitted. If You
*     institute patent litigation against any entity (including a
*     cross-claim or counterclaim in a lawsuit) alleging that the Work
*     or a Contribution incorporated within the Work constitutes direct
*     or contributory patent infringement, then any patent licenses
*     granted to You under this License for that Work shall terminate
*     as of the date such litigation is filed.
*
*   4. Redistribution. You may reproduce and distribute copies of the
*     Work or Derivative Works thereof in any medium, with or without
*     modifications, and in Source or Object form, provided that You
*     meet the following conditions:
*
*     (a) You must give any other recipients of the Work or
*         Derivative Works a copy of this License; and
```

```
*
*     (b) You must cause any modified files to carry prominent notices
*         stating that You changed the files; and
*
*     (c) You must retain, in the Source form of any Derivative Works
*         that You distribute, all copyright, patent, trademark, and
*         attribution notices from the Source form of the Work,
*         excluding those notices that do not pertain to any part of
*         the Derivative Works; and
*
*     (d) If the Work includes a "NOTICE" text file as part of its
*         distribution, then any Derivative Works that You distribute must
*         include a readable copy of the attribution notices contained
*         within such NOTICE file, excluding those notices that do not
*         pertain to any part of the Derivative Works, in at least one
*         of the following places: within a NOTICE text file distributed
*         as part of the Derivative Works; within the Source form or
*         documentation, if provided along with the Derivative Works; or,
*         within a display generated by the Derivative Works, if and
*         wherever such third-party notices normally appear. The contents
*         of the NOTICE file are for informational purposes only and
*         do not modify the License. You may add Your own attribution
*         notices within Derivative Works that You distribute, alongside
*         or as an addendum to the NOTICE text from the Work, provided
*         that such additional attribution notices cannot be construed
*         as modifying the License.
*
*     You may add Your own copyright statement to Your modifications and
*     may provide additional or different license terms and conditions
*     for use, reproduction, or distribution of Your modifications, or
*     for any such Derivative Works as a whole, provided Your use,
*     reproduction, and distribution of the Work otherwise complies with
*     the conditions stated in this License.
*
*   5. Submission of Contributions. Unless You explicitly state otherwise,
*     any Contribution intentionally submitted for inclusion in the Work
*     by You to the Licensor shall be under the terms and conditions of
*     this License, without any additional terms or conditions.
*     Notwithstanding the above, nothing herein shall supersede or modify
*     the terms of any separate license agreement you may have executed
*     with Licensor regarding such Contributions.
*
*   6. Trademarks. This License does not grant permission to use the trade
*     names, trademarks, service marks, or product names of the Licensor,
*     except as required for reasonable and customary use in describing the
*     origin of the Work and reproducing the content of the NOTICE file.
*
*   7. Disclaimer of Warranty. Unless required by applicable law or
```

```
*     agreed to in writing, Licensor provides the Work (and each
*     Contributor provides its Contributions) on an "AS IS" BASIS,
*     WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
*     implied, including, without limitation, any warranties or conditions
*     of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
*     PARTICULAR PURPOSE. You are solely responsible for determining the
*     appropriateness of using or redistributing the Work and assume any
*     risks associated with Your exercise of permissions under this License.
*
*  8. Limitation of Liability. In no event and under no legal theory,
*     whether in tort (including negligence), contract, or otherwise,
*     unless required by applicable law (such as deliberate and grossly
*     negligent acts) or agreed to in writing, shall any Contributor be
*     liable to You for damages, including any direct, indirect, special,
*     incidental, or consequential damages of any character arising as a
*     result of this License or out of the use or inability to use the
*     Work (including but not limited to damages for loss of goodwill,
*     work stoppage, computer failure or malfunction, or any and all
*     other commercial damages or losses), even if such Contributor
*     has been advised of the possibility of such damages.
*
*  9. Accepting Warranty or Additional Liability. While redistributing
*     the Work or Derivative Works thereof, You may choose to offer,
*     and charge a fee for, acceptance of support, warranty, indemnity,
*     or other liability obligations and/or rights consistent with this
*     License. However, in accepting such obligations, You may act only
*     on Your own behalf and on Your sole responsibility, not on behalf
*     of any other Contributor, and only if You agree to indemnify,
*     defend, and hold each Contributor harmless for any liability
*     incurred by, or claims asserted against, such Contributor by reason
*     of your accepting any such warranty or additional liability.
*
*  END OF TERMS AND CONDITIONS
*
*  APPENDIX: How to apply the Apache License to your work.
*
*     To apply the Apache License to your work, attach the following
*     boilerplate notice, with the fields enclosed by brackets "[]"
*     replaced with your own identifying information. (Don't include
*     the brackets!)  The text should be enclosed in the appropriate
*     comment syntax for the file format. We also recommend that a
*     file or class name and description of purpose be included on the
*     same "printed page" as the copyright notice for easier
*     identification within third-party archives.
*
*  Copyright 2007 Kasper B. Graversen
*
*  Licensed under the Apache License, Version 2.0 (the "License");
```

```
*   you may not use this file except in compliance with the License.
*   You may obtain a copy of the License at
*
*       http://www.apache.org/licenses/LICENSE-2.0
*
*   Unless required by applicable law or agreed to in writing, software
*   distributed under the License is distributed on an "AS IS" BASIS,
*   WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
*   See the License for the specific language governing permissions and
*   limitations under the License.
*/
ICU4J, (under lucene/analysis/icu) is licensed under an MIT style license
(modules/analysis/icu/lib/icu4j-LICENSE-BSD_LIKE.txt) and Copyright (c) 1995-2012
International Business Machines Corporation and others
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.security.authz.permission;

import org.elasticsearch.ElasticsearchParseException;
import org.elasticsearch.common.bytes.BytesArray;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.xpack.core.security.authz.RoleDescriptor;
import org.elasticsearch.xpack.core.security.authz.permission.FieldPermissions;
import org.elasticsearch.xpack.core.security.authz.permission.FieldPermissionsDefinition;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.concurrent.CountDownLatch;
import java.util.concurrent.atomic.AtomicReferenceArray;

import static org.hamcrest.Matchers.containsString;

public class FieldPermissionsTests extends ESTestCase {

  public void testParseFieldPermissions() throws Exception {
    String q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
        "\"field_security\": {" +
        "\"grant\": [\"f1\", \"f2\", \"f3\", \"f4\"]," +
        "\"except\": [\"f3\",\"f4\"]" +
        "}}]}";
    RoleDescriptor rd =
        RoleDescriptor.parse("test", new BytesArray(q), false, XContentType.JSON);
```

```
assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(),
        new String[] { "f1", "f2", "f3", "f4" });
assertArrayEquals(rd.getIndicesPrivileges()[0].getDeniedFields(),
        new String[] { "f3", "f4" });

q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
        "\"field_security\": {" +
        "\"except\": [\"f3\",\"f4\"]," +
        "\"grant\": [\"f1\", \"f2\", \"f3\", \"f4\"]" +
        "}}]}";
rd = RoleDescriptor.parse("test", new BytesArray(q), false, XContentType.JSON);
assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(),
        new String[] { "f1", "f2", "f3", "f4" });
assertArrayEquals(rd.getIndicesPrivileges()[0].getDeniedFields(),
        new String[] { "f3", "f4" });

q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
        "\"field_security\": {" +
        "\"grant\": [\"f1\", \"f2\"]" +
        "}}]}";
rd = RoleDescriptor.parse("test", new BytesArray(q), false, XContentType.JSON);
assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(),
        new String[] { "f1", "f2" });
assertNull(rd.getIndicesPrivileges()[0].getDeniedFields());

q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
        "\"field_security\": {" +
        "\"grant\": []" +
        "}}]}";
rd = RoleDescriptor.parse("test", new BytesArray(q), false, XContentType.JSON);
assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(), new String[] {});
assertNull(rd.getIndicesPrivileges()[0].getDeniedFields());

q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
        "\"field_security\": {" +
        "\"except\": []," +
        "\"grant\": []" +
        "}}]}";
rd = RoleDescriptor.parse("test", new BytesArray(q), false, XContentType.JSON);
assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(), new String[] {});
assertArrayEquals(rd.getIndicesPrivileges()[0].getDeniedFields(), new String[] {});

final String exceptWithoutGrant = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\":" +
        " [\"p3\"], \"field_security\": {" +
        "\"except\": [\"f1\"]" +
        "}}]}";
ElasticsearchParseException e = expectThrows(ElasticsearchParseException.class,
        () -> RoleDescriptor.parse("test", new BytesArray(exceptWithoutGrant), false,
```

```
            XContentType.JSON));
assertThat(e.getDetailedMessage(),
        containsString("failed to parse indices privileges for role [test]. field_security"
            + " requires grant if except is given"));

final String grantNull = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"]," +
        " \"field_security\": {" +
        "\"grant\": null" +
        "}}]}";
e = expectThrows(ElasticsearchParseException.class,
        () -> RoleDescriptor.parse("test", new BytesArray(grantNull), false,
            XContentType.JSON));
assertThat(e.getDetailedMessage(), containsString("failed to parse indices privileges for" +
        " role [test]. grant must not be null."));

final String exceptNull = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": " +
        "[\"p3\"], \"field_security\": {" +
        "\"grant\": [\"*\"]," +
        "\"except\": null" +
        "}}]}";
e = expectThrows(ElasticsearchParseException.class,
        () -> RoleDescriptor.parse("test", new BytesArray(exceptNull), false,
            XContentType.JSON));
assertThat(e.getDetailedMessage(),
        containsString("failed to parse indices privileges for role [test]. except must" +
            " not be null."));

final String exceptGrantNull = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": " +
        "[\"p3\"], \"field_security\": {" +
        "\"grant\": null," +
        "\"except\": null" +
        "}}]}";
e = expectThrows(ElasticsearchParseException.class,
        () -> RoleDescriptor.parse("test", new BytesArray(exceptGrantNull), false,
            XContentType.JSON));
assertThat(e.getDetailedMessage(), containsString("failed to parse indices privileges " +
        "for role [test]. grant must not be null."));

final String bothFieldsMissing = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": " +
        "[\"p3\"], \"field_security\": {" +
        "}}]}";
e = expectThrows(ElasticsearchParseException.class,
        () -> RoleDescriptor.parse("test", new BytesArray(bothFieldsMissing), false,
            XContentType.JSON));
assertThat(e.getDetailedMessage(), containsString("failed to parse indices privileges " +
        "for role [test]. \"field_security\" must not be empty."));

// try with two indices and mix order a little
```

```
    q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
            "\"field_security\": {" +
            "\"grant\": []" +
            "}}," +
            "{\"names\": \"idx3\",\n" +
            " \"field_security\": {\n" +
            " \"grant\": [\"*\"], \n" +
            " \"except\": [\"f2\"]}," +
            "\"privileges\": [\"p3\"]}]}";
    rd = RoleDescriptor.parse("test", new BytesArray(q), false, XContentType.JSON);
    assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(), new String[] {});
    assertNull(rd.getIndicesPrivileges()[0].getDeniedFields());
    assertArrayEquals(rd.getIndicesPrivileges()[1].getGrantedFields(), new String[] {"*"});
    assertArrayEquals(rd.getIndicesPrivileges()[1].getDeniedFields(), new String[] {"f2"});
}

// test old syntax for field permissions
public void testBWCFieldPermissions() throws Exception {
    String q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
            "\"fields\": [\"f1\", \"f2\"]" +
            "}]}";
    RoleDescriptor rd = RoleDescriptor.parse("test", new BytesArray(q), true,
            XContentType.JSON);
    assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(),
            new String[]{"f1", "f2"});
    assertNull(rd.getIndicesPrivileges()[0].getDeniedFields());

    final String failingQuery = q;
    ElasticsearchParseException e = expectThrows(ElasticsearchParseException.class,
            () -> RoleDescriptor.parse("test", new BytesArray(failingQuery), false,
                    XContentType.JSON));
    assertThat(e.getDetailedMessage(), containsString("[\"fields\": [...]] format has " +
            "changed for field permissions in role [test]" +
            ", use [\"field_security\": {\"grant\":[...],\"except\":[...]}] instead"));

    q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
            "\"fields\": []" +
            "}]}";
    rd = RoleDescriptor.parse("test", new BytesArray(q), true, XContentType.JSON);
    assertArrayEquals(rd.getIndicesPrivileges()[0].getGrantedFields(), new String[]{});
    assertNull(rd.getIndicesPrivileges()[0].getDeniedFields());
    final String failingQuery2 = q;
    e = expectThrows(ElasticsearchParseException.class,
            () -> RoleDescriptor.parse("test", new BytesArray(failingQuery2), false,
                    XContentType.JSON));
    assertThat(e.getDetailedMessage(), containsString("[\"fields\": [...]] format has " +
            "changed for field permissions in role [test]" +
            ", use [\"field_security\": {\"grant\":[...],\"except\":[...]}] instead"));
```

```
    q = "{\"indices\": [ {\"names\": \"idx2\", \"privileges\": [\"p3\"], " +
            "\"fields\": null" +
            "}]}";
    rd = RoleDescriptor.parse("test", new BytesArray(q), true, XContentType.JSON);
    assertNull(rd.getIndicesPrivileges()[0].getGrantedFields());
    assertNull(rd.getIndicesPrivileges()[0].getDeniedFields());
    final String failingQuery3 = q;
    e = expectThrows(ElasticsearchParseException.class,
            () -> RoleDescriptor.parse("test", new BytesArray(failingQuery3), false,
                    XContentType.JSON));
    assertThat(e.getDetailedMessage(), containsString("[\"fields\": [...]] format has " +
            "changed for field permissions in role [test]" +
            ", use [\"field_security\": {\"grant\":[...],\"except\":[...]}] instead"));
}

public void testFieldPermissionsHashCodeThreadSafe() throws Exception {
    final int numThreads = scaledRandomIntBetween(4, 16);
    final FieldPermissions fieldPermissions =new FieldPermissions(
            new FieldPermissionsDefinition(new String[] { "*" }, new String[] { "foo" }));
    final CountDownLatch latch = new CountDownLatch(numThreads + 1);
    final AtomicReferenceArray<Integer> hashCodes = new AtomicReferenceArray<>(numThreads);
    List<Thread> threads = new ArrayList<>(numThreads);
    for (int i = 0; i < numThreads; i++) {
        final int threadNum = i;
        threads.add(new Thread(() -> {
            latch.countDown();
            try {
                latch.await();
            } catch (InterruptedException e) {
                Thread.currentThread().interrupt();
            }
            final int hashCode = fieldPermissions.hashCode();
            hashCodes.set(threadNum, hashCode);
        }));
    }

    for (Thread thread : threads) {
        thread.start();
    }
    latch.countDown();
    for (Thread thread : threads) {
        thread.join();
    }

    final int hashCode = fieldPermissions.hashCode();
    for (int i = 0; i < numThreads; i++) {
        assertEquals((Integer) hashCode, hashCodes.get(i));
```

```
    }
  }

  public void testAllFieldIsAutomaticallyExcludedIfNotExplicitlyGranted() throws Exception {
    final FieldPermissions fieldPermissions = new FieldPermissions(
        new FieldPermissionsDefinition(new String[] { "_a*" }, new String[0]));
    assertTrue(fieldPermissions.grantsAccessTo("_animal"));
    assertFalse(fieldPermissions.grantsAccessTo("_all"));
  }

  public void testAllFieldIsNotExcludedIfExplicitlyGranted() throws Exception {
    final String[] grant = { "foo", "bar", "baz", "_all" };
    Collections.shuffle(Arrays.asList(grant), random());

    final FieldPermissions fieldPermissions = new FieldPermissions(
        new FieldPermissionsDefinition(grant, new String[0]));

    assertTrue(fieldPermissions.grantsAccessTo("_all"));
    assertTrue(fieldPermissions.grantsAccessTo("foo"));
    assertTrue(fieldPermissions.grantsAccessTo("bar"));
    assertTrue(fieldPermissions.grantsAccessTo("baz"));
    assertFalse(fieldPermissions.grantsAccessTo(randomAlphaOfLengthBetween(5, 8)));
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.security.authz.permission;

import org.elasticsearch.action.get.GetAction;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.xpack.core.security.authz.permission.Role;
import org.elasticsearch.xpack.core.security.authz.privilege.Privilege;
import org.junit.Before;

import java.util.function.Predicate;

import static org.elasticsearch.xpack.core.security.authz.privilege.IndexPrivilege.MONITOR;
import static org.elasticsearch.xpack.core.security.authz.privilege.IndexPrivilege.READ;
import static org.hamcrest.Matchers.is;
import static org.hamcrest.Matchers.notNullValue;

public class PermissionTests extends ESTestCase {
  private Role permission;

  @Before
```

```
    public void init() {
        Role.Builder builder = Role.builder("test");
        builder.add(MONITOR, "test_*", "/foo.*/");
        builder.add(READ, "baz_*foo", "/fool.*bar/");
        builder.add(MONITOR, "/bar.*/");
        permission = builder.build();
    }

    public void testAllowedIndicesMatcherAction() throws Exception {
        testAllowedIndicesMatcher(permission.indices().allowedIndicesMatcher(GetAction.NAME));
    }

    public void testAllowedIndicesMatcherActionCaching() throws Exception {
        Predicate<String> matcher1 = permission.indices().allowedIndicesMatcher(GetAction.NAME);
        Predicate<String> matcher2 = permission.indices().allowedIndicesMatcher(GetAction.NAME);
        assertThat(matcher1, is(matcher2));
    }

    public void testBuildEmptyRole() {
        Role.Builder permission = Role.builder(new String[] { "some_role" });
        Role role = permission.build();
        assertThat(role, notNullValue());
        assertThat(role.cluster(), notNullValue());
        assertThat(role.indices(), notNullValue());
        assertThat(role.runAs(), notNullValue());
    }

    public void testRunAs() {
        Role permission = Role.builder("some_role")
                .runAs(new Privilege("name", "user1", "run*"))
                .build();
        assertThat(permission.runAs().check("user1"), is(true));
        assertThat(permission.runAs().check("user"), is(false));
        assertThat(permission.runAs().check("run" + randomAlphaOfLengthBetween(1, 10)), is(true));
    }

    // "baz_*foo", "/fool.*bar/"
    private void testAllowedIndicesMatcher(Predicate<String> indicesMatcher) {
        assertThat(indicesMatcher.test("foobar"), is(false));
        assertThat(indicesMatcher.test("fool"), is(false));
        assertThat(indicesMatcher.test("fool2bar"), is(true));
        assertThat(indicesMatcher.test("baz_foo"), is(true));
        assertThat(indicesMatcher.test("barbapapa"), is(false));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
```

```
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchSecurityException;
import org.elasticsearch.action.DocWriteResponse;
import org.elasticsearch.action.admin.cluster.health.ClusterHealthResponse;
import org.elasticsearch.action.admin.cluster.node.info.NodesInfoRequest;
import org.elasticsearch.action.admin.cluster.node.stats.NodesStatsResponse;
import org.elasticsearch.action.admin.cluster.stats.ClusterStatsIndices;
import org.elasticsearch.action.admin.cluster.stats.ClusterStatsResponse;
import org.elasticsearch.action.admin.indices.stats.IndicesStatsResponse;
import org.elasticsearch.action.index.IndexResponse;
import org.elasticsearch.client.Client;
import org.elasticsearch.client.Request;
import org.elasticsearch.client.RequestOptions;
import org.elasticsearch.client.Response;
import org.elasticsearch.client.ResponseException;
import org.elasticsearch.client.transport.NoNodeAvailableException;
import org.elasticsearch.client.transport.TransportClient;
import org.elasticsearch.cluster.routing.ShardRoutingState;
import org.elasticsearch.common.network.NetworkModule;
import org.elasticsearch.common.settings.SecureString;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.util.concurrent.ThreadContext;
import org.elasticsearch.discovery.DiscoveryModule;
import org.elasticsearch.node.MockNode;
import org.elasticsearch.node.Node;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.rest.RestStatus;
import org.elasticsearch.test.SecurityIntegTestCase;
import org.elasticsearch.test.SecuritySettingsSource;
import org.elasticsearch.test.SecuritySettingsSourceField;
import org.elasticsearch.test.discovery.TestZenDiscovery;
import org.elasticsearch.test.junit.annotations.TestLogging;
import org.elasticsearch.transport.Netty4Plugin;
import org.elasticsearch.transport.Transport;
import org.elasticsearch.xpack.core.TestXPackTransportClient;
import org.elasticsearch.xpack.core.XPackField;
import org.elasticsearch.xpack.core.security.SecurityField;
import org.elasticsearch.xpack.core.security.action.user.PutUserResponse;
import org.elasticsearch.xpack.core.security.authc.support.Hasher;
import org.elasticsearch.xpack.core.security.authc.support.UsernamePasswordToken;
import org.elasticsearch.xpack.core.security.client.SecurityClient;
import org.elasticsearch.xpack.security.LocalStateSecurity;
import org.junit.After;
import org.junit.Before;
```

```
import java.nio.file.Files;
import java.nio.file.Path;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collection;
import java.util.List;
import java.util.stream.Collectors;

import static org.elasticsearch.common.xcontent.XContentFactory.jsonBuilder;
import static
org.elasticsearch.discovery.zen.SettingsBasedHostsProvider.DISCOVERY_ZEN_PING_UNICAST_HOSTS_SETT
ING;
import static org.elasticsearch.test.hamcrest.ElasticsearchAssertions.assertNoFailures;
import static org.hamcrest.Matchers.containsString;
import static org.hamcrest.Matchers.greaterThanOrEqualTo;
import static org.hamcrest.Matchers.hasItem;
import static org.hamcrest.Matchers.is;
import static org.hamcrest.Matchers.notNullValue;

@TestLogging("org.elasticsearch.cluster.service:TRACE,org.elasticsearch.discovery.zen:TRACE,org.elasticsearch.
action.search:TRACE," +
  "org.elasticsearch.search:TRACE")
public class LicensingTests extends SecurityIntegTestCase {
  public static final String ROLES =
        SecuritySettingsSource.TEST_ROLE + ":\n" +
              "  cluster: [ all ]\n" +
              "  indices:\n" +
              "    - names: '*'\n" +
              "      privileges: [manage]\n" +
              "    - names: '/.*/'\n" +
              "      privileges: [write]\n" +
              "    - names: 'test'\n" +
              "      privileges: [read]\n" +
              "    - names: 'test1'\n" +
              "      privileges: [read]\n" +
              "\n" +
              "role_a:\n" +
              "  indices:\n" +
              "    - names: 'a'\n" +
              "      privileges: [all]\n" +
              "\n" +
              "role_b:\n" +
              "  indices:\n" +
              "    - names: 'b'\n" +
              "      privileges: [all]\n";

  public static final String USERS_ROLES =
        SecuritySettingsSource.CONFIG_STANDARD_USER_ROLES +
```

```
            "role_a:user_a,user_b\n" +
            "role_b:user_b\n";

@Override
protected String configRoles() {
    return ROLES;
}

@Override
protected String configUsers() {
    return SecuritySettingsSource.CONFIG_STANDARD_USER +
        "user_a:{plain}passwd\n" +
        "user_b:{plain}passwd\n";
}

@Override
protected String configUsersRoles() {
    return USERS_ROLES;
}

@Override
public Settings nodeSettings(int nodeOrdinal) {
    return Settings.builder().put(super.nodeSettings(nodeOrdinal))
            .put(NetworkModule.HTTP_ENABLED.getKey(), true)
        .put(TestZenDiscovery.USE_MOCK_PINGS.getKey(), false)
            .build();
}

@Override
protected Collection<Class<? extends Plugin>> nodePlugins() {
    ArrayList<Class<? extends Plugin>> plugins = new ArrayList<>(super.nodePlugins());
    plugins.add(Netty4Plugin.class); // for http
    return plugins;
}

@Override
protected int maxNumberOfNodes() {
    return super.maxNumberOfNodes() + 1;
}

@Before
public void resetLicensing() {
    enableLicensing();
}

@After
public void cleanupSecurityIndex() {
    deleteSecurityIndex();
```

```
    }

    public void testEnableDisableBehaviour() throws Exception {
        IndexResponse indexResponse = index("test", "type", jsonBuilder()
                .startObject()
                .field("name", "value")
                .endObject());
        assertEquals(DocWriteResponse.Result.CREATED, indexResponse.getResult());


        indexResponse = index("test1", "type", jsonBuilder()
                .startObject()
                .field("name", "value1")
                .endObject());
        assertEquals(DocWriteResponse.Result.CREATED, indexResponse.getResult());

        refresh();
        // wait for all replicas to be started (to make sure that there are no more cluster state updates when we disable
licensing)
        assertBusy(() -> assertTrue(client().admin().cluster().prepareState().get().getState().routingTable()
                .shardsWithState(ShardRoutingState.INITIALIZING).isEmpty()));

        Client client = internalCluster().transportClient();

        disableLicensing();

        assertElasticsearchSecurityException(() -> client.admin().indices().prepareStats().get());
        assertElasticsearchSecurityException(() -> client.admin().cluster().prepareClusterStats().get());
        assertElasticsearchSecurityException(() -> client.admin().cluster().prepareHealth().get());
        assertElasticsearchSecurityException(() -> client.admin().cluster().prepareNodesStats().get());

        enableLicensing(randomFrom(License.OperationMode.values()));

        IndicesStatsResponse indicesStatsResponse = client.admin().indices().prepareStats().get();
        assertNoFailures(indicesStatsResponse);

        ClusterStatsResponse clusterStatsNodeResponse = client.admin().cluster().prepareClusterStats().get();
        assertThat(clusterStatsNodeResponse, notNullValue());
        ClusterStatsIndices indices = clusterStatsNodeResponse.getIndicesStats();
        assertThat(indices, notNullValue());
        assertThat(indices.getIndexCount(), greaterThanOrEqualTo(2));

        ClusterHealthResponse clusterIndexHealth = client.admin().cluster().prepareHealth().get();
        assertThat(clusterIndexHealth, notNullValue());

        NodesStatsResponse nodeStats = client.admin().cluster().prepareNodesStats().get();
        assertThat(nodeStats, notNullValue());
    }
```

```
  public void testRestAuthenticationByLicenseType() throws Exception {
    Response unauthorizedRootResponse = getRestClient().performRequest(new Request("GET", "/"));
    // the default of the licensing tests is basic
    assertThat(unauthorizedRootResponse.getStatusLine().getStatusCode(), is(200));
    ResponseException e = expectThrows(ResponseException.class,
        () -> getRestClient().performRequest(new Request("GET", "/_xpack/security/_authenticate")));
    assertThat(e.getResponse().getStatusLine().getStatusCode(), is(403));

    // generate a new license with a mode that enables auth
    License.OperationMode mode = randomFrom(License.OperationMode.GOLD,
License.OperationMode.TRIAL,
        License.OperationMode.PLATINUM, License.OperationMode.STANDARD);
    enableLicensing(mode);
    e = expectThrows(ResponseException.class, () -> getRestClient().performRequest(new Request("GET", "/")));
    assertThat(e.getResponse().getStatusLine().getStatusCode(), is(401));
    e = expectThrows(ResponseException.class,
      () -> getRestClient().performRequest(new Request("GET", "/_xpack/security/_authenticate")));
    assertThat(e.getResponse().getStatusLine().getStatusCode(), is(401));

    RequestOptions.Builder optionsBuilder = RequestOptions.DEFAULT.toBuilder();
    optionsBuilder.addHeader("Authorization",
UsernamePasswordToken.basicAuthHeaderValue(SecuritySettingsSource.TEST_USER_NAME,
        new SecureString(SecuritySettingsSourceField.TEST_PASSWORD.toCharArray())));
    RequestOptions options = optionsBuilder.build();

    Request rootRequest = new Request("GET", "/");
    rootRequest.setOptions(options);
    Response authorizedRootResponse = getRestClient().performRequest(rootRequest);
    assertThat(authorizedRootResponse.getStatusLine().getStatusCode(), is(200));
    Request authenticateRequest = new Request("GET", "/_xpack/security/_authenticate");
    authenticateRequest.setOptions(options);
    Response authorizedAuthenticateResponse = getRestClient().performRequest(authenticateRequest);
    assertThat(authorizedAuthenticateResponse.getStatusLine().getStatusCode(), is(200));
  }

  public void testSecurityActionsByLicenseType() throws Exception {
    // security actions should not work!
    Settings settings = internalCluster().transportClient().settings();
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateSecurity.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
      new SecurityClient(client).preparePutUser("john", "password".toCharArray(), Hasher.BCRYPT).get();
      fail("security actions should not be enabled!");
    } catch (ElasticsearchSecurityException e) {
      assertThat(e.status(), is(RestStatus.FORBIDDEN));
      assertThat(e.getMessage(), containsString("non-compliant"));
    }
```

```
    // enable a license that enables security
    License.OperationMode mode = randomFrom(License.OperationMode.GOLD,
License.OperationMode.TRIAL,
            License.OperationMode.PLATINUM, License.OperationMode.STANDARD);
    enableLicensing(mode);
    // security actions should work!
    try (TransportClient client = new TestXPackTransportClient(settings, LocalStateSecurity.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        PutUserResponse response = new SecurityClient(client).preparePutUser("john", "password".toCharArray(),
Hasher.BCRYPT).get();
        assertNotNull(response);
    }
  }

  public void testTransportClientAuthenticationByLicenseType() throws Exception {
    Settings.Builder builder = Settings.builder()
            .put(internalCluster().transportClient().settings());
    // remove user info
    builder.remove(SecurityField.USER_SETTING.getKey());
    builder.remove(ThreadContext.PREFIX + "." + UsernamePasswordToken.BASIC_AUTH_HEADER);

    // basic has no auth
    try (TransportClient client = new TestXPackTransportClient(builder.build(), LocalStateSecurity.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        assertGreenClusterState(client);
    }

    // enable a license that enables security
    License.OperationMode mode = randomFrom(License.OperationMode.GOLD,
License.OperationMode.TRIAL,
            License.OperationMode.PLATINUM, License.OperationMode.STANDARD);
    enableLicensing(mode);

    try (TransportClient client = new TestXPackTransportClient(builder.build(), LocalStateSecurity.class)) {
client.addTransportAddress(internalCluster().getDataNodeInstance(Transport.class).boundAddress().publishAddress
());
        client.admin().cluster().prepareHealth().get();
        fail("should not have been able to connect to a node!");
    } catch (NoNodeAvailableException e) {
        // expected
    }
  }

  public void testNodeJoinWithoutSecurityExplicitlyEnabled() throws Exception {
    License.OperationMode mode = randomFrom(License.OperationMode.GOLD,
```

```
License.OperationMode.PLATINUM, License.OperationMode.STANDARD);
        enableLicensing(mode);
        ensureGreen();

        final List<String> unicastHostsList = internalCluster().masterClient().admin().cluster().nodesInfo(new
NodesInfoRequest()).get()
            .getNodes().stream().map(n -> n.getTransport().getAddress().publishAddress().toString()).distinct()
            .collect(Collectors.toList());

        Path home = createTempDir();
        Path conf = home.resolve("config");
        Files.createDirectories(conf);
        Settings nodeSettings = Settings.builder()
            .put(nodeSettings(maxNumberOfNodes() - 1).filter(s -> "xpack.security.enabled".equals(s) == false))
            .put("node.name", "my-test-node")
            .put("network.host", "localhost")
            .put("cluster.name", internalCluster().getClusterName())
            .put("discovery.zen.minimum_master_nodes",
                internalCluster().getInstance(Settings.class).get("discovery.zen.minimum_master_nodes"))
            .put("path.home", home)
            .put(TestZenDiscovery.USE_MOCK_PINGS.getKey(), false)
            .put(DiscoveryModule.DISCOVERY_TYPE_SETTING.getKey(), "test-zen")
            .putList(DiscoveryModule.DISCOVERY_HOSTS_PROVIDER_SETTING.getKey())
            .putList(DISCOVERY_ZEN_PING_UNICAST_HOSTS_SETTING.getKey(), unicastHostsList)
            .build();
        Collection<Class<? extends Plugin>> mockPlugins = Arrays.asList(LocalStateSecurity.class,
TestZenDiscovery.TestPlugin.class);
        try (Node node = new MockNode(nodeSettings, mockPlugins)) {
            node.start();
            ensureStableCluster(cluster().size() + 1);
        }
    }

    private static void assertElasticsearchSecurityException(ThrowingRunnable runnable) {
        ElasticsearchSecurityException ee = expectThrows(ElasticsearchSecurityException.class, runnable);
        assertThat(ee.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA),
hasItem(XPackField.SECURITY));
        assertThat(ee.status(), is(RestStatus.FORBIDDEN));
    }

    public static void disableLicensing() {
        disableLicensing(License.OperationMode.BASIC);
    }

    public static void disableLicensing(License.OperationMode operationMode) {
        for (XPackLicenseState licenseState : internalCluster().getInstances(XPackLicenseState.class)) {
            licenseState.update(operationMode, false, null);
        }
```

```
  }

  public static void enableLicensing() {
    enableLicensing(License.OperationMode.BASIC);
  }

  public static void enableLicensing(License.OperationMode operationMode) {
    for (XPackLicenseState licenseState : internalCluster().getInstances(XPackLicenseState.class)) {
      licenseState.update(operationMode, true, null);
    }
  }
}
```

Copyright (c) 2000-2015 The Legion of the Bouncy Castle Inc. (http://www.bouncycastle.org)

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

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

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF
CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER
DEALINGS IN THE SOFTWARE.
Dropwizard
Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2016 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.
Copyright (c) 2004-2017 QOS.ch
All rights reserved.

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

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

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


```
<!DOCTYPE html>
<html lang="en">
 <head>
   <meta charset="utf-8">
 <link rel="dns-prefetch" href="https://assets-cdn.github.com">
 <link rel="dns-prefetch" href="https://avatars0.githubusercontent.com">
 <link rel="dns-prefetch" href="https://avatars1.githubusercontent.com">
 <link rel="dns-prefetch" href="https://avatars2.githubusercontent.com">
 <link rel="dns-prefetch" href="https://avatars3.githubusercontent.com">
 <link rel="dns-prefetch" href="https://github-cloud.s3.amazonaws.com">
 <link rel="dns-prefetch" href="https://user-images.githubusercontent.com/">


 <link crossorigin="anonymous" href="https://assets-cdn.github.com/assets/frameworks-
2d2d4c150f7000385741c6b992b302689ecd172246c6428904e0813be9bceca6.css" media="all" rel="stylesheet" />
 <link crossorigin="anonymous" href="https://assets-cdn.github.com/assets/github-
0522ae8d3b3bdc841d2f91f90efd5f1fd9040d910905674cd134ced43a6dfea6.css" media="all" rel="stylesheet" />


 <link crossorigin="anonymous" href="https://assets-cdn.github.com/assets/site-
cfab053e93f0e27f4c63d4ff6b7957bd25f711667fe678e747f8a4d88c47b38d.css" media="all" rel="stylesheet" />


 <meta name="viewport" content="width=device-width">

 <title>cryptacular/NOTICE at master  vt-middleware/cryptacular  GitHub</title>
 <link rel="search" type="application/opensearchdescription+xml" href="/opensearch.xml" title="GitHub">
 <link rel="fluid-icon" href="https://github.com/fluidicon.png" title="GitHub">
 <meta property="fb:app_id" content="1401488693436528">
```

<meta content="https://avatars7.githubusercontent.com/u/6122907?v=4&s=400" property="og:image" /><meta content="GitHub" property="og:site_name" /><meta content="object" property="og:type" /><meta content="vt-middleware/cryptacular" property="og:title" /><meta content="https://github.com/vt-middleware/cryptacular" property="og:url" /><meta content="cryptacular - The friendly complement to the BouncyCastle crypto API for Java." property="og:description" />

<link rel="assets" href="https://assets-cdn.github.com/">

<meta name="pjax-timeout" content="1000">

<meta name="request-id" content="E0E4:26F16:12A5AE6:1D11801:596C7978" data-pjax-transient>

<meta name="selected-link" value="repo_source" data-pjax-transient>

<meta name="google-site-verification" content="KT5gs8h0wvaagLKAVWq8bbeNwnZZK1r1XQysX3xurLU">
<meta name="google-site-verification" content="ZzhVyEFwb7w3e0-uOTltm8Jsck2F5StVihD0exw2fsA">
<meta name="google-analytics" content="UA-3769691-2">

<meta content="collector.githubapp.com" name="octolytics-host" /><meta content="github" name="octolytics-app-id" /><meta content="https://collector.githubapp.com/github-external/browser_event" name="octolytics-event-url" /><meta content="E0E4:26F16:12A5AE6:1D11801:596C7978" name="octolytics-dimension-request_id" /><meta content="sea" name="octolytics-dimension-region_edge" /><meta content="iad" name="octolytics-dimension-region_render" />
<meta content="/<user-name>/<repo-name>/blob/show" data-pjax-transient="true" name="analytics-location" />

<meta class="js-ga-set" name="dimension1" content="Logged Out">

<meta name="hostname" content="github.com">
<meta name="user-login" content="">

<meta name="expected-hostname" content="github.com">
<meta name="js-proxy-site-detection-payload" content="N2ZhMjk0NTA4MjI1NmZhYTVlNzM5NzVjZmFkOWY2NGFkNmMxYzcyMGViNzAzZGQxMGMzZmJhZDQ3YWZiZTI0OHx7InJlbW90ZV9hZGRyZXNzIjoiMTEwLjIwLjIyMC4xMzUiLCJyZXF1ZXN0X2lkIjoiRTBFNDoyNkYxNjoxMkE1QUU2OjFEMTE4MDE6NTk2Qzc5NzgiLCJ0aW1lc3RhbXAiOjE1MDAyODEyMDgsImhvc3QiOiJnaXRodWIuY29tIn0=">

<meta name="html-safe-nonce" content="5aa226b80a18dc40894e1d405e4eb31cfca7d616">

<meta http-equiv="x-pjax-version" content="f682644ce1bb9629b9d9d9bedf64801b">

<link href="https://github.com/vt-middleware/cryptacular/commits/master.atom" rel="alternate" title="Recent Commits to cryptacular:master" type="application/atom+xml">

<meta name="description" content="cryptacular - The friendly complement to the BouncyCastle crypto API for Java.">
<meta name="go-import" content="github.com/vt-middleware/cryptacular git https://github.com/vt-middleware/cryptacular.git">

<meta content="6122907" name="octolytics-dimension-user_id" /><meta content="vt-middleware" name="octolytics-dimension-user_login" /><meta content="15714989" name="octolytics-dimension-repository_id" /><meta content="vt-middleware/cryptacular" name="octolytics-dimension-repository_nwo" /><meta content="true" name="octolytics-dimension-repository_public" /><meta content="false" name="octolytics-dimension-repository_is_fork" /><meta content="15714989" name="octolytics-dimension-repository_network_root_id" /><meta content="vt-middleware/cryptacular" name="octolytics-dimension-repository_network_root_nwo" /><meta content="false" name="octolytics-dimension-repository_explore_github_marketplace_ci_cta_shown" />

<link rel="canonical" href="https://github.com/vt-middleware/cryptacular/blob/master/NOTICE" data-pjax-transient>

<meta name="browser-stats-url" content="https://api.github.com/_private/browser/stats">

<meta name="browser-errors-url" content="https://api.github.com/_private/browser/errors">

<link rel="mask-icon" href="https://assets-cdn.github.com/pinned-octocat.svg" color="#000000">
<link rel="icon" type="image/x-icon" href="https://assets-cdn.github.com/favicon.ico">

<meta name="theme-color" content="#1e2327">

</head>

<body class="logged-out env-production page-blob">

Skip to content

```
        <header class="site-header js-details-container Details" role="banner">
 <div class="site-nav-container">
   <a class="header-logo-invertocat" href="https://github.com/" aria-label="Homepage" data-ga-click="(Logged out)
Header, go to homepage, icon:logo-wordmark">
     <svg aria-hidden="true" class="octicon octicon-mark-github" height="32" version="1.1" viewBox="0 0 16 16"
width="32"><path fill-rule="evenodd" d="M8 0C3.58 0 0 3.58 0 8c0 3.54 2.29 6.53 5.47 7.59.4.07.55-.17.55-.38 0-
.19-.01-.82-.01-1.49-2.01.37-2.53-.49-2.69-.94-.09-.23-.48-.94-.82-1.13-.28-.15-.68-.52-.01-.53.63-.01 1.08.58
1.23.82.72 1.21 1.87.87 2.33.66.07-.52.28-.87.51-1.07-1.78-.2-3.64-.89-3.64-3.95 0-.87.31-1.59.82-2.15-.08-.2-.36-
1.02.08-2.12 0 0 .67-.21 2.2.82.64-.18 1.32-.27 2-.27.68 0 1.36.09 2 .27 1.53-1.04 2.2-.82 2.2-.82.44 1.1.16 1.92.08
2.12.51.56.82 1.27.82 2.15 0 3.07-1.87 3.75-3.65 3.95.29.25.54.73.54 1.48 0 1.07-.01 1.93-.01 2.2 0
.21.15.46.55.38A8.013 8.013 0 0 0 16 8c0-4.42-3.58-8-8-8z"/></svg>
   </a>

   <button class="btn-link float-right site-header-toggle js-details-target" type="button" aria-label="Toggle
navigation" aria-expanded="false">
     <svg aria-hidden="true" class="octicon octicon-three-bars" height="24" version="1.1" viewBox="0 0 12 16"
width="18"><path fill-rule="evenodd" d="M11.41 9H.59C0 9 0 8.59 0 8c0-.59 0-1 .59-1H11.4c.59 0 .59.41.59 1 0
.59 0 1-.59 1h.01zm0-4H.59C0 5 0 4.59 0 4c0-.59 0-1 .59-1H11.4c.59 0 .59.41.59 1 0 .59 0 1-.59 1h.01zM.59
11H11.4c.59 0 .59.41.59 1 0 .59 0 1-.59 1H.59C0 13 0 12.59 0 12c0-.59 0-1 .59-1z"/></svg>
   </button>

   <div class="site-header-menu">
     <nav class="site-header-nav">
       <a href="/features" class="js-selected-navigation-item nav-item" data-ga-click="Header, click, Nav menu -
item:features" data-selected-links="/features /features/code-review /features/project-management
/features/integrations /features">
        Features
</a>       <a href="/business" class="js-selected-navigation-item nav-item" data-ga-click="Header, click, Nav menu
- item:business" data-selected-links="/business /business/security /business/customers /business">
        Business
</a>       <a href="/explore" class="js-selected-navigation-item nav-item" data-ga-click="Header, click, Nav menu -
item:explore" data-selected-links="/explore /trending /trending/developers /stars /integrations
/integrations/feature/code /integrations/feature/collaborate /integrations/feature/ship showcases showcases_search
showcases_landing /explore">
        Explore
</a>           <a href="/marketplace" class="js-selected-navigation-item nav-item" data-ga-click="Header, click, Nav
menu - item:marketplace" data-selected-links=" /marketplace">
            Marketplace
</a>       <a href="/pricing" class="js-selected-navigation-item nav-item" data-ga-click="Header, click, Nav menu -
item:pricing" data-selected-links="/pricing /pricing/developer /pricing/team /pricing/business-hosted
/pricing/business-enterprise /pricing">
        Pricing
</a>     </nav>
```

```
    <div class="site-header-actions">
        <div class="header-search scoped-search site-scoped-search js-site-search" role="search">
 <!-- "" ` --><!-- </textarea></xmp> --></option></form><form accept-charset="UTF-8" action="/vt-
middleware/cryptacular/search" class="js-site-search-form" data-scoped-search-url="/vt-
middleware/cryptacular/search" data-unscoped-search-url="/search" method="get"><div
style="margin:0;padding:0;display:inline"><input name="utf8" type="hidden" value="&#x2713;" /></div>
   <label class="form-control header-search-wrapper js-chromeless-input-container">
       <a href="/vt-middleware/cryptacular/blob/master/NOTICE" class="header-search-scope no-underline">This
repository</a>
     <input type="text"
       class="form-control header-search-input js-site-search-focus js-site-search-field is-clearable"
       data-hotkey="s"
       name="q"
       value=""
       placeholder="Search"
       aria-label="Search this repository"
       data-unscoped-placeholder="Search GitHub"
       data-scoped-placeholder="Search"
       autocapitalize="off">
       <input type="hidden" class="js-site-search-type-field" name="type" >
   </label>
</form></div>


        <a class="text-bold site-header-link" href="/login?return_to=%2Fvt-
middleware%2Fcryptacular%2Fblob%2Fmaster%2FNOTICE" data-ga-click="(Logged out) Header, clicked Sign
in, text:sign-in">Sign in</a>
          <span class="text-gray">or</span>
          <a class="text-bold site-header-link" href="/join?source=header-repo" data-ga-click="(Logged out) Header,
clicked Sign up, text:sign-up">Sign up</a>
     </div>
   </div>
 </div>
</header>


 </div>

 <div id="start-of-content" class="show-on-focus"></div>

   <div id="js-flash-container">
</div>



 <div role="main">
       <div itemscope itemtype="http://schema.org/SoftwareSourceCode">
   <div id="js-repo-pjax-container" data-pjax-container>
```

```
  <div class="pagehead repohead instapaper_ignore readability-menu experiment-repo-nav">
    <div class="container repohead-details-container">

      <ul class="pagehead-actions">
 <li>
    <a href="/login?return_to=%2Fvt-middleware%2Fcryptacular"
  class="btn btn-sm btn-with-count tooltipped tooltipped-n"
  aria-label="You must be signed in to watch a repository" rel="nofollow">
  <svg aria-hidden="true" class="octicon octicon-eye" height="16" version="1.1" viewBox="0 0 16 16"
width="16"><path fill-rule="evenodd" d="M8.06 2C3 2 0 8 0 8s3 6 8.06 6C13 14 16 8 16 8s-3-6-7.94-6zM8 12c-
2.2 0-4-1.78-4-4 0-2.2 1.8-4 4-4 2.22 0 4 1.8 4 4 0 2.22-1.78 4-4 4zm2-4c0 1.11-.89 2-2 2-1.11 0-2-.89-2-2 0-
1.11.89-2 2-2 1.11 0 2 .89 2 2z"/></svg>
  Watch
 </a>
 <a class="social-count" href="/vt-middleware/cryptacular/watchers"
   aria-label="9 users are watching this repository">
  9
 </a>

 </li>

 <li>
    <a href="/login?return_to=%2Fvt-middleware%2Fcryptacular"
  class="btn btn-sm btn-with-count tooltipped tooltipped-n"
  aria-label="You must be signed in to star a repository" rel="nofollow">
  <svg aria-hidden="true" class="octicon octicon-star" height="16" version="1.1" viewBox="0 0 14 16"
width="14"><path fill-rule="evenodd" d="M14 6l-4.9-.64L7 1 4.9 5.36 0 6l3.6 3.26L2.67 14 7 11.67 11.33 14l-.93-
4.74z"/></svg>
  Star
 </a>

  <a class="social-count js-social-count" href="/vt-middleware/cryptacular/stargazers"
    aria-label="15 users starred this repository">
    15
  </a>

 </li>

 <li>
    <a href="/login?return_to=%2Fvt-middleware%2Fcryptacular"
      class="btn btn-sm btn-with-count tooltipped tooltipped-n"
```

aria-label="You must be signed in to fork a repository" rel="nofollow">
<svg aria-hidden="true" class="octicon octicon-repo-forked" height="16" version="1.1" viewBox="0 0 10 16" width="10"><path fill-rule="evenodd" d="M8 1a1.993 1.993 0 0 0-1 3.72V6L5 8 3 6V4.72A1.993 1.993 0 0 0 2 1a1.993 1.993 0 0 0-1 3.72V6.5l3 3v1.78A1.993 1.993 0 0 0 5 15a1.993 1.993 0 0 0 1-3.72V9.5l3-3V4.72A1.993 1.993 0 0 0 8 1zM2 4.2C1.34 4.2.8 3.65.8 3c0-.65.55-1.2 1.2-1.2.65 0 1.2.55 1.2 1.2 0 .65-.55 1.2-1.2 1.2zm3 10c-.66 0-1.2-.55-1.2-1.2 0-.65.55-1.2 1.2-1.2.65 0 1.2.55 1.2 1.2 0 .65-.55 1.2-1.2 1.2zm3-10c-.66 0-1.2-.55-1.2-1.2 0-.65.55-1.2 1.2-1.2.65 0 1.2.55 1.2 1.2 0 .65-.55 1.2-1.2 1.2z"/></svg>
Fork

<a href="/vt-middleware/cryptacular/network" class="social-count"
aria-label="1 user forked this repository">
1

<h1 class="public ">
<svg aria-hidden="true" class="octicon octicon-repo" height="16" version="1.1" viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M4 9H3V8h1v1zm0-3H3v1h1V6zm0-2H3v1h1V4zm0-2H3v1h1V2zm8-1v12c0 .55-.45 1-1 1H6v2l-1.5-1.5L3 16v-2H1c-.55 0-1-.45-1-1V1c0-.55.45-1 1-1h10c.55 0 1 .45 1 1zm-1 10H1v2h2v-1h3v1h5v-2zm0-10H2v9h9V1z"/></svg>
vt-middleware<!--
-->/<!--
-->cryptacular

</h1>

<nav class="reponav js-repo-nav js-sidenav-container-pjax"
itemscope
itemtype="http://schema.org/BreadcrumbList"
role="navigation"
data-pjax="#js-repo-pjax-container">

<span itemscope itemtype="http://schema.org/ListItem" itemprop="itemListElement">
<a href="/vt-middleware/cryptacular" class="js-selected-navigation-item selected reponav-item" data-hotkey="g c" data-selected-links="repo_source repo_downloads repo_commits repo_releases repo_tags repo_branches /vt-middleware/cryptacular" itemprop="url">
<svg aria-hidden="true" class="octicon octicon-code" height="16" version="1.1" viewBox="0 0 14 16" width="14"><path fill-rule="evenodd" d="M9.5 3L8 4.5 11.5 8 8 11.5 9.5 13 14 8 9.5 3zm-5 0L0 8l4.5 5L6 11.5 2.5 8 6 4.5 4.5 3z"/></svg>
Code
<meta itemprop="position" content="1">

```
</a>  </span>

  <span itemscope itemtype="http://schema.org/ListItem" itemprop="itemListElement">
    <a href="/vt-middleware/cryptacular/issues" class="js-selected-navigation-item reponav-item" data-hotkey="g i"
data-selected-links="repo_issues repo_labels repo_milestones /vt-middleware/cryptacular/issues" itemprop="url">
      <svg aria-hidden="true" class="octicon octicon-issue-opened" height="16" version="1.1" viewBox="0 0 14 16"
width="14"><path fill-rule="evenodd" d="M7 2.3c3.14 0 5.7 2.56 5.7 5.7s-2.56 5.7-5.7 5.7A5.71 5.71 0 0 1 1.3
8c0-3.14 2.56-5.7 5.7-5.7zM7 1C3.14 1 0 4.14 0 8s3.14 7 7 7 7-3.14 7-7-3.14-7-7-7zm1 3H6v5h2V4zm0
6H6v2h2v-2z"/></svg>
      <span itemprop="name">Issues</span>
      <span class="Counter">4</span>
      <meta itemprop="position" content="2">
</a>    </span>

 <span itemscope itemtype="http://schema.org/ListItem" itemprop="itemListElement">
  <a href="/vt-middleware/cryptacular/pulls" class="js-selected-navigation-item reponav-item" data-hotkey="g p"
data-selected-links="repo_pulls /vt-middleware/cryptacular/pulls" itemprop="url">
    <svg aria-hidden="true" class="octicon octicon-git-pull-request" height="16" version="1.1" viewBox="0 0 12
16" width="12"><path fill-rule="evenodd" d="M11 11.28V5c-.03-.78-.34-1.47-.94-2.06C9.46 2.35 8.78 2.03 8
2H7V0L4 3l3 3V4h1c.27.02.48.11.69.31.21.2.3.42.31.69v6.28A1.993 1.993 0 0 0 10 15a1.993 1.993 0 0 0 1-
3.72zm-1 2.92c-.66 0-1.2-.55-1.2-1.2 0-.65.55-1.2 1.2-1.2.65 0 1.2.55 1.2 1.2 0 .65-.55 1.2-1.2 1.2zM4 3c0-1.11-
.89-2-2-2a1.993 1.993 0 0 0-1 3.72v6.56A1.993 1.993 0 0 0 2 15a1.993 1.993 0 0 0 1-3.72V4.72c.59-.34 1-.98 1-
1.72zm-.8 10c0 .66-.55 1.2-1.2 1.2-.65 0-1.2-.55-1.2-1.2 0-.65.55-1.2 1.2-1.2.65 0 1.2.55 1.2 1.2zM2 4.2C1.34 4.2.8
3.65.8 3c0-.65.55-1.2 1.2-1.2.65 0 1.2.55 1.2 1.2 0 .65-.55 1.2-1.2 1.2z"/></svg>
    <span itemprop="name">Pull requests</span>
    <span class="Counter">1</span>
    <meta itemprop="position" content="3">
</a>  </span>

  <a href="/vt-middleware/cryptacular/projects" class="js-selected-navigation-item reponav-item" data-selected-
links="repo_projects new_repo_project repo_project /vt-middleware/cryptacular/projects">
    <svg aria-hidden="true" class="octicon octicon-project" height="16" version="1.1" viewBox="0 0 15 16"
width="15"><path fill-rule="evenodd" d="M10 12h3V2h-3v10zm-4-2h3V2H6v8zm-4 4h3V2H2v12zm-1
1h13V1H1v14zM14 0H1a1 1 0 0 0-1 1v14a1 1 0 0 0 1 1h13a1 1 0 0 0 1-1V1a1 1 0 0 0-1-1z"/></svg>
    Projects
    <span class="Counter" >0</span>
</a>


  <div class="reponav-dropdown js-menu-container">
    <button type="button" class="btn-link reponav-item reponav-dropdown js-menu-target " data-no-toggle aria-
expanded="false" aria-haspopup="true">
      Insights
      <svg aria-hidden="true" class="octicon octicon-triangle-down v-align-middle text-gray" height="11"
version="1.1" viewBox="0 0 12 16" width="8"><path fill-rule="evenodd" d="M0 5l6 6 6-6z"/></svg>
    </button>
    <div class="dropdown-menu-content js-menu-content">
      <div class="dropdown-menu dropdown-menu-sw">
```

```
        <a class="dropdown-item" href="/vt-middleware/cryptacular/pulse" data-skip-pjax>
          <svg aria-hidden="true" class="octicon octicon-pulse" height="16" version="1.1" viewBox="0 0 14 16"
width="14"><path fill-rule="evenodd" d="M11.5 8L8.8 5.4 6.6 8.5 5.5 1.6 2.38 8H0v2h3.6l.9-1.8.9 5.4L9 8.5l1.6
1.5H14V8z"/></svg>
          Pulse
        </a>
        <a class="dropdown-item" href="/vt-middleware/cryptacular/graphs" data-skip-pjax>
          <svg aria-hidden="true" class="octicon octicon-graph" height="16" version="1.1" viewBox="0 0 16 16"
width="16"><path fill-rule="evenodd" d="M16 14v1H0V0h1v14h15zM5 13H3V8h2v5zm4 0H7V3h2v10zm4 0h-
2V6h2v7z"/></svg>
          Graphs
        </a>
      </div>
    </div>
  </div>
</nav>

    </div>
  </div>

<div class="container new-discussion-timeline experiment-repo-nav">
 <div class="repository-content">


 <a href="/vt-middleware/cryptacular/blob/c26911e3cd28497ce9daa3ce682e09cb2d1d8688/NOTICE" class="d-
none js-permalink-shortcut" data-hotkey="y">Permalink</a>

 <!-- blob contrib key: blob_contributors:v21:f5054b9e46039e0ad937c69e6151b7d4 -->

 <div class="file-navigation js-zeroclipboard-container">

<div class="select-menu branch-select-menu js-menu-container js-select-menu float-left">
 <button class=" btn btn-sm select-menu-button js-menu-target css-truncate" data-hotkey="w"

   type="button" aria-label="Switch branches or tags" aria-expanded="false" aria-haspopup="true">
    <i>Branch:</i>
    <span class="js-select-button css-truncate-target">master</span>
 </button>

 <div class="select-menu-modal-holder js-menu-content js-navigation-container" data-pjax>

   <div class="select-menu-modal">
    <div class="select-menu-header">
      <svg aria-label="Close" class="octicon octicon-x js-menu-close" height="16" role="img" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M7.48 8l3.75 3.75-1.48 1.48L6 9.48l-3.75 3.75-
1.48-1.48L4.52 8 .77 4.25l1.48-1.48L6 6.52l3.75-3.75 1.48 1.48z"/></svg>
      <span class="select-menu-title">Switch branches/tags</span>
    </div>
```

```
    <div class="select-menu-filters">
      <div class="select-menu-text-filter">
        <input type="text" aria-label="Filter branches/tags" id="context-commitish-filter-field" class="form-control
js-filterable-field js-navigation-enable" placeholder="Filter branches/tags">
      </div>
      <div class="select-menu-tabs">
        <ul>
          <li class="select-menu-tab">
            <a href="#" data-tab-filter="branches" data-filter-placeholder="Filter branches/tags" class="js-select-menu-
tab" role="tab">Branches</a>
          </li>
          <li class="select-menu-tab">
            <a href="#" data-tab-filter="tags" data-filter-placeholder="Find a tag" class="js-select-menu-tab"
role="tab">Tags</a>
          </li>
        </ul>
      </div>
    </div>

    <div class="select-menu-list select-menu-tab-bucket js-select-menu-tab-bucket" data-tab-filter="branches"
role="menu">

      <div data-filterable-for="context-commitish-filter-field" data-filterable-type="substring">



          <a class="select-menu-item js-navigation-item js-navigation-open "
            href="/vt-middleware/cryptacular/blob/ISSUE-31+32/NOTICE"
            data-name="ISSUE-31+32"
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
            <span class="select-menu-item-text css-truncate-target js-select-menu-filter-text">
              ISSUE-31+32
            </span>
          </a>
          <a class="select-menu-item js-navigation-item js-navigation-open "
            href="/vt-middleware/cryptacular/blob/gh-pages/NOTICE"
            data-name="gh-pages"
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
            <span class="select-menu-item-text css-truncate-target js-select-menu-filter-text">
              gh-pages
            </span>
          </a>
```

```
          <a class="select-menu-item js-navigation-item js-navigation-open selected"
            href="/vt-middleware/cryptacular/blob/master/NOTICE"
            data-name="master"
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
            <span class="select-menu-item-text css-truncate-target js-select-menu-filter-text">
              master
            </span>
          </a>
          <a class="select-menu-item js-navigation-item js-navigation-open "
            href="/vt-middleware/cryptacular/blob/v1.1/NOTICE"
            data-name="v1.1"
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
            <span class="select-menu-item-text css-truncate-target js-select-menu-filter-text">
              v1.1
            </span>
          </a>
        </div>

        <div class="select-menu-no-results">Nothing to show</div>
      </div>

      <div class="select-menu-list select-menu-tab-bucket js-select-menu-tab-bucket" data-tab-filter="tags">
        <div data-filterable-for="context-commitish-filter-field" data-filterable-type="substring">



          <a class="select-menu-item js-navigation-item js-navigation-open "
            href="/vt-middleware/cryptacular/tree/v1.2.1/NOTICE"
            data-name="v1.2.1"
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
            <span class="select-menu-item-text css-truncate-target" title="v1.2.1">
              v1.2.1
            </span>
          </a>
          <a class="select-menu-item js-navigation-item js-navigation-open "
            href="/vt-middleware/cryptacular/tree/v1.2.0/NOTICE"
            data-name="v1.2.0"
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
```

viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>

v1.2.0

<a class="select-menu-item js-navigation-item js-navigation-open "
href="/vt-middleware/cryptacular/tree/v1.1.2/NOTICE"
data-name="v1.1.2"
data-skip-pjax="true"
rel="nofollow">
<svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>

v1.1.2

<a class="select-menu-item js-navigation-item js-navigation-open "
href="/vt-middleware/cryptacular/tree/v1.1.1/NOTICE"
data-name="v1.1.1"
data-skip-pjax="true"
rel="nofollow">
<svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>

v1.1.1

<a class="select-menu-item js-navigation-item js-navigation-open "
href="/vt-middleware/cryptacular/tree/v1.1.0/NOTICE"
data-name="v1.1.0"
data-skip-pjax="true"
rel="nofollow">
<svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>

v1.1.0

<a class="select-menu-item js-navigation-item js-navigation-open "
href="/vt-middleware/cryptacular/tree/v1.0/NOTICE"
data-name="v1.0"
data-skip-pjax="true"
rel="nofollow">
<svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>

v1.0

```
</a>
<a class="select-menu-item js-navigation-item js-navigation-open "
  href="/vt-middleware/cryptacular/tree/v1.0-RC6/NOTICE"
  data-name="v1.0-RC6"
  data-skip-pjax="true"
  rel="nofollow">
  <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
  <span class="select-menu-item-text css-truncate-target" title="v1.0-RC6">
    v1.0-RC6
  </span>
</a>
<a class="select-menu-item js-navigation-item js-navigation-open "
  href="/vt-middleware/cryptacular/tree/v1.0-RC4/NOTICE"
  data-name="v1.0-RC4"
  data-skip-pjax="true"
  rel="nofollow">
  <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
  <span class="select-menu-item-text css-truncate-target" title="v1.0-RC4">
    v1.0-RC4
  </span>
</a>
<a class="select-menu-item js-navigation-item js-navigation-open "
  href="/vt-middleware/cryptacular/tree/v1.0-RC3/NOTICE"
  data-name="v1.0-RC3"
  data-skip-pjax="true"
  rel="nofollow">
  <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
  <span class="select-menu-item-text css-truncate-target" title="v1.0-RC3">
    v1.0-RC3
  </span>
</a>
<a class="select-menu-item js-navigation-item js-navigation-open "
  href="/vt-middleware/cryptacular/tree/v1.0-RC2/NOTICE"
  data-name="v1.0-RC2"
  data-skip-pjax="true"
  rel="nofollow">
  <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
  <span class="select-menu-item-text css-truncate-target" title="v1.0-RC2">
    v1.0-RC2
  </span>
</a>
<a class="select-menu-item js-navigation-item js-navigation-open "
  href="/vt-middleware/cryptacular/tree/v1.0-RC1/NOTICE"
  data-name="v1.0-RC1"
```

```
            data-skip-pjax="true"
            rel="nofollow">
            <svg aria-hidden="true" class="octicon octicon-check select-menu-item-icon" height="16" version="1.1"
viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M12 5l-8 8-4-4 1.5-1.5L4 10l6.5-6.5z"/></svg>
            <span class="select-menu-item-text css-truncate-target" title="v1.0-RC1">
              v1.0-RC1
            </span>
          </a>
        </div>

        <div class="select-menu-no-results">Nothing to show</div>
      </div>

    </div>
  </div>
</div>

    <div class="BtnGroup float-right">
      <a href="/vt-middleware/cryptacular/find/master"
           class="js-pjax-capture-input btn btn-sm BtnGroup-item"
           data-pjax
           data-hotkey="t">
        Find file
      </a>
      <button aria-label="Copy file path to clipboard" class="js-zeroclipboard btn btn-sm BtnGroup-item tooltipped
tooltipped-s" data-copied-hint="Copied!" type="button">Copy path</button>
    </div>
    <div class="breadcrumb js-zeroclipboard-target">
      <span class="repo-root js-repo-root"><span class="js-path-segment"><a href="/vt-
middleware/cryptacular"><span>cryptacular</span></a></span></span><span class="separator">/</span><strong
class="final-path">NOTICE</strong>
    </div>
  </div>



  <div class="commit-tease">
      <span class="float-right">
        <a class="commit-tease-sha" href="/vt-
middleware/cryptacular/commit/6dd6f199ac3ecc3b4c5aef9e04be3bbe265a30a1" data-pjax>
          6dd6f19
        </a>
        <relative-time datetime="2017-07-06T22:28:36Z">Jul 7, 2017</relative-time>
      </span>
      <div>
        <img alt="@dfish3r" class="avatar" height="20"
src="https://avatars6.githubusercontent.com/u/1051499?v=4&s=40" width="20" />
        <a href="/dfish3r" class="user-mention" rel="contributor">dfish3r</a>
```

```
        <a href="/vt-middleware/cryptacular/commit/6dd6f199ac3ecc3b4c5aef9e04be3bbe265a30a1"
class="message" data-pjax="true" title="Update year in notice.">Update year in notice.</a>
    </div>

  <div class="commit-tease-contributors">
    <button type="button" class="btn-link muted-link contributors-toggle" data-facebox="#blob_contributors_box">
      <strong>1</strong>
       contributor
    </button>

  </div>

  <div id="blob_contributors_box" style="display:none">
    <h2 class="facebox-header" data-facebox-id="facebox-header">Users who have contributed to this file</h2>
    <ul class="facebox-user-list" data-facebox-id="facebox-description">
        <li class="facebox-user-list-item">
          <img alt="@dfish3r" height="24" src="https://avatars4.githubusercontent.com/u/1051499?v=4&s=48"
width="24" />
          <a href="/dfish3r">dfish3r</a>
        </li>
    </ul>
  </div>
 </div>

 <div class="file">
   <div class="file-header">
 <div class="file-actions">

   <div class="BtnGroup">
     <a href="/vt-middleware/cryptacular/raw/master/NOTICE" class="btn btn-sm BtnGroup-item" id="raw-
url">Raw</a>
       <a href="/vt-middleware/cryptacular/blame/master/NOTICE" class="btn btn-sm js-update-url-with-hash
BtnGroup-item" data-hotkey="b">Blame</a>
     <a href="/vt-middleware/cryptacular/commits/master/NOTICE" class="btn btn-sm BtnGroup-item"
rel="nofollow">History</a>
   </div>


       <button type="button" class="btn-octicon disabled tooltipped tooltipped-nw"
         aria-label="You must be signed in to make or propose changes">
         <svg aria-hidden="true" class="octicon octicon-pencil" height="16" version="1.1" viewBox="0 0 14 16"
width="14"><path fill-rule="evenodd" d="M0 12v3h3l8-8-3-3-8 8zm3 2H1v-2h1v1h1v1zm10.3-9.3L12 6 9 3l1.3-
1.3a.996.996 0 0 1 1.41 0l1.59 1.59c.39.39.39 1.02 0 1.41z"/></svg>
       </button>
       <button type="button" class="btn-octicon btn-octicon-danger disabled tooltipped tooltipped-nw"
         aria-label="You must be signed in to make or propose changes">
         <svg aria-hidden="true" class="octicon octicon-trashcan" height="16" version="1.1" viewBox="0 0 12 16"
width="12"><path fill-rule="evenodd" d="M11 2H9c0-.55-.45-1-1-1H5c-.55 0-1 .45-1 1H2c-.55 0-1 .45-1 1v1c0
```

.55.45 1 1 1v9c0 .55.45 1 1 1h7c.55 0 1-.45 1-1V5c.55 0 1-.45 1-1V3c0-.55-.45-1-1-1zm-1
12H3V5h1v8h1V5h1v8h1V5h1v8h1V5h1v9zm1-10H2V3h9v1z"/></svg>
</button>

7 lines (5 sloc)

165 Bytes

<table class="highlight tab-size js-file-line-container" data-tab-size="8">
<tr>
<td id="L1" class="blob-num js-line-number" data-line-number="1"></td>
<td id="LC1" class="blob-code blob-code-inner js-file-line">Cryptacular Java Library</td>
</tr>
<tr>
<td id="L2" class="blob-num js-line-number" data-line-number="2"></td>
<td id="LC2" class="blob-code blob-code-inner js-file-line">Copyright (C) 2003-2017 Virginia Tech.</td>
</tr>
<tr>
<td id="L3" class="blob-num js-line-number" data-line-number="3"></td>
<td id="LC3" class="blob-code blob-code-inner js-file-line">All rights reserved.</td>
</tr>
<tr>
<td id="L4" class="blob-num js-line-number" data-line-number="4"></td>
<td id="LC4" class="blob-code blob-code-inner js-file-line">
</td>
</tr>
<tr>
<td id="L5" class="blob-num js-line-number" data-line-number="5"></td>
<td id="LC5" class="blob-code blob-code-inner js-file-line">This product includes software developed at</td>
</tr>
<tr>
<td id="L6" class="blob-num js-line-number" data-line-number="6"></td>
<td id="LC6" class="blob-code blob-code-inner js-file-line">Virginia Tech (http://www.vt.edu).</td>
</tr>
</table>

```
 <button type="button" data-facebox="#jump-to-line" data-facebox-class="linejump" data-hotkey="l" class="d-
none">Jump to Line</button>
 <div id="jump-to-line" style="display:none">
   <!-- "'` --><!-- </textarea></xmp> --></option></form><form accept-charset="UTF-8" action="" class="js-jump-
to-line-form" method="get"><div style="margin:0;padding:0;display:inline"><input name="utf8" type="hidden"
value="&#x2713;" /></div>
     <input class="form-control linejump-input js-jump-to-line-field" type="text" placeholder="Jump to line&hellip;"
aria-label="Jump to line" autofocus>
     <button type="submit" class="btn">Go</button>
</form>  </div>

 </div>
 <div class="modal-backdrop js-touch-events"></div>
</div>

   </div>
 </div>

 </div>



<div class="container site-footer-container">
 <div class="site-footer " role="contentinfo">
   <ul class="site-footer-links float-right">
       <li><a href="https://github.com/contact" data-ga-click="Footer, go to contact, text:contact">Contact
GitHub</a></li>
     <li><a href="https://developer.github.com" data-ga-click="Footer, go to api, text:api">API</a></li>
     <li><a href="https://training.github.com" data-ga-click="Footer, go to training, text:training">Training</a></li>
     <li><a href="https://shop.github.com" data-ga-click="Footer, go to shop, text:shop">Shop</a></li>
       <li><a href="https://github.com/blog" data-ga-click="Footer, go to blog, text:blog">Blog</a></li>
       <li><a href="https://github.com/about" data-ga-click="Footer, go to about, text:about">About</a></li>

   </ul>

   <a href="https://github.com" aria-label="Homepage" class="site-footer-mark" title="GitHub">
     <svg aria-hidden="true" class="octicon octicon-mark-github" height="24" version="1.1" viewBox="0 0 16 16"
width="24"><path fill-rule="evenodd" d="M8 0C3.58 0 0 3.58 0 8c0 3.54 2.29 6.53 5.47 7.59.4.07.55-.17.55-.38 0-
.19-.01-.82-.01-1.49-2.01.37-2.53-.49-2.69-.94-.09-.23-.48-.94-.82-1.13-.28-.15-.68-.52-.01-.53.63-.01 1.08.58
1.23.82.72 1.21 1.87.87 2.33.66.07-.52.28-.87.51-1.07-1.78-.2-3.64-.89-3.64-3.95 0-.87.31-1.59.82-2.15-.08-.2-.36-
1.02.08-2.12 0 0 .67-.21 2.2.82.64-.18 1.32-.27 2-.27.68 0 1.36.09 2 .27 1.53-1.04 2.2-.82 2.2-.82.44 1.1.16 1.92.08
2.12.51.56.82 1.27.82 2.15 0 3.07-1.87 3.75-3.65 3.95.29.25.54.73.54 1.48 0 1.07-.01 1.93-.01 2.2 0
.21.15.46.55.38A8.013 8.013 0 0 0 16 8c0-4.42-3.58-8-8-8z"/></svg>
</a>
   <ul class="site-footer-links">
     <li>&copy; 2017 <span title="0.10938s from unicorn-2925809464-f2ltq">GitHub</span>, Inc.</li>
       <li><a href="https://github.com/site/terms" data-ga-click="Footer, go to terms, text:terms">Terms</a></li>
       <li><a href="https://github.com/site/privacy" data-ga-click="Footer, go to privacy,
text:privacy">Privacy</a></li>
```

```
      <li><a href="https://github.com/security" data-ga-click="Footer, go to security,
text:security">Security</a></li>
      <li><a href="https://status.github.com/" data-ga-click="Footer, go to status, text:status">Status</a></li>
      <li><a href="https://help.github.com" data-ga-click="Footer, go to help, text:help">Help</a></li>
  </ul>
 </div>
</div>



 <div id="ajax-error-message" class="ajax-error-message flash flash-error">
   <svg aria-hidden="true" class="octicon octicon-alert" height="16" version="1.1" viewBox="0 0 16 16"
width="16"><path fill-rule="evenodd" d="M8.865 1.52c-.18-.31-.51-.5-.87-.5s-.69.19-.87.5L.275 13.5c-.18.31-
.18.69 0 1 .19.31.52.5.87.5h13.7c.36 0 .69-.19.86-.5.17-.31.18-.69.01-1L8.865 1.52zM8.995 13h-2v-2h2v2zm0-3h-
2V6h2v4z"/></svg>
   <button type="button" class="flash-close js-flash-close js-ajax-error-dismiss" aria-label="Dismiss error">
     <svg aria-hidden="true" class="octicon octicon-x" height="16" version="1.1" viewBox="0 0 12 16"
width="12"><path fill-rule="evenodd" d="M7.48 8l3.75 3.75-1.48 1.48L6 9.48l-3.75 3.75-1.48-1.48L4.52 8 .77
4.25l1.48-1.48L6 6.52l3.75-3.75 1.48 1.48z"/></svg>
   </button>
   You can't perform that action at this time.
 </div>


   <script crossorigin="anonymous" src="https://assets-cdn.github.com/assets/compat-
91f98c37fc84eac24836eec2567e9912742094369a04c4eba6e3cd1fa18902d9.js"></script>
   <script crossorigin="anonymous" src="https://assets-cdn.github.com/assets/frameworks-
f84bb87b149685d1e6c6f057ee324f2cd496e677f5a359a8b5db853313bb83e6.js"></script>

   <script async="async" crossorigin="anonymous" src="https://assets-cdn.github.com/assets/github-
13fa3aa50ac8f9fa9a7d198f0cd13b0905775d39446ad076d17d8f74a998438a.js"></script>



 <div class="js-stale-session-flash stale-session-flash flash flash-warn flash-banner d-none">
   <svg aria-hidden="true" class="octicon octicon-alert" height="16" version="1.1" viewBox="0 0 16 16"
width="16"><path fill-rule="evenodd" d="M8.865 1.52c-.18-.31-.51-.5-.87-.5s-.69.19-.87.5L.275 13.5c-.18.31-
.18.69 0 1 .19.31.52.5.87.5h13.7c.36 0 .69-.19.86-.5.17-.31.18-.69.01-1L8.865 1.52zM8.995 13h-2v-2h2v2zm0-3h-
2V6h2v4z"/></svg>
   <span class="signed-in-tab-flash">You signed in with another tab or window. <a href="">Reload</a> to refresh
your session.</span>
   <span class="signed-out-tab-flash">You signed out in another tab or window. <a href="">Reload</a> to refresh
your session.</span>
 </div>
 <div class="facebox" id="facebox" style="display:none;">
 <div class="facebox-popup">
   <div class="facebox-content" role="dialog" aria-labelledby="facebox-header" aria-describedby="facebox-
```

description">

  <button type="button" class="facebox-close js-facebox-close" aria-label="Close modal">
    <svg aria-hidden="true" class="octicon octicon-x" height="16" version="1.1" viewBox="0 0 12 16" width="12"><path fill-rule="evenodd" d="M7.48 8l3.75 3.75-1.48 1.48L6 9.48l-3.75 3.75-1.48-1.48L4.52 8 .77 4.25l1.48-1.48L6 6.52l3.75-3.75 1.48 1.48z"/></svg>
  </button>

 </body>
</html>

Apache XML Security for Java
Copyright 2000-2016 The Apache Software Foundation

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

Apache Log4j SLF4J Binding
Copyright 1999-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Copyright (c) 2002-2016, the original author or authors.
All rights reserved.

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

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

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

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

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

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,

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

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

package org.elasticsearch.xpack.ccr;

import org.elasticsearch.client.Request;
import org.elasticsearch.client.ResponseException;

import java.util.Locale;

import static org.hamcrest.Matchers.containsString;
import static org.hamcrest.Matchers.hasToString;

public class CcrMultiClusterLicenseIT extends ESCCRRestTestCase {

  public void testFollow() {
    if ("follow".equals(targetCluster)) {
      final Request request = new Request("PUT", "/follower/_ccr/follow");
      request.setJsonEntity("{\"remote_cluster\": \"leader_cluster\", \"leader_index\": \"leader\"}");
      assertNonCompliantLicense(request);
    }
  }

  public void testAutoFollow() throws Exception {
    if ("follow".equals(targetCluster)) {
      final Request request = new Request("PUT", "/_ccr/auto_follow/test_pattern");
      request.setJsonEntity("{\"leader_index_patterns\":[\"*\"], \"remote_cluster\": \"leader_cluster\"}");
      final ResponseException e = expectThrows(ResponseException.class, () ->
client().performRequest(request));
      final String expected = String.format(
        Locale.ROOT,
        "can not fetch remote cluster state as the remote cluster [%s] is not licensed for [ccr]; " +
          "the license mode [BASIC] on cluster [%s] does not enable [ccr]",
        "leader_cluster",
```

```
            "leader_cluster");
        assertThat(e, hasToString(containsString(expected)));
    }
  }

  private static void assertNonCompliantLicense(final Request request) {
    final ResponseException e = expectThrows(ResponseException.class, () -> client().performRequest(request));
    final String expected = String.format(
            Locale.ROOT,
            "can not fetch remote index [%s] metadata as the remote cluster [%s] is not licensed for [ccr]; " +
                    "the license mode [BASIC] on cluster [%s] does not enable [ccr]",
            "leader_cluster:leader",
            "leader_cluster",
            "leader_cluster");
    assertThat(e, hasToString(containsString(expected)));
  }

}
import org.elasticsearch.gradle.test.RestIntegTestTask

apply plugin: 'elasticsearch.standalone-test'

dependencies {
  testCompile project(path: xpackModule('core'), configuration: 'testArtifacts')
  testCompile project(path: xpackModule('ccr'), configuration: 'runtime')
  testCompile project(':x-pack:plugin:ccr:qa:')
}

task leaderClusterTest(type: RestIntegTestTask) {
  mustRunAfter(precommit)
}

leaderClusterTestCluster {
  numNodes = 1
  clusterName = 'leader-cluster'
}

leaderClusterTestRunner {
  systemProperty 'tests.target_cluster', 'leader'
}

task followClusterTest(type: RestIntegTestTask) {}

followClusterTestCluster {
  dependsOn leaderClusterTestRunner
  numNodes = 1
  clusterName = 'follow-cluster'
  setting 'xpack.license.self_generated.type', 'trial'
```

```
  setting 'cluster.remote.leader_cluster.seeds', "\"${-> leaderClusterTest.nodes.get(0).transportUri()}\""
}

followClusterTestRunner {
  systemProperty 'tests.target_cluster', 'follow'
  systemProperty 'tests.leader_host', "${-> leaderClusterTest.nodes.get(0).httpUri()}"
  finalizedBy 'leaderClusterTestCluster#stop'
}

check.dependsOn followClusterTest
test.enabled = false
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.ccr;

import org.apache.lucene.util.Constants;
import org.elasticsearch.client.Request;
import org.elasticsearch.client.ResponseException;
import org.elasticsearch.client.RestClient;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.io.PathUtils;
import org.elasticsearch.common.settings.Settings;

import java.io.IOException;
import java.nio.file.Files;
import java.util.Iterator;
import java.util.List;
import java.util.Map;

import static org.elasticsearch.xpack.core.watcher.support.xcontent.ObjectPath.eval;
import static org.hamcrest.Matchers.containsString;
import static org.hamcrest.Matchers.equalTo;
import static org.hamcrest.Matchers.greaterThanOrEqualTo;
import static org.hamcrest.core.Is.is;

public class FollowIndexIT extends ESCCRRestTestCase {

  public void testDowngradeRemoteClusterToBasic() throws Exception {
    if ("follow".equals(targetCluster) == false) {
      return;
    }

    {
      Request request = new Request("PUT", "/_ccr/auto_follow/test_pattern");
      request.setJsonEntity("{\"leader_index_patterns\": [\"logs-*\"], \"remote_cluster\": \"leader_cluster\"}");
```

```
        assertOK(client().performRequest(request));
    }

    String index1 = "logs-20190101";
    try (RestClient leaderClient = buildLeaderClient()) {
        createNewIndexAndIndexDocs(leaderClient, index1);
    }

    assertBusy(() -> {
        ensureYellow(index1);
        verifyDocuments(index1, 5, "filtered_field:true");
    });

    String index2 = "logs-20190102";
    try (RestClient leaderClient = buildLeaderClient()) {
        Request request = new Request("POST", "/_xpack/license/start_basic");
        request.addParameter("acknowledge", "true");
        Map<?, ?> response = toMap(leaderClient.performRequest(request));
        assertThat(response.get("basic_was_started"), is(true));
        assertThat(response.get("acknowledged"), is(true));

        createNewIndexAndIndexDocs(leaderClient, index2);
        index(leaderClient, index1, "5", "field", 5, "filtered_field", "true");
    }

    assertBusy(() -> {
        Request statsRequest = new Request("GET", "/_ccr/stats");
        Map<?, ?> response = toMap(client().performRequest(statsRequest));
        assertThat(eval("auto_follow_stats.number_of_successful_follow_indices", response), equalTo(1));
        assertThat(eval("auto_follow_stats.number_of_failed_remote_cluster_state_requests", response),
greaterThanOrEqualTo(1));
        assertThat(eval("auto_follow_stats.recent_auto_follow_errors.0.auto_follow_exception.reason", response),
            containsString("the license mode [BASIC] on cluster [leader_cluster] does not enable [ccr]"));

        // Follow indices actively following leader indices before the downgrade to basic license remain to follow
        // the leader index after the downgrade, so document with id 5 should be replicated to follower index:
        verifyDocuments(index1, 6, "filtered_field:true");

        // Index2 was created in leader cluster after the downgrade and therefor the auto follow coordinator in
        // follow cluster should not pick that index up:
        assertThat(indexExists(index2), is(false));

        // parse the logs and ensure that the auto-coordinator skipped coordination on the leader cluster
        // (does not work on windows...)
        if (Constants.WINDOWS == false) {
            assertBusy(() -> {
                final List<String> lines = Files.readAllLines(PathUtils.get(System.getProperty("log")));
                final Iterator<String> it = lines.iterator();
```

```
                boolean warn = false;
                while (it.hasNext()) {
                    final String line = it.next();
                    if (line.matches(".*\\[WARN\\s*\\]\\[o\\.e\\.x\\.c\\.a\\.AutoFollowCoordinator\\s*\\] \\[node-0\\] " +
                        "failure occurred while fetching cluster state for auto follow pattern \\[test_pattern\\]")) {
                        warn = true;
                        break;
                    }
                }
                assertTrue(warn);
                assertTrue(it.hasNext());
                final String lineAfterWarn = it.next();
                assertThat(
                    lineAfterWarn,
                    equalTo("org.elasticsearch.ElasticsearchStatusException: " +
                        "can not fetch remote cluster state as the remote cluster [leader_cluster] is not licensed for [ccr]; " +
                        "the license mode [BASIC] on cluster [leader_cluster] does not enable [ccr]"));
            });
        }
    });

    // Manually following index2 also does not work after the downgrade:
    Exception e = expectThrows(ResponseException.class, () -> followIndex("leader_cluster", index2));
    assertThat(e.getMessage(), containsString("the license mode [BASIC] on cluster [leader_cluster] does not
enable [ccr]"));
  }

  private void createNewIndexAndIndexDocs(RestClient client, String index) throws IOException {
    Settings settings = Settings.builder()
        .put("index.soft_deletes.enabled", true)
        .build();
    Request request = new Request("PUT", "/" + index);
    request.setJsonEntity("{\"settings\": " + Strings.toString(settings) +
        ", \"mappings\": {\"_doc\": {\"properties\": {\"field\": {\"type\": \"keyword\"}}}} }");
    assertOK(client.performRequest(request));

    for (int i = 0; i < 5; i++) {
        String id = Integer.toString(i);
        index(client, index, id, "field", i, "filtered_field", "true");
    }
  }

}
import org.elasticsearch.gradle.test.RestIntegTestTask

apply plugin: 'elasticsearch.standalone-test'

dependencies {
```

```
  testCompile project(path: xpackModule('core'), configuration: 'testArtifacts')
  testCompile project(path: xpackModule('ccr'), configuration: 'runtime')
  testCompile project(':x-pack:plugin:ccr:qa')
}

task leaderClusterTest(type: RestIntegTestTask) {
  mustRunAfter(precommit)
}

leaderClusterTestCluster {
  numNodes = 1
  clusterName = 'leader-cluster'
  setting 'xpack.license.self_generated.type', 'trial'
}

leaderClusterTestRunner {
  systemProperty 'tests.target_cluster', 'leader'
}

task writeJavaPolicy {
  doLast {
    final File tmp = file("${buildDir}/tmp")
    if (tmp.exists() == false && tmp.mkdirs() == false) {
      throw new GradleException("failed to create temporary directory [${tmp}]")
    }
    final File javaPolicy = file("${tmp}/java.policy")
    javaPolicy.write(
        [
            "grant {",
            "  permission java.io.FilePermission \"${-> followClusterTest.getNodes().get(0).homeDir}/logs/${->
followClusterTest.getNodes().get(0).clusterName}.log\", \"read\";",
            "};"
        ].join("\n"))
  }
}

task followClusterTest(type: RestIntegTestTask) {}
followClusterTest.dependsOn writeJavaPolicy

followClusterTestCluster {
  dependsOn leaderClusterTestRunner
  numNodes = 1
  clusterName = 'follow-cluster'
  setting 'xpack.monitoring.collection.enabled', 'true'
  setting 'xpack.license.self_generated.type', 'trial'
  setting 'cluster.remote.leader_cluster.seeds', "\"${-> leaderClusterTest.nodes.get(0).transportUri()}\""
}
```

```
followClusterTestRunner {
  systemProperty 'java.security.policy', "file://${buildDir}/tmp/java.policy"
  systemProperty 'tests.target_cluster', 'follow'
  systemProperty 'tests.leader_host', "${-> leaderClusterTest.nodes.get(0).httpUri()}"
  systemProperty 'log', "${-> followClusterTest.getNodes().get(0).homeDir}/logs/${->
followClusterTest.getNodes().get(0).clusterName}.log"
  finalizedBy 'leaderClusterTestCluster#stop'
}

check.dependsOn followClusterTest
test.enabled = false // no unit tests for multi-cluster-search, only the rest integration test
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.util.set.Sets;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.xpack.core.security.authz.privilege.ApplicationPrivilege;
import org.elasticsearch.xpack.core.security.authz.privilege.ApplicationPrivilegeDescriptor;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.Set;
import java.util.stream.Collectors;
import java.util.stream.Stream;

import static java.util.Collections.singletonList;
import static org.hamcrest.Matchers.containsInAnyOrder;
import static org.hamcrest.Matchers.equalTo;

public class ApplicationPermissionTests extends ESTestCase {

  private List<ApplicationPrivilegeDescriptor> store = new ArrayList<>();

  private ApplicationPrivilege app1All = storePrivilege("app1", "all", "*");
  private ApplicationPrivilege app1Empty = storePrivilege("app1", "empty");
  private ApplicationPrivilege app1Read = storePrivilege("app1", "read", "read/*");
  private ApplicationPrivilege app1Write = storePrivilege("app1", "write", "write/*");
  private ApplicationPrivilege app1Delete = storePrivilege("app1", "delete", "write/delete");
  private ApplicationPrivilege app1Create = storePrivilege("app1", "create", "write/create");
  private ApplicationPrivilege app2Read = storePrivilege("app2", "read", "read/*");
```

```
  private ApplicationPrivilege storePrivilege(String app, String name, String... patterns) {
      store.add(new ApplicationPrivilegeDescriptor(app, name, Sets.newHashSet(patterns),
Collections.emptyMap()));
      return new ApplicationPrivilege(app, name, patterns);
  }

  public void testCheckSimplePermission() {
      final ApplicationPermission hasPermission = buildPermission(app1Write, "*");
      assertThat(hasPermission.grants(app1Write, "*"), equalTo(true));
      assertThat(hasPermission.grants(app1Write, "foo"), equalTo(true));
      assertThat(hasPermission.grants(app1Delete, "*"), equalTo(true));
      assertThat(hasPermission.grants(app1Create, "foo"), equalTo(true));

      assertThat(hasPermission.grants(app1Read, "*"), equalTo(false));
      assertThat(hasPermission.grants(app1Read, "foo"), equalTo(false));
      assertThat(hasPermission.grants(app1All, "*"), equalTo(false));
      assertThat(hasPermission.grants(app1All, "foo"), equalTo(false));
  }

  public void testNonePermission() {
      final ApplicationPermission hasPermission = buildPermission(ApplicationPrivilege.NONE.apply("app1"), "*");
      for (ApplicationPrivilege privilege : Arrays.asList(app1All, app1Empty, app1Create, app1Delete, app1Read,
app1Write, app2Read)) {
          assertThat("Privilege " + privilege + " on *", hasPermission.grants(privilege, "*"), equalTo(false));
          final String resource = randomAlphaOfLengthBetween(1, 6);
          assertThat("Privilege " + privilege + " on " + resource, hasPermission.grants(privilege, resource),
equalTo(false));
      }
  }

  public void testResourceMatching() {
      final ApplicationPermission hasPermission = buildPermission(app1All, "dashboard/*", "audit/*",
"user/12345");

      assertThat(hasPermission.grants(app1Write, "*"), equalTo(false));
      assertThat(hasPermission.grants(app1Write, "dashboard"), equalTo(false));
      assertThat(hasPermission.grants(app1Write, "dashboard/999"), equalTo(true));

      assertThat(hasPermission.grants(app1Create, "audit/2018-02-21"), equalTo(true));
      assertThat(hasPermission.grants(app1Create, "report/2018-02-21"), equalTo(false));

      assertThat(hasPermission.grants(app1Read, "user/12345"), equalTo(true));
      assertThat(hasPermission.grants(app1Read, "user/67890"), equalTo(false));

      assertThat(hasPermission.grants(app1All, "dashboard/999"), equalTo(true));
      assertThat(hasPermission.grants(app1All, "audit/2018-02-21"), equalTo(true));
      assertThat(hasPermission.grants(app1All, "user/12345"), equalTo(true));
  }
```

```
public void testActionMatching() {
    final ApplicationPermission hasPermission = buildPermission(app1Write, "allow/*");

    final ApplicationPrivilege update = actionPrivilege("app1", "write/update");
    assertThat(hasPermission.grants(update, "allow/1"), equalTo(true));
    assertThat(hasPermission.grants(update, "deny/1"), equalTo(false));

    final ApplicationPrivilege updateCreate = actionPrivilege("app1", "write/update", "write/create");
    assertThat(hasPermission.grants(updateCreate, "allow/1"), equalTo(true));
    assertThat(hasPermission.grants(updateCreate, "deny/1"), equalTo(false));

    final ApplicationPrivilege manage = actionPrivilege("app1", "admin/manage");
    assertThat(hasPermission.grants(manage, "allow/1"), equalTo(false));
    assertThat(hasPermission.grants(manage, "deny/1"), equalTo(false));
}

public void testDoesNotMatchAcrossApplications() {
    assertThat(buildPermission(app1Read, "*").grants(app1Read, "123"), equalTo(true));
    assertThat(buildPermission(app1All, "*").grants(app1Read, "123"), equalTo(true));

    assertThat(buildPermission(app1Read, "*").grants(app2Read, "123"), equalTo(false));
    assertThat(buildPermission(app1All, "*").grants(app2Read, "123"), equalTo(false));
}

public void testMergedPermissionChecking() {
    final ApplicationPrivilege app1ReadWrite = compositePrivilege("app1", app1Read, app1Write);
    final ApplicationPermission hasPermission = buildPermission(app1ReadWrite, "allow/*");

    assertThat(hasPermission.grants(app1Read, "allow/1"), equalTo(true));
    assertThat(hasPermission.grants(app1Write, "allow/1"), equalTo(true));

    assertThat(hasPermission.grants(app1Read, "deny/1"), equalTo(false));
    assertThat(hasPermission.grants(app1Write, "deny/1"), equalTo(false));

    assertThat(hasPermission.grants(app1All, "allow/1"), equalTo(false));
    assertThat(hasPermission.grants(app2Read, "allow/1"), equalTo(false));
}

public void testInspectPermissionContents() {
    final ApplicationPrivilege app1ReadWrite = compositePrivilege("app1", app1Read, app1Write);
    ApplicationPermission perm = new ApplicationPermission(Arrays.asList(
        new Tuple<>(app1Read, Sets.newHashSet("obj/1", "obj/2")),
        new Tuple<>(app1Write, Sets.newHashSet("obj/3", "obj/4")),
        new Tuple<>(app1ReadWrite, Sets.newHashSet("obj/5")),
        new Tuple<>(app1All, Sets.newHashSet("obj/6", "obj/7")),
        new Tuple<>(app2Read, Sets.newHashSet("obj/1", "obj/8"))
    ));
```

```
        assertThat(perm.getApplicationNames(), containsInAnyOrder("app1", "app2"));
        assertThat(perm.getPrivileges("app1"), containsInAnyOrder(app1Read, app1Write, app1ReadWrite, app1All));
        assertThat(perm.getPrivileges("app2"), containsInAnyOrder(app2Read));
        assertThat(perm.getResourcePatterns(app1Read), containsInAnyOrder("obj/1", "obj/2", "obj/5", "obj/6",
"obj/7"));
        assertThat(perm.getResourcePatterns(app1Write), containsInAnyOrder("obj/3", "obj/4", "obj/5", "obj/6",
"obj/7"));
        assertThat(perm.getResourcePatterns(app1ReadWrite), containsInAnyOrder("obj/5", "obj/6", "obj/7"));
        assertThat(perm.getResourcePatterns(app1All), containsInAnyOrder("obj/6", "obj/7"));
        assertThat(perm.getResourcePatterns(app2Read), containsInAnyOrder("obj/1", "obj/8"));
    }

    private ApplicationPrivilege actionPrivilege(String appName, String... actions) {
        return ApplicationPrivilege.get(appName, Sets.newHashSet(actions), Collections.emptyList());
    }

    private ApplicationPrivilege compositePrivilege(String application, ApplicationPrivilege... children) {
        Set<String> names =
Stream.of(children).map(ApplicationPrivilege::name).flatMap(Set::stream).collect(Collectors.toSet());
        return ApplicationPrivilege.get(application, names, store);
    }


    private ApplicationPermission buildPermission(ApplicationPrivilege privilege, String... resources) {
        return new ApplicationPermission(singletonList(new Tuple<>(privilege, Sets.newHashSet(resources))));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.test.ESTestCase;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;

public class FieldPermissionsCacheTests extends ESTestCase {

    public void testFieldPermissionsCaching() {
        FieldPermissionsCache fieldPermissionsCache = new FieldPermissionsCache(Settings.EMPTY);
        String[] allowed = new String[]{randomAlphaOfLength(5) + "*", randomAlphaOfLength(5) + "*",
randomAlphaOfLength(5) + "*"};
```

```
        String[] denied = new String[]{allowed[0] + randomAlphaOfLength(5), allowed[1] +
randomAlphaOfLength(5),
                allowed[2] + randomAlphaOfLength(5)};
        FieldPermissions fieldPermissions = fieldPermissionsCache.getFieldPermissions(allowed, denied);
        assertNotNull(fieldPermissions);
        final String[] allowed2 = randomBoolean() ? allowed : Arrays.copyOf(allowed, allowed.length);
        final String[] denied2 = randomBoolean() ? denied : Arrays.copyOf(denied, denied.length);
        assertSame(fieldPermissions, fieldPermissionsCache.getFieldPermissions(allowed2, denied2));
    }

    public void testMergeFieldPermissions() {
        FieldPermissionsCache fieldPermissionsCache = new FieldPermissionsCache(Settings.EMPTY);
        String allowedPrefix1 = randomAlphaOfLength(5);
        String allowedPrefix2 = randomAlphaOfLength(5);
        String[] allowed1 = new String[]{allowedPrefix1 + "*"};
        String[] allowed2 = new String[]{allowedPrefix2 + "*"};
        String[] denied1 = new String[]{allowedPrefix1 + "a"};
        String[] denied2 = new String[]{allowedPrefix2 + "a"};
        FieldPermissions fieldPermissions1 = randomBoolean() ?
fieldPermissionsCache.getFieldPermissions(allowed1, denied1) :
                new FieldPermissions(fieldPermissionDef(allowed1, denied1));
        FieldPermissions fieldPermissions2 = randomBoolean() ?
fieldPermissionsCache.getFieldPermissions(allowed2, denied2) :
                new FieldPermissions(fieldPermissionDef(allowed2, denied2));
        FieldPermissions mergedFieldPermissions =
                fieldPermissionsCache.getFieldPermissions(Arrays.asList(fieldPermissions1, fieldPermissions2));
        assertTrue(mergedFieldPermissions.grantsAccessTo(allowedPrefix1 + "b"));
        assertTrue(mergedFieldPermissions.grantsAccessTo(allowedPrefix2 + "b"));
        assertFalse(mergedFieldPermissions.grantsAccessTo(denied1[0]));
        assertFalse(mergedFieldPermissions.grantsAccessTo(denied2[0]));

        allowed1 = new String[]{randomAlphaOfLength(5) + "*", randomAlphaOfLength(5) + "*"};
        allowed2 = null;
        denied1 = new String[]{allowed1[0] + "a", allowed1[1] + "a"};
        denied2 = null;
        fieldPermissions1 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed1, denied1) :
                new FieldPermissions(fieldPermissionDef(allowed1, denied1));
        fieldPermissions2 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed2, denied2) :
                new FieldPermissions(fieldPermissionDef(allowed2, denied2));
        mergedFieldPermissions =
                fieldPermissionsCache.getFieldPermissions(Arrays.asList(fieldPermissions1, fieldPermissions2));
        assertFalse(mergedFieldPermissions.hasFieldLevelSecurity());

        allowed1 = new String[]{};
        allowed2 = new String[]{randomAlphaOfLength(5) + "*", randomAlphaOfLength(5) + "*"};
        denied1 = new String[]{};
        denied2 = new String[]{allowed2[0] + "a", allowed2[1] + "a"};
        fieldPermissions1 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed1, denied1) :
```

```
        new FieldPermissions(fieldPermissionDef(allowed1, denied1));
fieldPermissions2 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed2, denied2) :
        new FieldPermissions(fieldPermissionDef(allowed2, denied2));
mergedFieldPermissions =
        fieldPermissionsCache.getFieldPermissions(Arrays.asList(fieldPermissions1, fieldPermissions2));
for (String field : allowed2) {
    assertTrue(mergedFieldPermissions.grantsAccessTo(field));
}
for (String field : denied2) {
    assertFalse(mergedFieldPermissions.grantsAccessTo(field));
}

allowed1 = randomBoolean() ? null : new String[]{"*"};
allowed2 = randomBoolean() ? null : new String[]{"*"};
denied1 = new String[]{"a"};
denied2 = new String[]{"b"};
fieldPermissions1 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed1, denied1) :
        new FieldPermissions(fieldPermissionDef(allowed1, denied1));
fieldPermissions2 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed2, denied2) :
        new FieldPermissions(fieldPermissionDef(allowed2, denied2));
mergedFieldPermissions =
        fieldPermissionsCache.getFieldPermissions(Arrays.asList(fieldPermissions1, fieldPermissions2));
assertTrue(mergedFieldPermissions.grantsAccessTo("a"));
assertTrue(mergedFieldPermissions.grantsAccessTo("b"));

// test merge does not remove _all
allowed1 = new String[]{"_all"};
allowed2 = new String[]{};
denied1 = null;
denied2 = null;
fieldPermissions1 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed1, denied1) :
        new FieldPermissions(fieldPermissionDef(allowed1, denied1));
fieldPermissions2 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed2, denied2) :
        new FieldPermissions(fieldPermissionDef(allowed2, denied2));
mergedFieldPermissions =
        fieldPermissionsCache.getFieldPermissions(Arrays.asList(fieldPermissions1, fieldPermissions2));
assertTrue(fieldPermissions1.grantsAccessTo("_all"));
assertFalse(fieldPermissions2.grantsAccessTo("_all"));
assertTrue(mergedFieldPermissions.grantsAccessTo("_all"));

allowed1 = new String[] { "a*" };
allowed2 = new String[] { "b*" };
denied1 = new String[] { "aa*" };
denied2 = null;
fieldPermissions1 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed1, denied1) :
        new FieldPermissions(fieldPermissionDef(allowed1, denied1));
fieldPermissions2 = randomBoolean() ? fieldPermissionsCache.getFieldPermissions(allowed2, denied2) :
        new FieldPermissions(fieldPermissionDef(allowed2, denied2));
```

```
        mergedFieldPermissions =
                fieldPermissionsCache.getFieldPermissions(Arrays.asList(fieldPermissions1, fieldPermissions2));
        assertTrue(mergedFieldPermissions.grantsAccessTo("a"));
        assertTrue(mergedFieldPermissions.grantsAccessTo("b"));
        assertFalse(mergedFieldPermissions.grantsAccessTo("aa"));
        assertFalse(mergedFieldPermissions.grantsAccessTo("aa1"));
        assertTrue(mergedFieldPermissions.grantsAccessTo("a1"));
    }

    public void testNonFlsAndFlsMerging() {
        List<FieldPermissions> permissionsList = new ArrayList<>();
        permissionsList.add(new FieldPermissions(fieldPermissionDef(new String[] {"field1"}, null)));
        permissionsList.add(new FieldPermissions(fieldPermissionDef(new String[] {"field2", "query*"}, null)));
        permissionsList.add(new FieldPermissions(fieldPermissionDef(new String[] {"field1", "field2"}, null)));
        permissionsList.add(new FieldPermissions(fieldPermissionDef(new String[] {}, null)));
        permissionsList.add(new FieldPermissions(fieldPermissionDef(null, null)));

        FieldPermissionsCache cache = new FieldPermissionsCache(Settings.EMPTY);
        for (int i = 0; i < scaledRandomIntBetween(1, 12); i++) {
            Collections.shuffle(permissionsList, random());
            FieldPermissions result = cache.getFieldPermissions(permissionsList);
            assertFalse(result.hasFieldLevelSecurity());
        }
    }

    private static FieldPermissionsDefinition fieldPermissionDef(String[] granted, String[] denied) {
        return new FieldPermissionsDefinition(granted, denied);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import java.io.IOException;

import org.elasticsearch.test.ESTestCase;

public class LicenseStatusTests extends ESTestCase {
    public void testSerialization() throws IOException {
        LicenseStatus status = randomFrom(LicenseStatus.values());
        assertSame(status, copyWriteable(status, writableRegistry(), LicenseStatus::readFrom));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
```

```
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.test.AbstractStreamableXContentTestCase;

import java.io.IOException;
import java.util.Collections;
import java.util.HashMap;
import java.util.Map;
import java.util.function.Function;
import java.util.function.Predicate;

public class PutLicenseResponseTests extends AbstractStreamableXContentTestCase<PutLicenseResponse> {

  @Override
  protected boolean supportsUnknownFields() {
    return true;
  }

  @Override
  protected Predicate<String> getRandomFieldsExcludeFilter() {
    // The structure of the response is such that unknown fields inside acknowledge cannot be supported since they
    // are treated as messages from new services
    return p -> p.startsWith("acknowledge");
  }

  @Override
  protected PutLicenseResponse createTestInstance() {
    boolean acknowledged = randomBoolean();
    LicensesStatus status = randomFrom(LicensesStatus.VALID, LicensesStatus.INVALID,
LicensesStatus.EXPIRED);
    String messageHeader;
    Map<String, String[]> ackMessages;
    if (randomBoolean()) {
      messageHeader = randomAlphaOfLength(10);
      ackMessages = randomAckMessages();
    } else {
      messageHeader = null;
      ackMessages = Collections.emptyMap();
    }

    return new PutLicenseResponse(acknowledged, status, messageHeader, ackMessages);
  }

  private static Map<String, String[]> randomAckMessages() {
```

```
    int nFeatures = randomIntBetween(1, 5);

    Map<String, String[]> ackMessages = new HashMap<>();

    for (int i = 0; i < nFeatures; i++) {
        String feature = randomAlphaOfLengthBetween(9, 15);
        int nMessages = randomIntBetween(1, 5);
        String[] messages = new String[nMessages];
        for (int j = 0; j < nMessages; j++) {
            messages[j] = randomAlphaOfLengthBetween(10, 30);
        }
        ackMessages.put(feature, messages);
    }

    return ackMessages;
}

@Override
protected PutLicenseResponse doParseInstance(XContentParser parser) throws IOException {
    return PutLicenseResponse.fromXContent(parser);
}

@Override
protected PutLicenseResponse createBlankInstance() {
    return new PutLicenseResponse();
}

@Override
protected PutLicenseResponse mutateInstance(PutLicenseResponse response) {
    @SuppressWarnings("unchecked")
    Function<PutLicenseResponse, PutLicenseResponse> mutator = randomFrom(
        r -> new PutLicenseResponse(
            r.isAcknowledged() == false,
            r.status(),
            r.acknowledgeHeader(),
            r.acknowledgeMessages()),
        r -> new PutLicenseResponse(
            r.isAcknowledged(),
            mutateStatus(r.status()),
            r.acknowledgeHeader(),
            r.acknowledgeMessages()),
        r -> {
            if (r.acknowledgeMessages().isEmpty()) {
                return new PutLicenseResponse(
                    r.isAcknowledged(),
                    r.status(),
                    randomAlphaOfLength(10),
                    randomAckMessages()
```

```
                );
            } else {
                return new PutLicenseResponse(r.isAcknowledged(), r.status());
            }
        }

    );
    return mutator.apply(response);
  }

  private LicensesStatus mutateStatus(LicensesStatus status) {
    return randomValueOtherThan(status, () -> randomFrom(LicensesStatus.values()));
  }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.license.License.OperationMode;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.test.VersionUtils;
import org.elasticsearch.xpack.core.XPackField;
import org.elasticsearch.xpack.core.XPackSettings;

import java.util.Arrays;
import java.util.function.Predicate;
import java.util.stream.Collectors;

import static org.elasticsearch.license.License.OperationMode.BASIC;
import static org.elasticsearch.license.License.OperationMode.GOLD;
import static org.elasticsearch.license.License.OperationMode.MISSING;
import static org.elasticsearch.license.License.OperationMode.PLATINUM;
import static org.elasticsearch.license.License.OperationMode.STANDARD;
import static org.elasticsearch.license.License.OperationMode.TRIAL;
import static org.hamcrest.Matchers.is;

/**
* Unit tests for the {@link XPackLicenseState}
*/
public class XPackLicenseStateTests extends ESTestCase {

  /** Creates a license state with the given license type and active state, and checks the given method returns
```

```
expected. */
  void assertAllowed(OperationMode mode, boolean active, Predicate<XPackLicenseState> predicate, boolean
expected) {
      XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
      licenseState.update(mode, active, null);
      assertEquals(expected, predicate.test(licenseState));
  }

  /**
   * Checks the ack message going from the  {@code from} license type to {@code to} license type.
   * TODO: check the actual messages, not just the number of them! This was copied from previous license tests...
   */
  void assertAckMesssages(String feature, OperationMode from, OperationMode to, int expectedMessages) {
      String[] gotMessages = XPackLicenseState.ACKNOWLEDGMENT_MESSAGES.get(feature).apply(from, to);
      assertEquals(expectedMessages, gotMessages.length);
  }

  static <T> T randomFrom(T[] values, Predicate<T> filter) {
      return randomFrom(Arrays.stream(values).filter(filter).collect(Collectors.toList()));
  }

  static OperationMode randomMode() {
      return randomFrom(OperationMode.values());
  }

  public static OperationMode randomTrialStandardGoldOrPlatinumMode() {
      return randomFrom(TRIAL, STANDARD, GOLD, PLATINUM);
  }

  public static OperationMode randomTrialOrPlatinumMode() {
      return randomFrom(TRIAL, PLATINUM);
  }

  public static OperationMode randomTrialBasicStandardGoldOrPlatinumMode() {
      return randomFrom(TRIAL, BASIC, STANDARD, GOLD, PLATINUM);
  }

  public static OperationMode randomBasicStandardOrGold() {
      return randomFrom(BASIC, STANDARD, GOLD);
  }

  public void testSecurityDefaults() {
      XPackLicenseState licenseState =
              new XPackLicenseState(Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(),
true).build());
      assertThat(licenseState.isAuthAllowed(), is(true));
      assertThat(licenseState.isIpFilteringAllowed(), is(true));
      assertThat(licenseState.isAuditingAllowed(), is(true));
```

```
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(true));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.ALL));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(true));

        licenseState =
            new XPackLicenseState(Settings.builder().put(XPackSettings.TRANSPORT_SSL_ENABLED.getKey(),
true).build());
        assertThat(licenseState.isAuthAllowed(), is(true));
        assertThat(licenseState.isIpFilteringAllowed(), is(true));
        assertThat(licenseState.isAuditingAllowed(), is(true));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(true));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.ALL));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(true));

        licenseState = new XPackLicenseState(Settings.EMPTY);
        assertThat(licenseState.isAuthAllowed(), is(false));
        assertThat(licenseState.isIpFilteringAllowed(), is(false));
        assertThat(licenseState.isAuditingAllowed(), is(false));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.NONE));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
    }

    public void testSecurityBasic() {
        XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
                Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
        licenseState.update(BASIC, true, null);

        assertThat(licenseState.isAuthAllowed(), is(false));
        assertThat(licenseState.isIpFilteringAllowed(), is(false));
        assertThat(licenseState.isAuditingAllowed(), is(false));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.NONE));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
    }

    public void testSecurityBasicExpired() {
        XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
                Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
        licenseState.update(BASIC, false, null);

        assertThat(licenseState.isAuthAllowed(), is(false));
        assertThat(licenseState.isIpFilteringAllowed(), is(false));
        assertThat(licenseState.isAuditingAllowed(), is(false));
```

```
    assertThat(licenseState.isStatsAndHealthAllowed(), is(false));
    assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
    assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.NONE));
    assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
}

public void testSecurityStandard() {
    XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
            Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
    licenseState.update(STANDARD, true, null);

    assertThat(licenseState.isAuthAllowed(), is(true));
    assertThat(licenseState.isIpFilteringAllowed(), is(false));
    assertThat(licenseState.isAuditingAllowed(), is(false));
    assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
    assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
    assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.NATIVE));
    assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
}

public void testSecurityStandardExpired() {
    XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
            Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
    licenseState.update(STANDARD, false, null);

    assertThat(licenseState.isAuthAllowed(), is(true));
    assertThat(licenseState.isIpFilteringAllowed(), is(false));
    assertThat(licenseState.isAuditingAllowed(), is(false));
    assertThat(licenseState.isStatsAndHealthAllowed(), is(false));
    assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
    assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.NATIVE));
    assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
}

public void testSecurityGold() {
    XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
            Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
    licenseState.update(GOLD, true, null);

    assertThat(licenseState.isAuthAllowed(), is(true));
    assertThat(licenseState.isIpFilteringAllowed(), is(true));
    assertThat(licenseState.isAuditingAllowed(), is(true));
    assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
    assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
    assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.DEFAULT));
    assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
}
```

```
    public void testSecurityGoldExpired() {
        XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
                Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
        licenseState.update(GOLD, false, null);

        assertThat(licenseState.isAuthAllowed(), is(true));
        assertThat(licenseState.isIpFilteringAllowed(), is(true));
        assertThat(licenseState.isAuditingAllowed(), is(true));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(false));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.DEFAULT));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
    }

    public void testSecurityPlatinum() {
        XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
                Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
        licenseState.update(PLATINUM, true, null);

        assertThat(licenseState.isAuthAllowed(), is(true));
        assertThat(licenseState.isIpFilteringAllowed(), is(true));
        assertThat(licenseState.isAuditingAllowed(), is(true));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(true));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.ALL));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(true));
    }

    public void testSecurityPlatinumExpired() {
        XPackLicenseState licenseState = new XPackLicenseState(randomFrom(Settings.EMPTY,
                Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), true).build()));
        licenseState.update(PLATINUM, false, null);

        assertThat(licenseState.isAuthAllowed(), is(true));
        assertThat(licenseState.isIpFilteringAllowed(), is(true));
        assertThat(licenseState.isAuditingAllowed(), is(true));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(false));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(true));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.ALL));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
    }

    public void testNewTrialDefaultsSecurityOff() {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(TRIAL, true, VersionUtils.randomVersionBetween(random(), Version.V_6_3_0,
Version.CURRENT));

        assertThat(licenseState.isSecurityDisabledByTrialLicense(), is(true));
```

```
        assertThat(licenseState.isAuthAllowed(), is(false));
        assertThat(licenseState.isIpFilteringAllowed(), is(false));
        assertThat(licenseState.isAuditingAllowed(), is(false));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(false));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.NONE));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(false));
    }

    public void testOldTrialDefaultsSecurityOn() {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(TRIAL, true, rarely() ? null : VersionUtils.randomVersionBetween(random(),
Version.V_5_6_0, Version.V_6_2_4));

        assertThat(licenseState.isSecurityDisabledByTrialLicense(), is(false));
        assertThat(licenseState.isAuthAllowed(), is(true));
        assertThat(licenseState.isIpFilteringAllowed(), is(true));
        assertThat(licenseState.isAuditingAllowed(), is(true));
        assertThat(licenseState.isStatsAndHealthAllowed(), is(true));
        assertThat(licenseState.isDocumentAndFieldLevelSecurityAllowed(), is(true));
        assertThat(licenseState.allowedRealmType(), is(XPackLicenseState.AllowedRealmType.ALL));
        assertThat(licenseState.isCustomRoleProvidersAllowed(), is(true));
    }

    public void testSecurityAckBasicToNotGoldOrStandard() {
        OperationMode toMode = randomFrom(OperationMode.values(), mode -> mode != GOLD && mode !=
STANDARD);
        assertAckMesssages(XPackField.SECURITY, BASIC, toMode, 0);
    }

    public void testSecurityAckAnyToTrialOrPlatinum() {
        assertAckMesssages(XPackField.SECURITY, randomMode(), randomTrialOrPlatinumMode(), 0);
    }

    public void testSecurityAckTrialStandardGoldOrPlatinumToBasic() {
        assertAckMesssages(XPackField.SECURITY, randomTrialStandardGoldOrPlatinumMode(), BASIC, 3);
    }

    public void testSecurityAckAnyToStandard() {
        OperationMode from = randomFrom(BASIC, GOLD, PLATINUM, TRIAL);
        assertAckMesssages(XPackField.SECURITY, from, STANDARD, 4);
    }

    public void testSecurityAckBasicStandardTrialOrPlatinumToGold() {
        OperationMode from = randomFrom(BASIC, PLATINUM, TRIAL, STANDARD);
        assertAckMesssages(XPackField.SECURITY, from, GOLD, 2);
    }
```

```
    public void testMonitoringAckBasicToAny() {
        assertAckMesssages(XPackField.MONITORING, BASIC, randomMode(), 0);
    }

    public void testMonitoringAckAnyToTrialGoldOrPlatinum() {
        assertAckMesssages(XPackField.MONITORING, randomMode(),
randomTrialStandardGoldOrPlatinumMode(), 0);
    }

    public void testMonitoringAckNotBasicToBasic() {
        OperationMode from = randomFrom(STANDARD, GOLD, PLATINUM, TRIAL);
        assertAckMesssages(XPackField.MONITORING, from, BASIC, 2);
    }

    public void testMonitoringAllowed() {
        assertAllowed(randomMode(), true, XPackLicenseState::isMonitoringAllowed, true);
        assertAllowed(randomMode(), false, XPackLicenseState::isMonitoringAllowed, false);
    }

    public void testMonitoringUpdateRetention() {
        assertAllowed(STANDARD, true, XPackLicenseState::isUpdateRetentionAllowed, true);
        assertAllowed(GOLD, true, XPackLicenseState::isUpdateRetentionAllowed, true);
        assertAllowed(PLATINUM, true, XPackLicenseState::isUpdateRetentionAllowed, true);
        assertAllowed(TRIAL, true, XPackLicenseState::isUpdateRetentionAllowed, true);
        assertAllowed(BASIC, true, XPackLicenseState::isUpdateRetentionAllowed, false);
        assertAllowed(MISSING, false, XPackLicenseState::isUpdateRetentionAllowed, false);
    }

    public void testWatcherPlatinumGoldTrialStandard() throws Exception {
        assertAllowed(TRIAL, true, XPackLicenseState::isWatcherAllowed, true);
        assertAllowed(GOLD, true, XPackLicenseState::isWatcherAllowed, true);
        assertAllowed(PLATINUM, true, XPackLicenseState::isWatcherAllowed, true);
        assertAllowed(STANDARD, true, XPackLicenseState::isWatcherAllowed, true);
    }

    public void testWatcherBasicLicense() throws Exception {
        assertAllowed(BASIC, true, XPackLicenseState::isWatcherAllowed, false);
    }

    public void testWatcherInactive() {
        assertAllowed(BASIC, false, XPackLicenseState::isWatcherAllowed, false);
    }

    public void testWatcherInactivePlatinumGoldTrial() throws Exception {
        assertAllowed(TRIAL, false, XPackLicenseState::isWatcherAllowed, false);
        assertAllowed(GOLD, false, XPackLicenseState::isWatcherAllowed, false);
        assertAllowed(PLATINUM, false, XPackLicenseState::isWatcherAllowed, false);
        assertAllowed(STANDARD, false, XPackLicenseState::isWatcherAllowed, false);
```

```
}

public void testGraphPlatinumTrial() throws Exception {
    assertAllowed(TRIAL, true, XPackLicenseState::isGraphAllowed, true);
    assertAllowed(PLATINUM, true, XPackLicenseState::isGraphAllowed, true);
}

public void testGraphBasic() throws Exception {
    assertAllowed(BASIC, true, XPackLicenseState::isGraphAllowed, false);
}

public void testGraphStandard() throws Exception {
    assertAllowed(STANDARD, true, XPackLicenseState::isGraphAllowed, false);
}

public void testGraphInactiveBasic() {
    assertAllowed(BASIC, false, XPackLicenseState::isGraphAllowed, false);
}

public void testGraphInactivePlatinumTrial() throws Exception {
    assertAllowed(TRIAL, false, XPackLicenseState::isMachineLearningAllowed, false);
    assertAllowed(PLATINUM, false, XPackLicenseState::isMachineLearningAllowed, false);
}

public void testMachineLearningPlatinumTrial() throws Exception {
    assertAllowed(TRIAL, true, XPackLicenseState::isMachineLearningAllowed, true);
    assertAllowed(PLATINUM, true, XPackLicenseState::isMachineLearningAllowed, true);
}

public void testMachineLearningBasic() throws Exception {
    assertAllowed(BASIC, true, XPackLicenseState::isMachineLearningAllowed, false);
}

public void testMachineLearningStandard() throws Exception {
    assertAllowed(STANDARD, true, XPackLicenseState::isMachineLearningAllowed, false);
}

public void testMachineLearningInactiveBasic() {
    assertAllowed(BASIC, false, XPackLicenseState::isMachineLearningAllowed, false);
}

public void testMachineLearningInactivePlatinumTrial() throws Exception {
    assertAllowed(TRIAL, false, XPackLicenseState::isMachineLearningAllowed, false);
    assertAllowed(PLATINUM, false, XPackLicenseState::isMachineLearningAllowed, false);
}

public void testLogstashPlatinumGoldTrialStandard() throws Exception {
    assertAllowed(TRIAL, true, XPackLicenseState::isLogstashAllowed, true);
```

```
    assertAllowed(GOLD, true, XPackLicenseState::isLogstashAllowed, true);
    assertAllowed(PLATINUM, true, XPackLicenseState::isLogstashAllowed, true);
    assertAllowed(STANDARD, true, XPackLicenseState::isLogstashAllowed, true);
}

public void testLogstashBasicLicense() throws Exception {
    assertAllowed(BASIC, true, XPackLicenseState::isLogstashAllowed, false);
}

public void testLogstashInactive() {
    assertAllowed(BASIC, false, XPackLicenseState::isLogstashAllowed, false);
    assertAllowed(TRIAL, false, XPackLicenseState::isLogstashAllowed, false);
    assertAllowed(GOLD, false, XPackLicenseState::isLogstashAllowed, false);
    assertAllowed(PLATINUM, false, XPackLicenseState::isLogstashAllowed, false);
    assertAllowed(STANDARD, false, XPackLicenseState::isLogstashAllowed, false);
}

public void testSqlDefaults() {
    XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
    assertThat(licenseState.isSqlAllowed(), is(true));
    assertThat(licenseState.isJdbcAllowed(), is(true));
}

public void testSqlBasic() {
    XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
    licenseState.update(BASIC, true, null);

    assertThat(licenseState.isSqlAllowed(), is(true));
    assertThat(licenseState.isJdbcAllowed(), is(false));
}

public void testSqlBasicExpired() {
    XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
    licenseState.update(BASIC, false, null);

    assertThat(licenseState.isSqlAllowed(), is(false));
    assertThat(licenseState.isJdbcAllowed(), is(false));
}

public void testSqlStandard() {
    XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
    licenseState.update(STANDARD, true, null);

    assertThat(licenseState.isSqlAllowed(), is(true));
    assertThat(licenseState.isJdbcAllowed(), is(false));
}

public void testSqlStandardExpired() {
```

```
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(STANDARD, false, null);

        assertThat(licenseState.isSqlAllowed(), is(false));
        assertThat(licenseState.isJdbcAllowed(), is(false));
    }

    public void testSqlGold() {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(GOLD, true, null);

        assertThat(licenseState.isSqlAllowed(), is(true));
        assertThat(licenseState.isJdbcAllowed(), is(false));
    }

    public void testSqlGoldExpired() {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(GOLD, false, null);

        assertThat(licenseState.isSqlAllowed(), is(false));
        assertThat(licenseState.isJdbcAllowed(), is(false));
    }

    public void testSqlPlatinum() {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(PLATINUM, true, null);

        assertThat(licenseState.isSqlAllowed(), is(true));
        assertThat(licenseState.isJdbcAllowed(), is(true));
    }

    public void testSqlPlatinumExpired() {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        licenseState.update(PLATINUM, false, null);

        assertThat(licenseState.isSqlAllowed(), is(false));
        assertThat(licenseState.isJdbcAllowed(), is(false));
    }

    public void testSqlAckAnyToTrialOrPlatinum() {
        assertAckMesssages(XPackField.SQL, randomMode(), randomTrialOrPlatinumMode(), 0);
    }

    public void testSqlAckTrialOrPlatinumToNotTrialOrPlatinum() {
        assertAckMesssages(XPackField.SQL, randomTrialOrPlatinumMode(), randomBasicStandardOrGold(), 1);
    }

}
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.xpack.core.scheduler.SchedulerEngine;
import org.junit.Before;

import static org.hamcrest.Matchers.equalTo;

public class LicenseScheduleTests extends ESTestCase {

  private License license;
  private SchedulerEngine.Schedule schedule;

  @Before
  public void setuo() throws Exception {
    license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(12));
    schedule = LicenseService.nextLicenseCheck(license);
  }

  public void testEnabledLicenseSchedule() throws Exception {
    int expiryDuration = (int) (license.expiryDate() - license.issueDate());
    long triggeredTime = license.issueDate() + between(0, expiryDuration);
    assertThat(schedule.nextScheduledTimeAfter(license.issueDate(), triggeredTime),
equalTo(license.expiryDate()));
  }

  public void testGraceLicenseSchedule() throws Exception {
    long triggeredTime = license.expiryDate() + between(1,
        ((int) LicenseService.GRACE_PERIOD_DURATION.getMillis()));
    assertThat(schedule.nextScheduledTimeAfter(license.issueDate(), triggeredTime),
        equalTo(license.expiryDate() + LicenseService.GRACE_PERIOD_DURATION.getMillis()));
  }

  public void testExpiredLicenseSchedule() throws Exception {
    long triggeredTime = license.expiryDate() + LicenseService.GRACE_PERIOD_DURATION.getMillis() +
        randomIntBetween(1, 1000);
    assertThat(schedule.nextScheduledTimeAfter(license.issueDate(), triggeredTime),
        equalTo(-1L));
  }

  public void testInvalidLicenseSchedule() throws Exception {
    long triggeredTime = license.issueDate() - randomIntBetween(1, 1000);
```

```
        assertThat(schedule.nextScheduledTimeAfter(triggeredTime, triggeredTime),
                equalTo(license.issueDate()));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.analysis.common.CommonAnalysisPlugin;
import org.elasticsearch.common.network.NetworkModule;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.test.ESIntegTestCase;
import org.elasticsearch.transport.Netty4Plugin;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;

import java.util.Arrays;
import java.util.Collection;

import static org.elasticsearch.test.ESIntegTestCase.Scope.TEST;

@ESIntegTestCase.ClusterScope(scope = TEST, numDataNodes = 0, numClientNodes = 0, maxNumDataNodes =
0, transportClientRatio = 0,
        autoMinMasterNodes = false)
public class LicenseServiceClusterNotRecoveredTests extends AbstractLicensesIntegrationTestCase {

    @Override
    protected Settings nodeSettings(int nodeOrdinal) {
        return nodeSettingsBuilder(nodeOrdinal).build();
    }

    private Settings.Builder nodeSettingsBuilder(int nodeOrdinal) {
        return Settings.builder()
                .put(super.nodeSettings(nodeOrdinal))
                .put("node.data", true)
                .put("resource.reload.interval.high", "500ms") // for license mode file watcher
                .put(NetworkModule.HTTP_ENABLED.getKey(), true);
    }

    @Override
    protected Collection<Class<? extends Plugin>> nodePlugins() {
        return Arrays.asList(LocalStateCompositeXPackPlugin.class, CommonAnalysisPlugin.class,
Netty4Plugin.class);
    }
```

```
    @Override
    protected Collection<Class<? extends Plugin>> transportClientPlugins() {
        return nodePlugins();
    }

    public void testClusterNotRecovered() throws Exception {
        logger.info("--> start one master out of two [recovery state]");
        internalCluster().startNode(nodeSettingsBuilder(0).put("discovery.zen.minimum_master_nodes",
2).put("node.master", true));
        logger.info("--> start second master out of two [recovered state]");
        internalCluster().startNode(nodeSettingsBuilder(1).put("discovery.zen.minimum_master_nodes",
2).put("node.master", true));
        assertLicenseActive(true);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.client.Client;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.util.concurrent.ThreadContext;
import org.elasticsearch.protocol.xpack.XPackInfoResponse;
import org.elasticsearch.protocol.xpack.license.LicenseStatus;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.threadpool.TestThreadPool;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.xpack.core.action.XPackInfoAction;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.concurrent.atomic.AtomicBoolean;
import java.util.concurrent.atomic.AtomicInteger;
import java.util.concurrent.atomic.AtomicReference;
import java.util.function.Consumer;

import static org.hamcrest.Matchers.contains;
import static org.hamcrest.Matchers.containsInAnyOrder;
import static org.hamcrest.Matchers.containsString;
import static org.hamcrest.Matchers.empty;
```

```
import static org.hamcrest.Matchers.equalTo;
import static org.hamcrest.Matchers.hasToString;
import static org.hamcrest.Matchers.instanceOf;
import static org.hamcrest.Matchers.is;
import static org.hamcrest.Matchers.not;
import static org.mockito.Matchers.any;
import static org.mockito.Matchers.anyString;
import static org.mockito.Matchers.argThat;
import static org.mockito.Matchers.same;
import static org.mockito.Mockito.doAnswer;
import static org.mockito.Mockito.mock;
import static org.mockito.Mockito.times;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.when;

public final class RemoteClusterLicenseCheckerTests extends ESTestCase {

  public void testIsNotRemoteIndex() {
    assertFalse(RemoteClusterLicenseChecker.isRemoteIndex("local-index"));
  }

  public void testIsRemoteIndex() {
    assertTrue(RemoteClusterLicenseChecker.isRemoteIndex("remote-cluster:remote-index"));
  }

  public void testNoRemoteIndex() {
    final List<String> indices = Arrays.asList("local-index1", "local-index2");
    assertFalse(RemoteClusterLicenseChecker.containsRemoteIndex(indices));
  }

  public void testRemoteIndex() {
    final List<String> indices = Arrays.asList("local-index", "remote-cluster:remote-index");
    assertTrue(RemoteClusterLicenseChecker.containsRemoteIndex(indices));
  }

  public void testNoRemoteIndices() {
    final List<String> indices = Collections.singletonList("local-index");
    assertThat(RemoteClusterLicenseChecker.remoteIndices(indices), is(empty()));
  }

  public void testRemoteIndices() {
    final List<String> indices = Arrays.asList("local-index1", "remote-cluster1:index1", "local-index2", "remote-
cluster2:index1");
    assertThat(
        RemoteClusterLicenseChecker.remoteIndices(indices),
        containsInAnyOrder("remote-cluster1:index1", "remote-cluster2:index1"));
  }
```

```
  public void testNoRemoteClusterAliases() {
      final List<String> indices = Arrays.asList("local-index1", "local-index2");
      assertThat(RemoteClusterLicenseChecker.remoteClusterAliases(indices), empty());
  }

  public void testOneRemoteClusterAlias() {
      final List<String> indices = Arrays.asList("local-index1", "remote-cluster1:remote-index1");
      assertThat(RemoteClusterLicenseChecker.remoteClusterAliases(indices), contains("remote-cluster1"));
  }

  public void testMoreThanOneRemoteClusterAlias() {
      final List<String> indices = Arrays.asList("remote-cluster1:remote-index1", "local-index1", "remote-
cluster2:remote-index1");
      assertThat(RemoteClusterLicenseChecker.remoteClusterAliases(indices), contains("remote-cluster1", "remote-
cluster2"));
  }

  public void testDuplicateRemoteClusterAlias() {
      final List<String> indices = Arrays.asList(
              "remote-cluster1:remote-index1", "local-index1", "remote-cluster2:index1", "remote-cluster2:remote-
index2");
      assertThat(RemoteClusterLicenseChecker.remoteClusterAliases(indices), contains("remote-cluster1", "remote-
cluster2"));
  }

  public void testCheckRemoteClusterLicensesGivenCompatibleLicenses() {
      final AtomicInteger index = new AtomicInteger();
      final List<XPackInfoResponse> responses = new ArrayList<>();

      final ThreadPool threadPool = createMockThreadPool();
      final Client client = createMockClient(threadPool);
      doAnswer(invocationMock -> {
          @SuppressWarnings("unchecked") ActionListener<XPackInfoResponse> listener =
                  (ActionListener<XPackInfoResponse>) invocationMock.getArguments()[2];
          listener.onResponse(responses.get(index.getAndIncrement()));
          return null;
      }).when(client).execute(same(XPackInfoAction.INSTANCE), any(), any());

      final List<String> remoteClusterAliases = Arrays.asList("valid1", "valid2", "valid3");
      responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
      responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
      responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));

      final RemoteClusterLicenseChecker licenseChecker =
              new RemoteClusterLicenseChecker(client, XPackLicenseState::isPlatinumOrTrialOperationMode);
      final AtomicReference<RemoteClusterLicenseChecker.LicenseCheck> licenseCheck = new
AtomicReference<>();
```

```
        licenseChecker.checkRemoteClusterLicenses(
                remoteClusterAliases,
                doubleInvocationProtectingListener(new ActionListener<RemoteClusterLicenseChecker.LicenseCheck>()
{

                    @Override
                    public void onResponse(final RemoteClusterLicenseChecker.LicenseCheck response) {
                        licenseCheck.set(response);
                    }

                    @Override
                    public void onFailure(final Exception e) {
                        fail(e.getMessage());
                    }

                }));

        verify(client, times(3)).execute(same(XPackInfoAction.INSTANCE), any(), any());
        assertNotNull(licenseCheck.get());
        assertTrue(licenseCheck.get().isSuccess());
    }

    public void testCheckRemoteClusterLicensesGivenIncompatibleLicense() {
        final AtomicInteger index = new AtomicInteger();
        final List<String> remoteClusterAliases = Arrays.asList("good", "cluster-with-basic-license", "good2");
        final List<XPackInfoResponse> responses = new ArrayList<>();
        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
        responses.add(new XPackInfoResponse(null, createBasicLicenseResponse(), null));
        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));

        final ThreadPool threadPool = createMockThreadPool();
        final Client client = createMockClient(threadPool);
        doAnswer(invocationMock -> {
            @SuppressWarnings("unchecked") ActionListener<XPackInfoResponse> listener =
                    (ActionListener<XPackInfoResponse>) invocationMock.getArguments()[2];
            listener.onResponse(responses.get(index.getAndIncrement()));
            return null;
        }).when(client).execute(same(XPackInfoAction.INSTANCE), any(), any());

        final RemoteClusterLicenseChecker licenseChecker =
                new RemoteClusterLicenseChecker(client, XPackLicenseState::isPlatinumOrTrialOperationMode);
        final AtomicReference<RemoteClusterLicenseChecker.LicenseCheck> licenseCheck = new
AtomicReference<>();

        licenseChecker.checkRemoteClusterLicenses(
                remoteClusterAliases,
                doubleInvocationProtectingListener(new ActionListener<RemoteClusterLicenseChecker.LicenseCheck>()
{
```

```
                @Override
                public void onResponse(final RemoteClusterLicenseChecker.LicenseCheck response) {
                    licenseCheck.set(response);
                }

                @Override
                public void onFailure(final Exception e) {
                    fail(e.getMessage());
                }

            }));

        verify(client, times(2)).execute(same(XPackInfoAction.INSTANCE), any(), any());
        assertNotNull(licenseCheck.get());
        assertFalse(licenseCheck.get().isSuccess());
        assertThat(licenseCheck.get().remoteClusterLicenseInfo().clusterAlias(), equalTo("cluster-with-basic-
license"));
        assertThat(licenseCheck.get().remoteClusterLicenseInfo().licenseInfo().getType(), equalTo("BASIC"));
    }

    public void testCheckRemoteClusterLicencesGivenNonExistentCluster() {
        final AtomicInteger index = new AtomicInteger();
        final List<XPackInfoResponse> responses = new ArrayList<>();

        final List<String> remoteClusterAliases = Arrays.asList("valid1", "valid2", "valid3");
        final String failingClusterAlias = randomFrom(remoteClusterAliases);
        final ThreadPool threadPool = createMockThreadPool();
        final Client client = createMockClientThatThrowsOnGetRemoteClusterClient(threadPool, failingClusterAlias);
        doAnswer(invocationMock -> {
            @SuppressWarnings("unchecked") ActionListener<XPackInfoResponse> listener =
                    (ActionListener<XPackInfoResponse>) invocationMock.getArguments()[2];
            listener.onResponse(responses.get(index.getAndIncrement()));
            return null;
        }).when(client).execute(same(XPackInfoAction.INSTANCE), any(), any());

        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));

        final RemoteClusterLicenseChecker licenseChecker =
                new RemoteClusterLicenseChecker(client, XPackLicenseState::isPlatinumOrTrialOperationMode);
        final AtomicReference<Exception> exception = new AtomicReference<>();

        licenseChecker.checkRemoteClusterLicenses(
                remoteClusterAliases,
                doubleInvocationProtectingListener(new ActionListener<RemoteClusterLicenseChecker.LicenseCheck>()
{
```

```
                @Override
                public void onResponse(final RemoteClusterLicenseChecker.LicenseCheck response) {
                    fail();
                }

                @Override
                public void onFailure(final Exception e) {
                    exception.set(e);
                }

            }));

        assertNotNull(exception.get());
        assertThat(exception.get(), instanceOf(ElasticsearchException.class));
        assertThat(exception.get().getMessage(), equalTo("could not determine the license type for cluster [" +
failingClusterAlias + "]"));
        assertNotNull(exception.get().getCause());
        assertThat(exception.get().getCause(), instanceOf(IllegalArgumentException.class));
    }

    public void testRemoteClusterLicenseCallUsesSystemContext() throws InterruptedException {
        final ThreadPool threadPool = new TestThreadPool(getTestName());

        try {
            final Client client = createMockClient(threadPool);
            doAnswer(invocationMock -> {
                assertTrue(threadPool.getThreadContext().isSystemContext());
                @SuppressWarnings("unchecked") ActionListener<XPackInfoResponse> listener =
                        (ActionListener<XPackInfoResponse>) invocationMock.getArguments()[2];
                listener.onResponse(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
                return null;
            }).when(client).execute(same(XPackInfoAction.INSTANCE), any(), any());

            final RemoteClusterLicenseChecker licenseChecker =
                    new RemoteClusterLicenseChecker(client, XPackLicenseState::isPlatinumOrTrialOperationMode);

            final List<String> remoteClusterAliases = Collections.singletonList("valid");
            licenseChecker.checkRemoteClusterLicenses(
                    remoteClusterAliases, doubleInvocationProtectingListener(ActionListener.wrap(() -> {})));

            verify(client, times(1)).execute(same(XPackInfoAction.INSTANCE), any(), any());
        } finally {
            terminate(threadPool);
        }
    }

    public void testListenerIsExecutedWithCallingContext() throws InterruptedException {
```

```
    final AtomicInteger index = new AtomicInteger();
    final List<XPackInfoResponse> responses = new ArrayList<>();

    final ThreadPool threadPool = new TestThreadPool(getTestName());

    try {
        final List<String> remoteClusterAliases = Arrays.asList("valid1", "valid2", "valid3");
        final Client client;
        final boolean failure = randomBoolean();
        if (failure) {
            client = createMockClientThatThrowsOnGetRemoteClusterClient(threadPool,
randomFrom(remoteClusterAliases));
        } else {
            client = createMockClient(threadPool);
        }
        doAnswer(invocationMock -> {
            @SuppressWarnings("unchecked") ActionListener<XPackInfoResponse> listener =
                    (ActionListener<XPackInfoResponse>) invocationMock.getArguments()[2];
            listener.onResponse(responses.get(index.getAndIncrement()));
            return null;
        }).when(client).execute(same(XPackInfoAction.INSTANCE), any(), any());

        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));
        responses.add(new XPackInfoResponse(null, createPlatinumLicenseResponse(), null));

        final RemoteClusterLicenseChecker licenseChecker =
                new RemoteClusterLicenseChecker(client, XPackLicenseState::isPlatinumOrTrialOperationMode);

        final AtomicBoolean listenerInvoked = new AtomicBoolean();
        threadPool.getThreadContext().putHeader("key", "value");
        licenseChecker.checkRemoteClusterLicenses(
                remoteClusterAliases,
                doubleInvocationProtectingListener(new
ActionListener<RemoteClusterLicenseChecker.LicenseCheck>() {

                    @Override
                    public void onResponse(final RemoteClusterLicenseChecker.LicenseCheck response) {
                        if (failure) {
                            fail();
                        }
                        assertThat(threadPool.getThreadContext().getHeader("key"), equalTo("value"));
                        assertFalse(threadPool.getThreadContext().isSystemContext());
                        listenerInvoked.set(true);
                    }

                    @Override
                    public void onFailure(final Exception e) {
```

```
                if (failure == false) {
                    fail();
                }
                assertThat(threadPool.getThreadContext().getHeader("key"), equalTo("value"));
                assertFalse(threadPool.getThreadContext().isSystemContext());
                listenerInvoked.set(true);
            }

        }));

        assertTrue(listenerInvoked.get());
    } finally {
        terminate(threadPool);
    }
}

public void testBuildErrorMessageForActiveCompatibleLicense() {
    final XPackInfoResponse.LicenseInfo platinumLicence = createPlatinumLicenseResponse();
    final RemoteClusterLicenseChecker.RemoteClusterLicenseInfo info =
            new RemoteClusterLicenseChecker.RemoteClusterLicenseInfo("platinum-cluster", platinumLicence);
    final AssertionError e = expectThrows(
            AssertionError.class,
            () -> RemoteClusterLicenseChecker.buildErrorMessage("", info,
RemoteClusterLicenseChecker::isLicensePlatinumOrTrial));
    assertThat(e, hasToString(containsString("license must be incompatible to build error message")));
}

public void testBuildErrorMessageForIncompatibleLicense() {
    final XPackInfoResponse.LicenseInfo basicLicense = createBasicLicenseResponse();
    final RemoteClusterLicenseChecker.RemoteClusterLicenseInfo info =
            new RemoteClusterLicenseChecker.RemoteClusterLicenseInfo("basic-cluster", basicLicense);
    assertThat(
            RemoteClusterLicenseChecker.buildErrorMessage("Feature", info,
RemoteClusterLicenseChecker::isLicensePlatinumOrTrial),
            equalTo("the license mode [BASIC] on cluster [basic-cluster] does not enable [Feature]"));
}

public void testBuildErrorMessageForInactiveLicense() {
    final XPackInfoResponse.LicenseInfo expiredLicense = createExpiredLicenseResponse();
    final RemoteClusterLicenseChecker.RemoteClusterLicenseInfo info =
            new RemoteClusterLicenseChecker.RemoteClusterLicenseInfo("expired-cluster", expiredLicense);
    assertThat(
            RemoteClusterLicenseChecker.buildErrorMessage("Feature", info,
RemoteClusterLicenseChecker::isLicensePlatinumOrTrial),
            equalTo("the license on cluster [expired-cluster] is not active"));
}

private ActionListener<RemoteClusterLicenseChecker.LicenseCheck> doubleInvocationProtectingListener(
```

```
            final ActionListener<RemoteClusterLicenseChecker.LicenseCheck> listener) {
        final AtomicBoolean listenerInvoked = new AtomicBoolean();
        return new ActionListener<RemoteClusterLicenseChecker.LicenseCheck>() {

            @Override
            public void onResponse(final RemoteClusterLicenseChecker.LicenseCheck response) {
                if (listenerInvoked.compareAndSet(false, true) == false) {
                    fail("listener invoked twice");
                }
                listener.onResponse(response);
            }

            @Override
            public void onFailure(final Exception e) {
                if (listenerInvoked.compareAndSet(false, true) == false) {
                    fail("listener invoked twice");
                }
                listener.onFailure(e);
            }

        };
    }

    private ThreadPool createMockThreadPool() {
        final ThreadPool threadPool = mock(ThreadPool.class);
        when(threadPool.getThreadContext()).thenReturn(new ThreadContext(Settings.EMPTY));
        return threadPool;
    }

    private Client createMockClient(final ThreadPool threadPool) {
        return createMockClient(threadPool, client ->
when(client.getRemoteClusterClient(anyString())).thenReturn(client));
    }

    private Client createMockClientThatThrowsOnGetRemoteClusterClient(final ThreadPool threadPool, final String
clusterAlias) {
        return createMockClient(
                threadPool,
                client -> {
                    when(client.getRemoteClusterClient(clusterAlias)).thenThrow(new IllegalArgumentException());
                    when(client.getRemoteClusterClient(argThat(not(clusterAlias)))).thenReturn(client);
                });
    }

    private Client createMockClient(final ThreadPool threadPool, final Consumer<Client> finish) {
        final Client client = mock(Client.class);
        when(client.threadPool()).thenReturn(threadPool);
        finish.accept(client);
```

```
        return client;
    }

    private XPackInfoResponse.LicenseInfo createPlatinumLicenseResponse() {
        return new XPackInfoResponse.LicenseInfo("uid", "PLATINUM", "PLATINUM", LicenseStatus.ACTIVE,
randomNonNegativeLong());
    }

    private XPackInfoResponse.LicenseInfo createBasicLicenseResponse() {
        return new XPackInfoResponse.LicenseInfo("uid", "BASIC", "BASIC", LicenseStatus.ACTIVE,
randomNonNegativeLong());
    }

    private XPackInfoResponse.LicenseInfo createExpiredLicenseResponse() {
        return new XPackInfoResponse.LicenseInfo("uid", "PLATINUM", "PLATINUM", LicenseStatus.EXPIRED,
randomNonNegativeLong());
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionFuture;
import org.elasticsearch.action.support.master.AcknowledgedResponse;
import org.elasticsearch.common.bytes.BytesArray;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.node.Node;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.protocol.xpack.license.LicensesStatus;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;
import org.elasticsearch.test.ESSingleNodeTestCase;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackSettings;

import java.nio.charset.StandardCharsets;
import java.util.Collection;
import java.util.Collections;
import java.util.UUID;

import static org.elasticsearch.license.TestUtils.dateMath;
import static org.elasticsearch.license.TestUtils.generateExpiredNonBasicLicense;
import static org.elasticsearch.license.TestUtils.generateSignedLicense;
```

```
import static org.hamcrest.CoreMatchers.equalTo;
import static org.hamcrest.CoreMatchers.not;

public class LicensesTransportTests extends ESSingleNodeTestCase {

  @Override
  protected boolean resetNodeAfterTest() {
    return true;
  }

  @Override
  protected Collection<Class<? extends Plugin>> getPlugins() {
    return Collections.singletonList(LocalStateCompositeXPackPlugin.class);
  }

  @Override
  protected Settings nodeSettings() {
    Settings.Builder newSettings = Settings.builder();
    newSettings.put(super.nodeSettings());
    newSettings.put(XPackSettings.SECURITY_ENABLED.getKey(), false);
//    newSettings.put(XPackSettings.MONITORING_ENABLED.getKey(), false);
//    newSettings.put(XPackSettings.WATCHER_ENABLED.getKey(), false);
    newSettings.put(Node.NODE_DATA_SETTING.getKey(), true);
    return newSettings.build();
  }

  public void testEmptyGetLicense() throws Exception {
    // basic license is added async, we should wait for it
    assertBusy(() -> {
      try {
        final ActionFuture<GetLicenseResponse> getLicenseFuture =
            new GetLicenseRequestBuilder(client().admin().cluster(), GetLicenseAction.INSTANCE).execute();
        final GetLicenseResponse getLicenseResponse;
        getLicenseResponse = getLicenseFuture.get();
        assertNotNull(getLicenseResponse.license());
        assertThat(getLicenseResponse.license().operationMode(), equalTo(License.OperationMode.BASIC));
      } catch (Exception e) {
        throw new RuntimeException("unexpected exception", e);
      }
    });
  }

  public void testPutLicense() throws Exception {
    License signedLicense = generateSignedLicense(TimeValue.timeValueMinutes(2));

    // put license
    PutLicenseRequestBuilder putLicenseRequestBuilder =
        new PutLicenseRequestBuilder(client().admin().cluster(),
```

```
PutLicenseAction.INSTANCE).setLicense(signedLicense)
                .setAcknowledge(true);
    PutLicenseResponse putLicenseResponse = putLicenseRequestBuilder.get();
    assertThat(putLicenseResponse.isAcknowledged(), equalTo(true));
    assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.VALID));

    // get and check license
    GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
    assertThat(getLicenseResponse.license(), equalTo(signedLicense));
  }

  public void testPutLicenseFromString() throws Exception {
    License signedLicense = generateSignedLicense(TimeValue.timeValueMinutes(2));
    String licenseString = TestUtils.dumpLicense(signedLicense);

    // put license source
    PutLicenseRequestBuilder putLicenseRequestBuilder =
        new PutLicenseRequestBuilder(client().admin().cluster(), PutLicenseAction.INSTANCE)
                .setLicense(new BytesArray(licenseString.getBytes(StandardCharsets.UTF_8)),
XContentType.JSON)
                .setAcknowledge(true);
    PutLicenseResponse putLicenseResponse = putLicenseRequestBuilder.get();
    assertThat(putLicenseResponse.isAcknowledged(), equalTo(true));
    assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.VALID));

    // get and check license
    GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
    assertThat(getLicenseResponse.license(), equalTo(signedLicense));
  }

  public void testPutInvalidLicense() throws Exception {
    License signedLicense = generateSignedLicense(TimeValue.timeValueMinutes(2));

    // modify content of signed license
    License tamperedLicense = License.builder()
        .fromLicenseSpec(signedLicense, signedLicense.signature())
        .expiryDate(signedLicense.expiryDate() + 10 * 24 * 60 * 60 * 1000L)
        .validate()
        .build();

    PutLicenseRequestBuilder builder = new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE);
    builder.setLicense(tamperedLicense);

    // try to put license (should be invalid)
    final PutLicenseResponse putLicenseResponse = builder.get();
```

```
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.INVALID));

        // try to get invalid license
        GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
        assertThat(getLicenseResponse.license(), not(tamperedLicense));
    }

    public void testPutBasicLicenseIsInvalid() throws Exception {
        License signedLicense = generateSignedLicense("basic", License.VERSION_CURRENT, -1,
TimeValue.timeValueMinutes(2));

        PutLicenseRequestBuilder builder = new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE);
        builder.setLicense(signedLicense);

        // try to put license (should be invalid)
        IllegalArgumentException iae = expectThrows(IllegalArgumentException.class, builder::get);
        assertEquals(iae.getMessage(), "Registering basic licenses is not allowed.");

        // try to get invalid license
        GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
        assertThat(getLicenseResponse.license(), not(signedLicense));
    }

    public void testPutExpiredLicense() throws Exception {
        License expiredLicense = generateExpiredNonBasicLicense();
        PutLicenseRequestBuilder builder = new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE);
        builder.setLicense(expiredLicense);
        PutLicenseResponse putLicenseResponse = builder.get();
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.EXPIRED));
        // get license should not return the expired license
        GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
        assertThat(getLicenseResponse.license(), not(expiredLicense));
    }

    public void testPutLicensesSimple() throws Exception {
        License goldSignedLicense = generateSignedLicense("gold", TimeValue.timeValueMinutes(5));
        PutLicenseRequestBuilder putLicenseRequestBuilder =
                new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE).setLicense(goldSignedLicense)
                        .setAcknowledge(true);
        PutLicenseResponse putLicenseResponse = putLicenseRequestBuilder.get();
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.VALID));
        GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
```

```
GetLicenseAction.INSTANCE).get();
        assertThat(getLicenseResponse.license(), equalTo(goldSignedLicense));

        License platinumSignedLicense = generateSignedLicense("platinum", TimeValue.timeValueMinutes(2));
        putLicenseRequestBuilder.setLicense(platinumSignedLicense);
        putLicenseResponse = putLicenseRequestBuilder.get();
        assertThat(putLicenseResponse.isAcknowledged(), equalTo(true));
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.VALID));
        getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
        assertThat(getLicenseResponse.license(), equalTo(platinumSignedLicense));
    }

    public void testRemoveLicensesSimple() throws Exception {
        License goldLicense = generateSignedLicense("gold", TimeValue.timeValueMinutes(5));
        PutLicenseRequestBuilder putLicenseRequestBuilder =
                new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE).setLicense(goldLicense)
                        .setAcknowledge(true);
        PutLicenseResponse putLicenseResponse = putLicenseRequestBuilder.get();
        assertThat(putLicenseResponse.isAcknowledged(), equalTo(true));
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.VALID));
        GetLicenseResponse getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
        assertThat(getLicenseResponse.license(), equalTo(goldLicense));
        // delete all licenses
        DeleteLicenseRequestBuilder deleteLicenseRequestBuilder =
                new DeleteLicenseRequestBuilder(client().admin().cluster(), DeleteLicenseAction.INSTANCE);
        AcknowledgedResponse deleteLicenseResponse = deleteLicenseRequestBuilder.get();
        assertThat(deleteLicenseResponse.isAcknowledged(), equalTo(true));
        // get licenses (expected no licenses)
        getLicenseResponse = new GetLicenseRequestBuilder(client().admin().cluster(),
GetLicenseAction.INSTANCE).get();
        assertNull(getLicenseResponse.license());
    }

    public void testLicenseIsRejectWhenStartDateLaterThanNow() throws Exception {
        long now = System.currentTimeMillis();
        final License.Builder builder = License.builder()
                .uid(UUID.randomUUID().toString())
                .version(License.VERSION_CURRENT)
                .expiryDate(dateMath("now+2h", now))
                .startDate(dateMath("now+1h", now))
                .issueDate(now)
                .type(License.OperationMode.TRIAL.toString())
                .issuedTo("customer")
                .issuer("elasticsearch")
                .maxNodes(5);
```

```
        License license = TestUtils.generateSignedLicense(builder);

        PutLicenseRequestBuilder putLicenseRequestBuilder =
                new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE).setLicense(license)
                        .setAcknowledge(true);
        PutLicenseResponse putLicenseResponse = putLicenseRequestBuilder.get();
        assertThat(putLicenseResponse.isAcknowledged(), equalTo(true));
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.INVALID));
    }

    public void testLicenseIsAcceptedWhenStartDateBeforeThanNow() throws Exception {
        long now = System.currentTimeMillis();
        final License.Builder builder = License.builder()
                .uid(UUID.randomUUID().toString())
                .version(License.VERSION_CURRENT)
                .expiryDate(dateMath("now+2h", now))
                .startDate(now)
                .issueDate(now)
                .type(License.OperationMode.TRIAL.toString())
                .issuedTo("customer")
                .issuer("elasticsearch")
                .maxNodes(5);
        License license = TestUtils.generateSignedLicense(builder);

        PutLicenseRequestBuilder putLicenseRequestBuilder =
                new PutLicenseRequestBuilder(client().admin().cluster(),
PutLicenseAction.INSTANCE).setLicense(license)
                        .setAcknowledge(true);
        PutLicenseResponse putLicenseResponse = putLicenseRequestBuilder.get();
        assertThat(putLicenseResponse.isAcknowledged(), equalTo(true));
        assertThat(putLicenseResponse.status(), equalTo(LicensesStatus.VALID));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.action.support.PlainActionFuture;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.node.DiscoveryNode;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.transport.TransportAddress;
import org.elasticsearch.common.unit.TimeValue;
```

```
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;

import java.net.InetAddress;

import static java.util.Collections.emptyMap;
import static java.util.Collections.emptySet;
import static org.hamcrest.Matchers.containsString;
import static org.mockito.Matchers.any;
import static org.mockito.Mockito.times;
import static org.mockito.Mockito.verify;

public class LicenseTLSTests extends AbstractLicenseServiceTestCase {

  private InetAddress inetAddress;

  public void testApplyLicenseInDevMode() throws Exception {
    License newLicense = TestUtils.generateSignedLicense(randomFrom("gold", "platinum"),
TimeValue.timeValueHours(24L));
    PutLicenseRequest request = new PutLicenseRequest();
    request.acknowledge(true);
    request.license(newLicense);
    Settings settings = Settings.builder().put("xpack.security.enabled", true).build();
    XPackLicenseState licenseState = new XPackLicenseState(settings);
    inetAddress = InetAddress.getLoopbackAddress();

    setInitialState(null, licenseState, settings);
    licenseService.start();
    PlainActionFuture<PutLicenseResponse> responseFuture = new PlainActionFuture<>();
    licenseService.registerLicense(request, responseFuture);
    verify(clusterService).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));

    inetAddress = TransportAddress.META_ADDRESS;
    settings = Settings.builder()
        .put("xpack.security.enabled", true)
        .put("discovery.type", "single-node")
        .build();
    licenseService.stop();
    licenseState = new XPackLicenseState(settings);
    setInitialState(null, licenseState, settings);
    licenseService.start();
    licenseService.registerLicense(request, responseFuture);
    verify(clusterService, times(2)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
  }

  public void testApplyLicenseInProdMode() throws Exception {
    final String licenseType = randomFrom("GOLD", "PLATINUM");
    License newLicense = TestUtils.generateSignedLicense(licenseType, TimeValue.timeValueHours(24L));
    PutLicenseRequest request = new PutLicenseRequest();
```

```
        request.acknowledge(true);
        request.license(newLicense);
        Settings settings = Settings.builder().put("xpack.security.enabled", true).build();
        XPackLicenseState licenseState = new XPackLicenseState(settings);
        inetAddress = TransportAddress.META_ADDRESS;

        setInitialState(null, licenseState, settings);
        licenseService.start();
        PlainActionFuture<PutLicenseResponse> responseFuture = new PlainActionFuture<>();
        IllegalStateException e = expectThrows(IllegalStateException.class, () ->
licenseService.registerLicense(request, responseFuture));
        assertThat(e.getMessage(),
                containsString("Cannot install a [" + licenseType + "] license unless TLS is configured or security is
disabled"));

        settings = Settings.builder().put("xpack.security.enabled", false).build();
        licenseService.stop();
        licenseState = new XPackLicenseState(settings);
        setInitialState(null, licenseState, settings);
        licenseService.start();
        licenseService.registerLicense(request, responseFuture);
        verify(clusterService).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));

        settings = Settings.builder()
                .put("xpack.security.enabled", true)
                .put("xpack.security.transport.ssl.enabled", true)
                .build();
        licenseService.stop();
        licenseState = new XPackLicenseState(settings);
        setInitialState(null, licenseState, settings);
        licenseService.start();
        licenseService.registerLicense(request, responseFuture);
        verify(clusterService, times(2)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
    }

    @Override
    protected DiscoveryNode getLocalNode() {
        return new DiscoveryNode("localnode", new TransportAddress(inetAddress, randomIntBetween(9300, 9399)),
                emptyMap(), emptySet(), Version.CURRENT);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import org.elasticsearch.Version;
import org.elasticsearch.cluster.ClusterName;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlocks;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.node.DiscoveryNode;
import org.elasticsearch.cluster.node.DiscoveryNodes;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.component.Lifecycle;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.env.Environment;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.watcher.ResourceWatcherService;
import org.elasticsearch.xpack.core.XPackPlugin;
import org.elasticsearch.xpack.core.watcher.watch.ClockMock;
import org.junit.After;
import org.junit.Before;

import java.nio.file.Path;
import java.util.Arrays;

import static java.util.Collections.emptySet;
import static java.util.Collections.singletonMap;
import static org.mockito.Mockito.mock;
import static org.mockito.Mockito.when;

public abstract class AbstractLicenseServiceTestCase extends ESTestCase {

  protected LicenseService licenseService;
  protected ClusterService clusterService;
  protected ResourceWatcherService resourceWatcherService;
  protected ClockMock clock;
  protected DiscoveryNodes discoveryNodes;
  protected Environment environment;
  protected String licenseType;

  @Before
  public void init() throws Exception {
    clusterService = mock(ClusterService.class);
    clock = ClockMock.frozen();
    discoveryNodes = mock(DiscoveryNodes.class);
    resourceWatcherService = mock(ResourceWatcherService.class);
    environment = mock(Environment.class);
  }

  protected void setInitialState(License license, XPackLicenseState licenseState, Settings settings) {
    setInitialState(license, licenseState, settings, randomBoolean() ? "trial" : "basic");
  }
```

```
  protected void setInitialState(License license, XPackLicenseState licenseState, Settings settings, String
selfGeneratedType) {
      Path tempDir = createTempDir();
      when(environment.configFile()).thenReturn(tempDir);
      licenseType = selfGeneratedType;
      settings =
Settings.builder().put(settings).put(LicenseService.SELF_GENERATED_LICENSE_TYPE.getKey(),
licenseType).build();
      licenseService = new LicenseService(settings, clusterService, clock, environment, resourceWatcherService,
licenseState);
      ClusterState state = mock(ClusterState.class);
      final ClusterBlocks noBlock = ClusterBlocks.builder().build();
      when(state.blocks()).thenReturn(noBlock);
      MetaData metaData = mock(MetaData.class);
      when(metaData.custom(LicensesMetaData.TYPE)).thenReturn(new LicensesMetaData(license, null));
      when(state.metaData()).thenReturn(metaData);
      final DiscoveryNode mockNode = getLocalNode();
      when(discoveryNodes.getMasterNode()).thenReturn(mockNode);
      when(discoveryNodes.spliterator()).thenReturn(Arrays.asList(mockNode).spliterator());
      when(discoveryNodes.isLocalNodeElectedMaster()).thenReturn(false);
      when(state.nodes()).thenReturn(discoveryNodes);
      when(state.getNodes()).thenReturn(discoveryNodes); // it is really ridiculous we have nodes() and getNodes()...
      when(clusterService.state()).thenReturn(state);
      when(clusterService.lifecycleState()).thenReturn(Lifecycle.State.STARTED);
      when(clusterService.getClusterName()).thenReturn(new ClusterName("a"));
      when(clusterService.localNode()).thenReturn(mockNode);
  }

  protected DiscoveryNode getLocalNode() {
      return new DiscoveryNode("b", buildNewFakeTransportAddress(),
singletonMap(XPackPlugin.XPACK_INSTALLED_NODE_ATTR, "true"),
          emptySet(), Version.CURRENT);
  }

  @After
  public void tearDown() throws Exception {
      super.tearDown();
      licenseService.stop();
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.test.ESTestCase;

import java.io.IOException;
import java.nio.ByteBuffer;
import java.util.Base64;
import java.util.Collections;
import java.util.UUID;

import static org.elasticsearch.license.CryptUtils.encryptV3Format;
import static org.hamcrest.Matchers.equalTo;


public class SelfGeneratedLicenseTests extends ESTestCase {

  public void testBasic() throws Exception {
    long issueDate = System.currentTimeMillis();
    License.Builder specBuilder = License.builder()
        .uid(UUID.randomUUID().toString())
        .issuedTo("customer")
        .maxNodes(5)
        .type(randomBoolean() ? "trial" : "basic")
        .issueDate(issueDate)
        .expiryDate(issueDate + TimeValue.timeValueHours(2).getMillis());
    License trialLicense = SelfGeneratedLicense.create(specBuilder, License.VERSION_CURRENT);
    assertThat(SelfGeneratedLicense.verify(trialLicense), equalTo(true));
  }

  public void testTampered() throws Exception {
    long issueDate = System.currentTimeMillis();
    License.Builder specBuilder = License.builder()
        .uid(UUID.randomUUID().toString())
        .issuedTo("customer")
        .type(randomBoolean() ? "trial" : "basic")
        .maxNodes(5)
        .issueDate(issueDate)
        .expiryDate(issueDate + TimeValue.timeValueHours(2).getMillis());
    License trialLicense = SelfGeneratedLicense.create(specBuilder, License.VERSION_CURRENT);
    final String originalSignature = trialLicense.signature();
    License tamperedLicense = License.builder().fromLicenseSpec(trialLicense, originalSignature)
        .expiryDate(System.currentTimeMillis() + TimeValue.timeValueHours(5).getMillis())
        .build();
    assertThat(SelfGeneratedLicense.verify(trialLicense), equalTo(true));
```

```
        assertThat(SelfGeneratedLicense.verify(tamperedLicense), equalTo(false));
    }

    public void testFrom1x() throws Exception {
        long issueDate = System.currentTimeMillis();
        License.Builder specBuilder = License.builder()
                .uid(UUID.randomUUID().toString())
                .issuedTo("customer")
                .type("subscription")
                .subscriptionType("trial")
                .issuer("elasticsearch")
                .feature("")
                .version(License.VERSION_START)
                .maxNodes(5)
                .issueDate(issueDate)
                .expiryDate(issueDate + TimeValue.timeValueHours(2).getMillis());
        License pre20TrialLicense = specBuilder.build();
        License license =
SelfGeneratedLicense.create(License.builder().fromPre20LicenseSpec(pre20TrialLicense).type("trial"),
            License.VERSION_CURRENT);
        assertThat(SelfGeneratedLicense.verify(license), equalTo(true));
    }

    public void testTrialLicenseVerifyWithOlderVersion() throws Exception {
        assumeFalse("Can't run in a FIPS JVM. We can't generate old licenses since PBEWithSHA1AndDESede is not
available", inFipsJvm());
        long issueDate = System.currentTimeMillis();
        License.Builder specBuilder = License.builder()
                .issuedTo("customer")
                .maxNodes(5)
                .issueDate(issueDate)
                .expiryDate(issueDate + TimeValue.timeValueHours(2).getMillis())
                .feature("")
                .subscriptionType("trial")
                .version(1);
        License trialLicenseV1 = createTrialLicense(specBuilder);
        assertThat(SelfGeneratedLicense.verify(trialLicenseV1), equalTo(true));
    }

    private static License createTrialLicense(License.Builder specBuilder) {
        License spec = specBuilder
                .type(randomBoolean() ? "trial" : "basic")
                .issuer("elasticsearch")
                .uid(UUID.randomUUID().toString())
                .build();
        final String signature;
        try {
            XContentBuilder contentBuilder = XContentFactory.contentBuilder(XContentType.JSON);
```

```
        spec.toXContent(contentBuilder, new
ToXContent.MapParams(Collections.singletonMap(License.LICENSE_SPEC_VIEW_MODE, "true")));
        byte[] encrypt = encryptV3Format(BytesReference.toBytes(BytesReference.bytes(contentBuilder)));
        byte[] bytes = new byte[4 + 4 + encrypt.length];
        ByteBuffer byteBuffer = ByteBuffer.wrap(bytes);
        byteBuffer.putInt(-spec.version())
                .putInt(encrypt.length)
                .put(encrypt);
        signature = Base64.getEncoder().encodeToString(bytes);
    } catch (IOException e) {
        throw new IllegalStateException(e);
    }
    return License.builder().fromLicenseSpec(spec, signature).build();
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.cluster.ClusterName;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.settings.Settings;
import org.mockito.ArgumentCaptor;
import org.mockito.Mockito;

import java.util.UUID;

import static org.elasticsearch.license.TestUtils.dateMath;
import static org.mockito.Matchers.any;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.when;

public class LicenseRegistrationTests extends AbstractLicenseServiceTestCase {

  public void testSelfGeneratedTrialLicense() throws Exception {
    XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
    setInitialState(null, licenseState, Settings.EMPTY, "trial");
    when(discoveryNodes.isLocalNodeElectedMaster()).thenReturn(true);
    licenseService.start();

    ClusterState state = ClusterState.builder(new ClusterName("a")).build();
    ArgumentCaptor<ClusterStateUpdateTask> stateUpdater =
ArgumentCaptor.forClass(ClusterStateUpdateTask.class);
```

```
        verify(clusterService, Mockito.times(1)).submitStateUpdateTask(any(), stateUpdater.capture());
        ClusterState stateWithLicense = stateUpdater.getValue().execute(state);
        LicensesMetaData licenseMetaData = stateWithLicense.metaData().custom(LicensesMetaData.TYPE);
        assertNotNull(licenseMetaData);
        assertNotNull(licenseMetaData.getLicense());
        assertFalse(licenseMetaData.isEligibleForTrial());
        assertEquals("trial", licenseMetaData.getLicense().type());
        assertEquals(clock.millis() +
LicenseService.NON_BASIC_SELF_GENERATED_LICENSE_DURATION.millis(),
                licenseMetaData.getLicense().expiryDate());
    }

    public void testSelfGeneratedBasicLicense() throws Exception {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        setInitialState(null, licenseState, Settings.EMPTY, "basic");
        when(discoveryNodes.isLocalNodeElectedMaster()).thenReturn(true);
        licenseService.start();

        ClusterState state = ClusterState.builder(new ClusterName("a")).build();
        ArgumentCaptor<ClusterStateUpdateTask> stateUpdater =
ArgumentCaptor.forClass(ClusterStateUpdateTask.class);
        verify(clusterService, Mockito.times(1)).submitStateUpdateTask(any(), stateUpdater.capture());
        ClusterState stateWithLicense = stateUpdater.getValue().execute(state);
        LicensesMetaData licenseMetaData = stateWithLicense.metaData().custom(LicensesMetaData.TYPE);
        assertNotNull(licenseMetaData);
        assertNotNull(licenseMetaData.getLicense());
        assertTrue(licenseMetaData.isEligibleForTrial());
        assertEquals("basic", licenseMetaData.getLicense().type());
        assertEquals(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS,
licenseMetaData.getLicense().expiryDate());
    }

    public void testNonSelfGeneratedBasicLicenseIsReplaced() throws Exception {
        long now = System.currentTimeMillis();
        String uid = UUID.randomUUID().toString();
        final License.Builder builder = License.builder()
                .uid(uid)
                .version(License.VERSION_CURRENT)
                .expiryDate(dateMath("now+2h", now))
                .startDate(now)
                .issueDate(now)
                .type("basic")
                .issuedTo("customer")
                .issuer("elasticsearch")
                .maxNodes(5);
        License license = TestUtils.generateSignedLicense(builder);

        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
```

```
        setInitialState(license, licenseState, Settings.EMPTY);
        when(discoveryNodes.isLocalNodeElectedMaster()).thenReturn(true);
        licenseService.start();

        MetaData.Builder mdBuilder = MetaData.builder();
        mdBuilder.putCustom(LicensesMetaData.TYPE, new LicensesMetaData(license, null));
        ClusterState state = ClusterState.builder(new ClusterName("a")).metaData(mdBuilder.build()).build();
        ArgumentCaptor<ClusterStateUpdateTask> stateUpdater =
ArgumentCaptor.forClass(ClusterStateUpdateTask.class);
        verify(clusterService, Mockito.times(1)).submitStateUpdateTask(any(), stateUpdater.capture());
        ClusterState stateWithLicense = stateUpdater.getValue().execute(state);
        LicensesMetaData licenseMetaData = stateWithLicense.metaData().custom(LicensesMetaData.TYPE);
        assertNotNull(licenseMetaData);
        assertNotNull(licenseMetaData.getLicense());
        assertTrue(licenseMetaData.isEligibleForTrial());
        assertEquals("basic", licenseMetaData.getLicense().type());
        assertEquals(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS,
licenseMetaData.getLicense().expiryDate());
        assertEquals(uid, licenseMetaData.getLicense().uid());
    }

    public void testExpiredSelfGeneratedBasicLicenseIsExtended() throws Exception {
        long now = System.currentTimeMillis();
        String uid = UUID.randomUUID().toString();
        License.Builder builder = License.builder()
                .uid(uid)
                .issuedTo("name")
                .maxNodes(1000)
                .issueDate(dateMath("now-10h", now))
                .type("basic")
                .expiryDate(dateMath("now-2h", now));
        License license = SelfGeneratedLicense.create(builder, License.VERSION_CURRENT);

        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        setInitialState(license, licenseState, Settings.EMPTY);
        when(discoveryNodes.isLocalNodeElectedMaster()).thenReturn(true);
        licenseService.start();

        MetaData.Builder mdBuilder = MetaData.builder();
        mdBuilder.putCustom(LicensesMetaData.TYPE, new LicensesMetaData(license, null));
        ClusterState state = ClusterState.builder(new ClusterName("a")).metaData(mdBuilder.build()).build();
        ArgumentCaptor<ClusterStateUpdateTask> stateUpdater =
ArgumentCaptor.forClass(ClusterStateUpdateTask.class);
        verify(clusterService, Mockito.times(1)).submitStateUpdateTask(any(), stateUpdater.capture());
        ClusterState stateWithLicense = stateUpdater.getValue().execute(state);
        LicensesMetaData licenseMetaData = stateWithLicense.metaData().custom(LicensesMetaData.TYPE);
        assertNotNull(licenseMetaData);
        assertNotNull(licenseMetaData.getLicense());
```

```
        assertTrue(licenseMetaData.isEligibleForTrial());
        assertEquals("basic", licenseMetaData.getLicense().type());
        assertEquals(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS,
licenseMetaData.getLicense().expiryDate());
        assertEquals(uid, licenseMetaData.getLicense().uid());
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import com.carrotsearch.randomizedtesting.RandomizedTest;
import org.elasticsearch.Version;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.joda.FormatDateTimeFormatter;
import org.elasticsearch.common.joda.Joda;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.time.DateMathParser;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.license.licensor.LicenseSigner;
import org.elasticsearch.protocol.xpack.license.LicensesStatus;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;
import org.hamcrest.MatcherAssert;
import org.joda.time.format.DateTimeFormatter;
import org.junit.Assert;

import java.io.IOException;
import java.io.InputStream;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.StandardCopyOption;
import java.util.ArrayList;
import java.util.List;
import java.util.UUID;
import java.util.concurrent.CountDownLatch;
import java.util.concurrent.atomic.AtomicReference;

import static com.carrotsearch.randomizedtesting.RandomizedTest.randomBoolean;
import static com.carrotsearch.randomizedtesting.RandomizedTest.randomInt;
```

```
import static org.apache.lucene.util.LuceneTestCase.createTempFile;
import static org.elasticsearch.common.xcontent.XContentFactory.jsonBuilder;
import static org.elasticsearch.test.ESTestCase.randomAlphaOfLength;
import static org.elasticsearch.test.ESTestCase.randomFrom;
import static org.elasticsearch.test.ESTestCase.randomIntBetween;
import static org.hamcrest.core.IsEqual.equalTo;
import static org.junit.Assert.assertThat;

public class TestUtils {

  private static final FormatDateTimeFormatter formatDateTimeFormatter = Joda.forPattern("yyyy-MM-dd");
  private static final DateMathParser dateMathParser = formatDateTimeFormatter.toDateMathParser();
  private static final DateTimeFormatter dateTimeFormatter = formatDateTimeFormatter.printer();

  public static String dateMathString(String time, final long now) {
    return dateTimeFormatter.print(dateMathParser.parse(time, () -> now));
  }

  public static long dateMath(String time, final long now) {
    return dateMathParser.parse(time, () -> now);
  }

  public static LicenseSpec generateRandomLicenseSpec(int version) {
    boolean datesInMillis = randomBoolean();
    long now = System.currentTimeMillis();
    String uid = UUID.randomUUID().toString();
    String feature = "feature__" + randomInt();
    String issuer = "issuer__"  + randomInt();
    String issuedTo = "issuedTo__" + randomInt();
    final String type;
    final String subscriptionType;
    if (version < License.VERSION_NO_FEATURE_TYPE) {
      subscriptionType = randomFrom("gold", "silver", "platinum");
      type = "subscription";//randomFrom("subscription", "internal", "development");
    } else {
      subscriptionType = null;
      type = randomFrom("basic", "dev", "gold", "silver", "platinum");
    }
    int maxNodes = RandomizedTest.randomIntBetween(5, 100);
    if (datesInMillis) {
      long issueDateInMillis = dateMath("now", now);
      long expiryDateInMillis = dateMath("now+10d/d", now);
      return new LicenseSpec(version, uid, feature, issueDateInMillis, expiryDateInMillis, type, subscriptionType,
issuedTo, issuer,
          maxNodes);
    } else {
      String issueDate = dateMathString("now", now);
      String expiryDate = dateMathString("now+10d/d", now);
```

```
        return new LicenseSpec(version, uid, feature, issueDate, expiryDate, type, subscriptionType, issuedTo,
issuer, maxNodes);
    }
  }

  public static String generateLicenseSpecString(LicenseSpec licenseSpec) throws IOException {
    XContentBuilder licenses = jsonBuilder();
    licenses.startObject();
    licenses.startArray("licenses");
    licenses.startObject()
        .field("uid", licenseSpec.uid)
        .field("type", licenseSpec.type)
        .field("subscription_type", licenseSpec.subscriptionType)
        .field("issued_to", licenseSpec.issuedTo)
        .field("issuer", licenseSpec.issuer)
        .field("feature", licenseSpec.feature)
        .field("max_nodes", licenseSpec.maxNodes);

    if (licenseSpec.issueDate != null) {
        licenses.field("issue_date", licenseSpec.issueDate);
    } else {
        licenses.field("issue_date_in_millis", licenseSpec.issueDateInMillis);
    }
    if (licenseSpec.expiryDate != null) {
        licenses.field("expiry_date", licenseSpec.expiryDate);
    } else {
        licenses.field("expiry_date_in_millis", licenseSpec.expiryDateInMillis);
    }
    licenses.field("version", licenseSpec.version);
    licenses.endObject();
    licenses.endArray();
    licenses.endObject();
    return Strings.toString(licenses);
  }

  public static License generateLicenses(LicenseSpec spec) {
    License.Builder builder = License.builder()
        .uid(spec.uid)
        .feature(spec.feature)
        .type(spec.type)
        .subscriptionType(spec.subscriptionType)
        .issuedTo(spec.issuedTo)
        .issuer(spec.issuer)
        .maxNodes(spec.maxNodes);

    if (spec.expiryDate != null) {
        builder.expiryDate(DateUtils.endOfTheDay(spec.expiryDate));
    } else {
```

```
            builder.expiryDate(spec.expiryDateInMillis);
        }
        if (spec.issueDate != null) {
            builder.issueDate(DateUtils.beginningOfTheDay(spec.issueDate));
        } else {
            builder.issueDate(spec.issueDateInMillis);
        }
        return builder.build();
    }

    public static void assertLicenseSpec(LicenseSpec spec, License license) {
        MatcherAssert.assertThat(license.uid(), equalTo(spec.uid));
        MatcherAssert.assertThat(license.issuedTo(), equalTo(spec.issuedTo));
        MatcherAssert.assertThat(license.issuer(), equalTo(spec.issuer));
        MatcherAssert.assertThat(license.type(), equalTo(spec.type));
        MatcherAssert.assertThat(license.maxNodes(), equalTo(spec.maxNodes));
        if (spec.issueDate != null) {
            MatcherAssert.assertThat(license.issueDate(), equalTo(DateUtils.beginningOfTheDay(spec.issueDate)));
        } else {
            MatcherAssert.assertThat(license.issueDate(), equalTo(spec.issueDateInMillis));
        }
        if (spec.expiryDate != null) {
            MatcherAssert.assertThat(license.expiryDate(), equalTo(DateUtils.endOfTheDay(spec.expiryDate)));
        } else {
            MatcherAssert.assertThat(license.expiryDate(), equalTo(spec.expiryDateInMillis));
        }
    }

    public static class LicenseSpec {
        public final int version;
        public final String feature;
        public final String issueDate;
        public final long issueDateInMillis;
        public final String expiryDate;
        public final long expiryDateInMillis;
        public final String uid;
        public final String type;
        public final String subscriptionType;
        public final String issuedTo;
        public final String issuer;
        public final int maxNodes;

        public LicenseSpec(String issueDate, String expiryDate) {
            this(License.VERSION_CURRENT, UUID.randomUUID().toString(), "feature", issueDate, expiryDate,
"trial", "none", "customer",
                "elasticsearch", 5);
        }
```

```
    public LicenseSpec(int version, String uid, String feature, long issueDateInMillis, long expiryDateInMillis,
String type,
                    String subscriptionType, String issuedTo, String issuer, int maxNodes) {
        this.version = version;
        this.feature = feature;
        this.issueDateInMillis = issueDateInMillis;
        this.issueDate = null;
        this.expiryDateInMillis = expiryDateInMillis;
        this.expiryDate = null;
        this.uid = uid;
        this.type = type;
        this.subscriptionType = subscriptionType;
        this.issuedTo = issuedTo;
        this.issuer = issuer;
        this.maxNodes = maxNodes;
    }

    public LicenseSpec(int version, String uid, String feature, String issueDate, String expiryDate, String type,
                    String subscriptionType, String issuedTo, String issuer, int maxNodes) {
        this.version = version;
        this.feature = feature;
        this.issueDate = issueDate;
        this.issueDateInMillis = -1;
        this.expiryDate = expiryDate;
        this.expiryDateInMillis = -1;
        this.uid = uid;
        this.type = type;
        this.subscriptionType = subscriptionType;
        this.issuedTo = issuedTo;
        this.issuer = issuer;
        this.maxNodes = maxNodes;
    }
}
private static Path getTestPriKeyPath() throws Exception {
    return getResourcePath("/private.key");
}

private static Path getTestPubKeyPath() throws Exception {
    return getResourcePath("/public.key");
}

public static String dumpLicense(License license) throws Exception {
    XContentBuilder builder = XContentFactory.contentBuilder(XContentType.JSON);
    builder.startObject();
    builder.startObject("license");
    license.toInnerXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.endObject();
    builder.endObject();
```

```
        return Strings.toString(builder);
    }

    public static License generateSignedLicense(TimeValue expiryDuration) throws Exception {
        return generateSignedLicense(null, -1, expiryDuration);
    }

    public static License generateSignedLicense(String type, TimeValue expiryDuration) throws Exception {
        return generateSignedLicense(type, -1, expiryDuration);
    }

    public static License generateSignedLicense(long issueDate, TimeValue expiryDuration) throws Exception {
        return generateSignedLicense(null, issueDate, expiryDuration);
    }

    public static License generateSignedLicense(String type, long issueDate, TimeValue expiryDuration) throws
Exception {
        return generateSignedLicense(type, randomIntBetween(License.VERSION_START,
License.VERSION_CURRENT), issueDate, expiryDuration);
    }

    public static License generateSignedLicenseOldSignature() {
        long issueDate = System.currentTimeMillis();
        License.Builder specBuilder = License.builder()
                .uid(UUID.randomUUID().toString())
                .version(License.VERSION_START_DATE)
                .issuedTo("customer")
                .maxNodes(5)
                .type("trial")
                .issueDate(issueDate)
                .expiryDate(issueDate + TimeValue.timeValueHours(24).getMillis());
        return SelfGeneratedLicense.create(specBuilder, License.VERSION_START_DATE);
    }

    /**
     * This method which chooses the license type randomly if the type is null. However, it will not randomly
     * choose trial or basic types as those types can only be self-generated.
     */
    public static License generateSignedLicense(String type, int version, long issueDate, TimeValue expiryDuration)
throws Exception {
        long issue = (issueDate != -1L) ? issueDate : System.currentTimeMillis() -
TimeValue.timeValueHours(2).getMillis();
        final String licenseType;
        if (version < License.VERSION_NO_FEATURE_TYPE) {
            licenseType = randomFrom("subscription", "internal", "development");
        } else {
            licenseType = (type != null) ? type : randomFrom("silver", "dev", "gold", "platinum");
        }
```

```
    final License.Builder builder = License.builder()
            .uid(UUID.randomUUID().toString())
            .version(version)
            .expiryDate(System.currentTimeMillis() + expiryDuration.getMillis())
            .issueDate(issue)
            .type(licenseType)
            .issuedTo("customer")
            .issuer("elasticsearch")
            .maxNodes(5);
    if (version == License.VERSION_START) {
        builder.subscriptionType((type != null) ? type : randomFrom("dev", "gold", "platinum", "silver"));
        builder.feature(randomAlphaOfLength(10));
    }
    final LicenseSigner signer = new LicenseSigner(getTestPriKeyPath(), getTestPubKeyPath());
    return signer.sign(builder.build());
}

public static License generateSignedLicense(License.Builder builder) throws Exception {
    LicenseSigner signer = new LicenseSigner(getTestPriKeyPath(), getTestPubKeyPath());
    return signer.sign(builder.build());
}

public static License generateExpiredNonBasicLicense(long expiryDate) throws Exception {
    return generateExpiredNonBasicLicense(randomFrom("silver", "dev", "gold", "platinum"), expiryDate);
}

public static License generateExpiredNonBasicLicense() throws Exception {
    return generateExpiredNonBasicLicense(randomFrom("silver", "dev", "gold", "platinum"));
}

public static License generateExpiredNonBasicLicense(String type) throws Exception {
    return generateExpiredNonBasicLicense(type,
            System.currentTimeMillis() - TimeValue.timeValueHours(randomIntBetween(1, 10)).getMillis());
}

public static License generateExpiredNonBasicLicense(String type, long expiryDate) throws Exception {
    final License.Builder builder = License.builder()
            .uid(UUID.randomUUID().toString())
            .version(License.VERSION_CURRENT)
            .expiryDate(expiryDate)
            .issueDate(expiryDate - TimeValue.timeValueMinutes(10).getMillis())
            .type(type)
            .issuedTo("customer")
            .issuer("elasticsearch")
            .maxNodes(5);
    LicenseSigner signer = new LicenseSigner(getTestPriKeyPath(), getTestPubKeyPath());
    return signer.sign(builder.build());
}
```

```
private static Path getResourcePath(String resource) throws Exception {
    Path resourceFile = createTempFile();
    try (InputStream resourceInput = TestUtils.class.getResourceAsStream(resource)) {
        Files.copy(resourceInput, resourceFile, StandardCopyOption.REPLACE_EXISTING);
    }
    return resourceFile;
}

public static void registerAndAckSignedLicenses(final LicenseService licenseService, License license,
                                                final LicensesStatus expectedStatus) {
    PutLicenseRequest putLicenseRequest = new PutLicenseRequest().license(license).acknowledge(true);
    final CountDownLatch latch = new CountDownLatch(1);
    final AtomicReference<LicensesStatus> status = new AtomicReference<>();
    licenseService.registerLicense(putLicenseRequest, new ActionListener<PutLicenseResponse>() {
        @Override
        public void onResponse(PutLicenseResponse licensesUpdateResponse) {
            status.set(licensesUpdateResponse.status());
            latch.countDown();
        }

        @Override
        public void onFailure(Exception e) {
            latch.countDown();
        }
    });
    try {
        latch.await();
    } catch (InterruptedException e) {
        Assert.fail(e.getMessage());
    }
    assertThat(status.get(), equalTo(expectedStatus));
}

public static class AssertingLicenseState extends XPackLicenseState {
    public final List<License.OperationMode> modeUpdates = new ArrayList<>();
    public final List<Boolean> activeUpdates = new ArrayList<>();
    public final List<Version> trialVersionUpdates = new ArrayList<>();

    public AssertingLicenseState() {
        super(Settings.EMPTY);
    }

    @Override
    void update(License.OperationMode mode, boolean active, Version mostRecentTrialVersion) {
        modeUpdates.add(mode);
        activeUpdates.add(active);
        trialVersionUpdates.add(mostRecentTrialVersion);
```

```
        }
    }

    /**
     * A license state that makes the {@link #update(License.OperationMode, boolean, Version)}
     * method public for use in tests.
     */
    public static class UpdatableLicenseState extends XPackLicenseState {
        public UpdatableLicenseState() {
            this(Settings.EMPTY);
        }

        public UpdatableLicenseState(Settings settings) {
            super(settings);
        }

        @Override
        public void update(License.OperationMode mode, boolean active, Version mostRecentTrialVersion) {
            super.update(mode, active, mostRecentTrialVersion);
        }
    }

    public static void putLicense(MetaData.Builder builder, License license) {
        builder.putCustom(LicensesMetaData.TYPE, new LicensesMetaData(license, null));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.cluster.ack.ClusterStateUpdateResponse;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.protocol.xpack.license.LicensesStatus;
import org.elasticsearch.test.ESSingleNodeTestCase;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackSettings;
import org.junit.Before;

import java.util.Collection;
import java.util.Collections;
```

```
import java.util.concurrent.CountDownLatch;
import java.util.concurrent.atomic.AtomicBoolean;

import static org.hamcrest.Matchers.equalTo;
import static org.hamcrest.Matchers.not;

public class LicensesManagerServiceTests extends ESSingleNodeTestCase {

  @Override
  protected Collection<Class<? extends Plugin>> getPlugins() {
    return Collections.singletonList(LocalStateCompositeXPackPlugin.class);
  }

  @Override
  protected Settings nodeSettings() {
    return Settings.builder()
      .put(super.nodeSettings())
      .put(XPackSettings.SECURITY_ENABLED.getKey(), false)
      .put(XPackSettings.MONITORING_ENABLED.getKey(), false)
      .put(XPackSettings.WATCHER_ENABLED.getKey(), false)
      .put(XPackSettings.GRAPH_ENABLED.getKey(), false)
      .put(XPackSettings.MACHINE_LEARNING_ENABLED.getKey(), false)
      .build();
  }

  @Override
  protected boolean resetNodeAfterTest() {
    return true;
  }

  @Before
  public void waitForTrialLicenseToBeGenerated() throws Exception {
    assertBusy(() ->
assertNotNull(getInstanceFromNode(ClusterService.class).state().metaData().custom(LicensesMetaData.TYPE)));
  }

  public void testStoreAndGetLicenses() throws Exception {
    LicenseService licenseService = getInstanceFromNode(LicenseService.class);
    ClusterService clusterService = getInstanceFromNode(ClusterService.class);
    License goldLicense = TestUtils.generateSignedLicense("gold", TimeValue.timeValueHours(1));
    TestUtils.registerAndAckSignedLicenses(licenseService, goldLicense, LicensesStatus.VALID);
    License silverLicense = TestUtils.generateSignedLicense("silver", TimeValue.timeValueHours(2));
    TestUtils.registerAndAckSignedLicenses(licenseService, silverLicense, LicensesStatus.VALID);
    License platinumLicense = TestUtils.generateSignedLicense("platinum", TimeValue.timeValueHours(1));
    TestUtils.registerAndAckSignedLicenses(licenseService, platinumLicense, LicensesStatus.VALID);
    LicensesMetaData licensesMetaData = clusterService.state().metaData().custom(LicensesMetaData.TYPE);
    assertThat(licensesMetaData.getLicense(), equalTo(platinumLicense));
    final License getLicenses = licenseService.getLicense();
```

```
    assertThat(getLicenses, equalTo(platinumLicense));
}

// TODO: Add test/feature blocking the registration of basic license

public void testEffectiveLicenses() throws Exception {
    final LicenseService licenseService = getInstanceFromNode(LicenseService.class);
    final ClusterService clusterService = getInstanceFromNode(ClusterService.class);
    License goldLicense = TestUtils.generateSignedLicense("gold", TimeValue.timeValueSeconds(5));
    // put gold license
    TestUtils.registerAndAckSignedLicenses(licenseService, goldLicense, LicensesStatus.VALID);
    LicensesMetaData licensesMetaData = clusterService.state().metaData().custom(LicensesMetaData.TYPE);
    assertThat(LicenseService.getLicense(licensesMetaData), equalTo(goldLicense));

    License platinumLicense = TestUtils.generateSignedLicense("platinum", TimeValue.timeValueSeconds(3));
    // put platinum license
    TestUtils.registerAndAckSignedLicenses(licenseService, platinumLicense, LicensesStatus.VALID);
    licensesMetaData = clusterService.state().metaData().custom(LicensesMetaData.TYPE);
    assertThat(LicenseService.getLicense(licensesMetaData), equalTo(platinumLicense));
}

public void testInvalidLicenseStorage() throws Exception {
    LicenseService licenseService = getInstanceFromNode(LicenseService.class);
    ClusterService clusterService = getInstanceFromNode(ClusterService.class);
    License signedLicense = TestUtils.generateSignedLicense(TimeValue.timeValueMinutes(2));

    // modify content of signed license
    License tamperedLicense = License.builder()
            .fromLicenseSpec(signedLicense, signedLicense.signature())
            .expiryDate(signedLicense.expiryDate() + 10 * 24 * 60 * 60 * 1000L)
            .validate()
            .build();

    TestUtils.registerAndAckSignedLicenses(licenseService, tamperedLicense, LicensesStatus.INVALID);

    // ensure that the invalid license never made it to cluster state
    LicensesMetaData licensesMetaData = clusterService.state().metaData().custom(LicensesMetaData.TYPE);
    assertThat(licensesMetaData.getLicense(), not(equalTo(tamperedLicense)));
}

public void testRemoveLicenses() throws Exception {
    LicenseService licenseService = getInstanceFromNode(LicenseService.class);
    ClusterService clusterService = getInstanceFromNode(ClusterService.class);

    // generate signed licenses
    License license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(1));
    TestUtils.registerAndAckSignedLicenses(licenseService, license, LicensesStatus.VALID);
    LicensesMetaData licensesMetaData = clusterService.state().metaData().custom(LicensesMetaData.TYPE);
```

```
        assertThat(licensesMetaData.getLicense(), not(LicensesMetaData.LICENSE_TOMBSTONE));

        // remove signed licenses
        removeAndAckSignedLicenses(licenseService);
        licensesMetaData = clusterService.state().metaData().custom(LicensesMetaData.TYPE);
        assertThat(licensesMetaData.getLicense(), equalTo(LicensesMetaData.LICENSE_TOMBSTONE));
    }

    private void removeAndAckSignedLicenses(final LicenseService licenseService) {
        final CountDownLatch latch = new CountDownLatch(1);
        final AtomicBoolean success = new AtomicBoolean(false);
        licenseService.removeLicense(new DeleteLicenseRequest(), new
ActionListener<ClusterStateUpdateResponse>() {
            @Override
            public void onResponse(ClusterStateUpdateResponse clusterStateUpdateResponse) {
                if (clusterStateUpdateResponse.isAcknowledged()) {
                    success.set(true);
                }
                latch.countDown();
            }

            @Override
            public void onFailure(Exception throwable) {
                latch.countDown();
            }
        });
        try {
            latch.await();
        } catch (InterruptedException e) {
            fail(e.getMessage());
        }
        assertThat("remove license(s) failed", success.get(), equalTo(true));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.analysis.common.CommonAnalysisPlugin;
import org.elasticsearch.common.network.NetworkModule;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.env.Environment;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.test.ESIntegTestCase.ClusterScope;
```

```
import org.elasticsearch.transport.Netty4Plugin;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackPlugin;

import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Path;
import java.util.Arrays;
import java.util.Collection;

import static org.elasticsearch.test.ESIntegTestCase.Scope.TEST;
import static org.hamcrest.CoreMatchers.equalTo;
import static org.hamcrest.CoreMatchers.nullValue;

@ClusterScope(scope = TEST, numDataNodes = 0, numClientNodes = 0, maxNumDataNodes = 0,
transportClientRatio = 0)
public class LicenseServiceClusterTests extends AbstractLicensesIntegrationTestCase {

  @Override
  protected Settings nodeSettings(int nodeOrdinal) {
    return nodeSettingsBuilder(nodeOrdinal).build();
  }

  private Settings.Builder nodeSettingsBuilder(int nodeOrdinal) {
    return Settings.builder()
        .put(super.nodeSettings(nodeOrdinal))
        .put("node.data", true)
        .put("resource.reload.interval.high", "500ms") // for license mode file watcher
        .put(NetworkModule.HTTP_ENABLED.getKey(), true);
  }

  @Override
  protected Collection<Class<? extends Plugin>> nodePlugins() {
    return Arrays.asList(LocalStateCompositeXPackPlugin.class, CommonAnalysisPlugin.class,
Netty4Plugin.class);
  }

  @Override
  protected Collection<Class<? extends Plugin>> transportClientPlugins() {
    return nodePlugins();
  }

  public void testClusterRestartWithLicense() throws Exception {
    wipeAllLicenses();

    int numNodes = randomIntBetween(1, 5);
    logger.info("--> starting {} node(s)", numNodes);
    internalCluster().startNodes(numNodes);
```

```
        ensureGreen();

        logger.info("--> put signed license");
        LicensingClient licensingClient = new LicensingClient(client());
        License license = TestUtils.generateSignedLicense(TimeValue.timeValueMinutes(1));
        putLicense(license);
        assertThat(licensingClient.prepareGetLicense().get().license(), equalTo(license));
        assertOperationMode(license.operationMode());

        logger.info("--> restart all nodes");
        internalCluster().fullRestart();
        ensureYellow();
        licensingClient = new LicensingClient(client());
        logger.info("--> get and check signed license");
        assertThat(licensingClient.prepareGetLicense().get().license(), equalTo(license));
        logger.info("--> remove licenses");
        licensingClient.prepareDeleteLicense().get();
        assertOperationMode(License.OperationMode.MISSING);

        logger.info("--> restart all nodes");
        internalCluster().fullRestart();
        licensingClient = new LicensingClient(client());
        ensureYellow();
        assertThat(licensingClient.prepareGetLicense().get().license(), nullValue());
        assertOperationMode(License.OperationMode.MISSING);


        wipeAllLicenses();
    }

    public void testCloudInternalLicense() throws Exception {
        wipeAllLicenses();

        int numNodes = randomIntBetween(1, 5);
        logger.info("--> starting {} node(s)", numNodes);
        internalCluster().startNodes(numNodes);
        ensureGreen();

        logger.info("--> put signed license");
        LicensingClient licensingClient = new LicensingClient(client());
        License license = TestUtils.generateSignedLicense("cloud_internal", License.VERSION_CURRENT,
System.currentTimeMillis(),
                TimeValue.timeValueMinutes(1));
        putLicense(license);
        assertThat(licensingClient.prepareGetLicense().get().license(), equalTo(license));
        assertOperationMode(License.OperationMode.PLATINUM);
        writeCloudInternalMode("gold");
        assertOperationMode(License.OperationMode.GOLD);
```

```
        writeCloudInternalMode("basic");
        assertOperationMode(License.OperationMode.BASIC);
    }

    public void testClusterRestartWhileEnabled() throws Exception {
        wipeAllLicenses();
        internalCluster().startNode();
        ensureGreen();
        assertLicenseActive(true);
        logger.info("--> restart node");
        internalCluster().fullRestart();
        ensureYellow();
        logger.info("--> await node for enabled");
        assertLicenseActive(true);
    }

    public void testClusterRestartWhileGrace() throws Exception {
        wipeAllLicenses();
        internalCluster().startNode();
        assertLicenseActive(true);
        putLicense(TestUtils.generateSignedLicense(TimeValue.timeValueMillis(0)));
        ensureGreen();
        assertLicenseActive(true);
        logger.info("--> restart node");
        internalCluster().fullRestart();
        ensureYellow();
        logger.info("--> await node for grace_period");
        assertLicenseActive(true);
    }

    public void testClusterRestartWhileExpired() throws Exception {
        wipeAllLicenses();
        internalCluster().startNode();
        ensureGreen();
        assertLicenseActive(true);
        putLicense(TestUtils.generateExpiredNonBasicLicense(System.currentTimeMillis() -
LicenseService.GRACE_PERIOD_DURATION.getMillis()));
        assertLicenseActive(false);
        logger.info("--> restart node");
        internalCluster().fullRestart();
        ensureYellow();
        logger.info("--> await node for disabled");
        assertLicenseActive(false);
    }

    public void testClusterRestartWithOldSignature() throws Exception {
        assumeFalse("Can't run in a FIPS JVM. We can't generate old licenses since PBEWithSHA1AndDESede is not
available", inFipsJvm());
```

```
        wipeAllLicenses();
        internalCluster().startNode();
        ensureGreen();
        assertLicenseActive(true);
        putLicense(TestUtils.generateSignedLicenseOldSignature());
        LicensingClient licensingClient = new LicensingClient(client());
        assertThat(licensingClient.prepareGetLicense().get().license().version(),
equalTo(License.VERSION_START_DATE));
        logger.info("--> restart node");
        internalCluster().fullRestart(); // restart so that license is updated
        ensureYellow();
        logger.info("--> await node for enabled");
        assertLicenseActive(true);
        licensingClient = new LicensingClient(client());
        assertThat(licensingClient.prepareGetLicense().get().license().version(),
equalTo(License.VERSION_CURRENT)); //license updated
        internalCluster().fullRestart(); // restart once more and verify updated license is active
        ensureYellow();
        logger.info("--> await node for enabled");
        assertLicenseActive(true);
    }

    private void assertOperationMode(License.OperationMode operationMode) throws InterruptedException {
        boolean success = awaitBusy(() -> {
            for (XPackLicenseState licenseState : internalCluster().getDataNodeInstances(XPackLicenseState.class)) {
                if (licenseState.getOperationMode() == operationMode) {
                    return true;
                }
            }
            return false;
        });
        assertTrue(success);
    }

    private void writeCloudInternalMode(String mode) throws Exception {
        for (Environment environment : internalCluster().getDataOrMasterNodeInstances(Environment.class)) {
            Path licenseModePath = XPackPlugin.resolveConfigFile(environment, "license_mode");
            Files.createDirectories(licenseModePath.getParent());
            Files.write(licenseModePath, mode.getBytes(StandardCharsets.UTF_8));
        }
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
package org.elasticsearch.license;

import org.elasticsearch.test.ESTestCase;

import java.util.Locale;

import static org.elasticsearch.license.License.OperationMode;
import static org.hamcrest.Matchers.equalTo;

/**
* Tests {@link License.OperationMode} for correctness.
* <p>
* If you change the behavior of these tests, then it means that licensing changes across the products!
*/
public class LicenseOperationModeTests extends ESTestCase {
  public void testResolveTrial() {
    // assert 1.x BWC
    assertResolve(OperationMode.TRIAL, "nONE", "DEv", "deveLopment");
    // assert expected (2.x+) variant
    assertResolve(OperationMode.TRIAL, "tRiAl", "trial");
  }

  public void testResolveBasic() {
    // assert expected (2.x+) variant (note: no 1.x variant of BASIC)
    assertResolve(OperationMode.BASIC, "bAsIc", "basic");
  }

  public void testResolveStandard() {
    // assert expected (2.x+) variant (note: no 1.x variant of STANDARD)
    assertResolve(OperationMode.STANDARD, "StAnDARd", "standard");
  }

  public void testResolveGold() {
    // assert expected (2.x+) variant (note: no different 1.x variant of GOLD)
    assertResolve(OperationMode.GOLD, "SiLvEr", "gOlD", "silver", "gold");
  }

  public void testResolvePlatinum() {
    // assert 1.x BWC
    assertResolve(OperationMode.PLATINUM, "iNtErNaL");
    // assert expected (2.x+) variant
    assertResolve(OperationMode.PLATINUM, "PlAtINum", "platinum");
  }

  public void testResolveUnknown() {
    // 'enterprise' is a type that exists in cloud but should be rejected under normal operation
    // See https://github.com/elastic/x-plugins/issues/3371
    String[] types = { "unknown", "fake", "enterprise" };
```

```
        for (String type : types) {
            try {
                OperationMode.resolve(type);

                fail(String.format(Locale.ROOT, "[%s] should not be recognized as an operation mode", type));
            }
            catch (IllegalArgumentException e) {
                assertThat(e.getMessage(), equalTo("unknown type [" + type + "]"));
            }
        }
    }

    private static void assertResolve(OperationMode expected, String... types) {
        for (String type : types) {
            assertThat(OperationMode.resolve(type), equalTo(expected));
        }
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

package org.elasticsearch.license;

import org.elasticsearch.action.support.PlainActionFuture;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;

import static org.hamcrest.Matchers.containsString;
import static org.mockito.Matchers.any;
import static org.mockito.Mockito.verify;

public class LicenseFIPSTests extends AbstractLicenseServiceTestCase {

    public void testFIPSCheckWithAllowedLicense() throws Exception {
        License newLicense = TestUtils.generateSignedLicense(randomFrom("trial", "platinum"),
TimeValue.timeValueHours(24L));
        PutLicenseRequest request = new PutLicenseRequest();
        request.acknowledge(true);
        request.license(newLicense);
        Settings settings = Settings.builder()
            .put("xpack.security.enabled", true)
            .put("xpack.security.transport.ssl.enabled", true)
```

```
            .put("xpack.security.fips_mode.enabled", randomBoolean())
            .build();
        XPackLicenseState licenseState = new XPackLicenseState(settings);

        setInitialState(null, licenseState, settings);
        licenseService.start();
        PlainActionFuture<PutLicenseResponse> responseFuture = new PlainActionFuture<>();
        licenseService.registerLicense(request, responseFuture);
        verify(clusterService).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
    }

    public void testFIPSCheckWithoutAllowedLicense() throws Exception {
        License newLicense = TestUtils.generateSignedLicense(randomFrom("gold", "standard"),
TimeValue.timeValueHours(24L));
        PutLicenseRequest request = new PutLicenseRequest();
        request.acknowledge(true);
        request.license(newLicense);
        Settings settings = Settings.builder()
            .put("xpack.security.enabled", true)
            .put("xpack.security.transport.ssl.enabled", true)
            .put("xpack.security.fips_mode.enabled", true)
            .build();
        XPackLicenseState licenseState = new XPackLicenseState(settings);

        setInitialState(null, licenseState, settings);
        licenseService.start();
        PlainActionFuture<PutLicenseResponse> responseFuture = new PlainActionFuture<>();
        IllegalStateException e = expectThrows(IllegalStateException.class, () ->
licenseService.registerLicense(request, responseFuture));
        assertThat(e.getMessage(),
            containsString("Cannot install a [" + newLicense.operationMode() + "] license unless FIPS mode is
disabled"));
        licenseService.stop();

        settings = Settings.builder()
            .put("xpack.security.enabled", true)
            .put("xpack.security.transport.ssl.enabled", true)
            .put("xpack.security.fips_mode.enabled", false)
            .build();
        licenseState = new XPackLicenseState(settings);

        setInitialState(null, licenseState, settings);
        licenseService.start();
        licenseService.registerLicense(request, responseFuture);
        verify(clusterService).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
    }
}
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.cluster.ClusterModule;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.metadata.RepositoriesMetaData;
import org.elasticsearch.cluster.metadata.RepositoryMetaData;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.common.xcontent.NamedXContentRegistry;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.ToXContent.Params;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.test.ESTestCase;

import java.util.Collections;
import java.util.UUID;
import java.util.stream.Collectors;
import java.util.stream.Stream;

import static org.hamcrest.Matchers.equalTo;
import static org.hamcrest.Matchers.notNullValue;
import static org.hamcrest.Matchers.nullValue;

public class LicensesMetaDataSerializationTests extends ESTestCase {

  public void testXContentSerializationOneSignedLicense() throws Exception {
      License license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(2));
      LicensesMetaData licensesMetaData = new LicensesMetaData(license, null);
      XContentBuilder builder = XContentFactory.jsonBuilder();
      builder.startObject();
      builder.startObject("licenses");
      licensesMetaData.toXContent(builder, ToXContent.EMPTY_PARAMS);
      builder.endObject();
      builder.endObject();
      LicensesMetaData licensesMetaDataFromXContent =
getLicensesMetaDataFromXContent(createParser(builder));
      assertThat(licensesMetaDataFromXContent.getLicense(), equalTo(license));
      assertNull(licensesMetaDataFromXContent.getMostRecentTrialVersion());
  }
```

```
  public void testXContentSerializationOneSignedLicenseWithUsedTrial() throws Exception {
    License license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(2));
    LicensesMetaData licensesMetaData = new LicensesMetaData(license, Version.CURRENT);
    XContentBuilder builder = XContentFactory.jsonBuilder();
    builder.startObject();
    builder.startObject("licenses");
    licensesMetaData.toXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.endObject();
    builder.endObject();
    LicensesMetaData licensesMetaDataFromXContent =
getLicensesMetaDataFromXContent(createParser(builder));
    assertThat(licensesMetaDataFromXContent.getLicense(), equalTo(license));
    assertEquals(licensesMetaDataFromXContent.getMostRecentTrialVersion(), Version.CURRENT);
  }

  public void testLicenseMetadataParsingDoesNotSwallowOtherMetaData() throws Exception {
    new Licensing(Settings.EMPTY); // makes sure LicensePlugin is registered in Custom MetaData
    License license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(2));
    LicensesMetaData licensesMetaData = new LicensesMetaData(license, Version.CURRENT);
    RepositoryMetaData repositoryMetaData = new RepositoryMetaData("repo", "fs", Settings.EMPTY);
    RepositoriesMetaData repositoriesMetaData = new
RepositoriesMetaData(Collections.singletonList(repositoryMetaData));
    final MetaData.Builder metaDataBuilder = MetaData.builder();
    if (randomBoolean()) { // random order of insertion
      metaDataBuilder.putCustom(licensesMetaData.getWriteableName(), licensesMetaData);
      metaDataBuilder.putCustom(repositoriesMetaData.getWriteableName(), repositoriesMetaData);
    } else {
      metaDataBuilder.putCustom(repositoriesMetaData.getWriteableName(), repositoriesMetaData);
      metaDataBuilder.putCustom(licensesMetaData.getWriteableName(), licensesMetaData);
    }
    // serialize metadata
    XContentBuilder builder = XContentFactory.jsonBuilder();
    Params params = new
ToXContent.MapParams(Collections.singletonMap(MetaData.CONTEXT_MODE_PARAM,
MetaData.CONTEXT_MODE_GATEWAY));
    builder.startObject();
    builder = metaDataBuilder.build().toXContent(builder, params);
    builder.endObject();
    // deserialize metadata again
    MetaData metaData = MetaData.Builder.fromXContent(createParser(builder));
    // check that custom metadata still present
    assertThat(metaData.custom(licensesMetaData.getWriteableName()), notNullValue());
    assertThat(metaData.custom(repositoriesMetaData.getWriteableName()), notNullValue());
  }

  public void testXContentSerializationOneTrial() throws Exception {
    long issueDate = System.currentTimeMillis();
    License.Builder specBuilder = License.builder()
```

```
            .uid(UUID.randomUUID().toString())
            .issuedTo("customer")
            .maxNodes(5)
            .issueDate(issueDate)
            .type(randomBoolean() ? "trial" : "basic")
            .expiryDate(issueDate + TimeValue.timeValueHours(2).getMillis());
        final License trialLicense = SelfGeneratedLicense.create(specBuilder, License.VERSION_CURRENT);
        LicensesMetaData licensesMetaData = new LicensesMetaData(trialLicense, Version.CURRENT);
        XContentBuilder builder = XContentFactory.jsonBuilder();
        builder.startObject();
        builder.startObject("licenses");
        licensesMetaData.toXContent(builder, ToXContent.EMPTY_PARAMS);
        builder.endObject();
        builder.endObject();
        LicensesMetaData licensesMetaDataFromXContent =
getLicensesMetaDataFromXContent(createParser(builder));
        assertThat(licensesMetaDataFromXContent.getLicense(), equalTo(trialLicense));
        assertEquals(licensesMetaDataFromXContent.getMostRecentTrialVersion(), Version.CURRENT);
    }

    public void testLicenseTombstoneFromXContext() throws Exception {
        final XContentBuilder builder = XContentFactory.jsonBuilder();
        builder.startObject();
        builder.startObject("licenses");
        builder.nullField("license");
        builder.endObject();
        builder.endObject();
        LicensesMetaData metaDataFromXContent = getLicensesMetaDataFromXContent(createParser(builder));
        assertThat(metaDataFromXContent.getLicense(), equalTo(LicensesMetaData.LICENSE_TOMBSTONE));
    }

    public void testLicenseTombstoneWithUsedTrialFromXContext() throws Exception {
        final XContentBuilder builder = XContentFactory.jsonBuilder();
        builder.startObject();
        builder.startObject("licenses");
        builder.nullField("license");
        builder.field("trial_license", Version.CURRENT.toString());
        builder.endObject();
        builder.endObject();
        LicensesMetaData metaDataFromXContent = getLicensesMetaDataFromXContent(createParser(builder));
        assertThat(metaDataFromXContent.getLicense(), equalTo(LicensesMetaData.LICENSE_TOMBSTONE));
        assertEquals(metaDataFromXContent.getMostRecentTrialVersion(), Version.CURRENT);
    }

    private static LicensesMetaData getLicensesMetaDataFromXContent(XContentParser parser) throws Exception {
        parser.nextToken(); // consume null
        parser.nextToken(); // consume "licenses"
        LicensesMetaData licensesMetaDataFromXContent = LicensesMetaData.fromXContent(parser);
```

```
        parser.nextToken(); // consume endObject
        assertThat(parser.nextToken(), nullValue());
        return licensesMetaDataFromXContent;
    }

    @Override
    protected NamedXContentRegistry xContentRegistry() {
        return new NamedXContentRegistry(Stream.concat(
                new Licensing(Settings.EMPTY).getNamedXContent().stream(),
                ClusterModule.getNamedXWriteables().stream()
        ).collect(Collectors.toList()));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.protocol.xpack.license.LicensesStatus;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;

import static org.elasticsearch.common.unit.TimeValue.timeValueHours;
import static org.hamcrest.Matchers.equalTo;
import static org.hamcrest.Matchers.not;
import static org.mockito.Matchers.any;
import static org.mockito.Mockito.times;
import static org.mockito.Mockito.verify;

public class LicensesAcknowledgementTests extends AbstractLicenseServiceTestCase {

    public void testAcknowledgment() throws Exception {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        setInitialState(TestUtils.generateSignedLicense("gold", timeValueHours(2)), licenseState, Settings.EMPTY);
        licenseService.start();
        // try installing a signed license
        long issueDate = System.currentTimeMillis() - TimeValue.timeValueHours(24 * 2).getMillis();
        License signedLicense = TestUtils.generateSignedLicense("trial", License.VERSION_CURRENT, issueDate,
timeValueHours(10));
        PutLicenseRequest putLicenseRequest = new PutLicenseRequest().license(signedLicense);
        // ensure acknowledgement message was part of the response
        licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(false,
LicensesStatus.VALID, true));
```

```
        assertThat(licenseService.getLicense(), not(signedLicense));
        verify(clusterService, times(0)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));

        // try installing a signed license with acknowledgement
        putLicenseRequest = new PutLicenseRequest().license(signedLicense).acknowledge(true);
        // ensure license was installed and no acknowledgment message was returned
        licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(true,
LicensesStatus.VALID, false));
        verify(clusterService, times(1)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
    }

    public void testRejectUpgradeToProductionWithoutTLS() throws Exception {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        setInitialState(TestUtils.generateSignedLicense("trial", timeValueHours(2)), licenseState, Settings.EMPTY);
        licenseService.start();
        // try installing a signed license
        License signedLicense = TestUtils.generateSignedLicense("platinum", timeValueHours(10));
        PutLicenseRequest putLicenseRequest = new PutLicenseRequest().license(signedLicense);
        // ensure acknowledgement message was part of the response
        IllegalStateException ise = expectThrows(IllegalStateException.class, () ->
                licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(false,
LicensesStatus.VALID, true)));
        assertEquals("Cannot install a [PLATINUM] license unless TLS is configured or security is disabled",
ise.getMessage());
    }

    public void testUpgradeToProductionWithoutTLSAndSecurityDisabled() throws Exception {
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        setInitialState(TestUtils.generateSignedLicense("trial", timeValueHours(2)), licenseState, Settings.builder()
                .put("xpack.security.enabled", false).build());
        licenseService.start();
        // try installing a signed license
        License signedLicense = TestUtils.generateSignedLicense("platinum", timeValueHours(10));
        PutLicenseRequest putLicenseRequest = new PutLicenseRequest().license(signedLicense);
        licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(false,
LicensesStatus.VALID, true));
        assertThat(licenseService.getLicense(), not(signedLicense));
        verify(clusterService, times(1)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));

        // try installing a signed license with acknowledgement
        putLicenseRequest = new PutLicenseRequest().license(signedLicense).acknowledge(true);
        // ensure license was installed and no acknowledgment message was returned
        licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(true,
LicensesStatus.VALID, false));
        verify(clusterService, times(2)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
    }

    public void testUpgradeToProductionWithTLSAndSecurity() throws Exception {
```

```
        XPackLicenseState licenseState = new XPackLicenseState(Settings.EMPTY);
        setInitialState(TestUtils.generateSignedLicense("trial", timeValueHours(2)), licenseState, Settings.builder()
                .put("xpack.security.enabled", true)
                .put("xpack.security.transport.ssl.enabled", true).build());
        licenseService.start();
        // try installing a signed license
        License signedLicense = TestUtils.generateSignedLicense("platinum", timeValueHours(10));
        PutLicenseRequest putLicenseRequest = new PutLicenseRequest().license(signedLicense);
        licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(false,
LicensesStatus.VALID, true));
        assertThat(licenseService.getLicense(), not(signedLicense));
        verify(clusterService, times(1)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));

        // try installing a signed license with acknowledgement
        putLicenseRequest = new PutLicenseRequest().license(signedLicense).acknowledge(true);
        // ensure license was installed and no acknowledgment message was returned
        licenseService.registerLicense(putLicenseRequest, new AssertingLicensesUpdateResponse(true,
LicensesStatus.VALID, false));
        verify(clusterService, times(2)).submitStateUpdateTask(any(String.class), any(ClusterStateUpdateTask.class));
    }

    private static class AssertingLicensesUpdateResponse implements ActionListener<PutLicenseResponse> {
        private final boolean expectedAcknowledgement;
        private final LicensesStatus expectedStatus;
        private final boolean expectAckMessages;

        AssertingLicensesUpdateResponse(boolean expectedAcknowledgement, LicensesStatus expectedStatus,
                                        boolean expectAckMessages) {
            this.expectedAcknowledgement = expectedAcknowledgement;
            this.expectedStatus = expectedStatus;
            this.expectAckMessages = expectAckMessages;
        }

        @Override
        public void onResponse(PutLicenseResponse licensesUpdateResponse) {
            assertThat(licensesUpdateResponse.isAcknowledged(), equalTo(expectedAcknowledgement));
            assertThat(licensesUpdateResponse.status(), equalTo(expectedStatus));
            assertEquals(licensesUpdateResponse.acknowledgeMessages().isEmpty(), expectAckMessages == false);
        }

        @Override
        public void onFailure(Exception throwable) {
            throw new RuntimeException(throwable);
        }
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
```

```
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.test.ESTestCase;

import static org.elasticsearch.common.unit.TimeValue.timeValueMillis;
import static org.hamcrest.Matchers.equalTo;
import static org.hamcrest.Matchers.nullValue;

public class ExpirationCallbackTests extends ESTestCase {

  public void testPostExpirationDelay() throws Exception {
    TimeValue expiryDuration = TimeValue.timeValueSeconds(randomIntBetween(5, 10));
    TimeValue min = TimeValue.timeValueSeconds(1);
    TimeValue max = TimeValue.timeValueSeconds(4);
    TimeValue frequency = TimeValue.timeValueSeconds(1);
    NoopPostExpirationCallback post = new NoopPostExpirationCallback(min, max, frequency);
    long now = System.currentTimeMillis();
    long expiryDate = now + expiryDuration.getMillis();
    assertThat(post.delay(expiryDate, now),
        equalTo(TimeValue.timeValueMillis(expiryDuration.getMillis() + min.getMillis()))); // before license
expiry
    assertThat(post.delay(expiryDate, expiryDate), equalTo(min)); // on license expiry
    int latestValidTriggerDelay = (int) (expiryDuration.getMillis() + max.getMillis());
    int earliestValidTriggerDelay = (int) (expiryDuration.getMillis() + min.getMillis());
    assertExpirationCallbackDelay(post, expiryDuration.millis(), latestValidTriggerDelay,
earliestValidTriggerDelay);
  }

  public void testPreExpirationDelay() throws Exception {
    TimeValue expiryDuration = TimeValue.timeValueSeconds(randomIntBetween(5, 10));
    TimeValue min = TimeValue.timeValueSeconds(1);
    TimeValue max = TimeValue.timeValueSeconds(4);
    TimeValue frequency = TimeValue.timeValueSeconds(1);
    NoopPreExpirationCallback pre = new NoopPreExpirationCallback(min, max, frequency);
    long now = System.currentTimeMillis();
    long expiryDate = now + expiryDuration.getMillis();
    assertThat(pre.delay(expiryDate, expiryDate), nullValue()); // on license expiry
    int latestValidTriggerDelay = (int) (expiryDuration.getMillis() - min.getMillis());
    int earliestValidTriggerDelay = (int) (expiryDuration.getMillis() - max.getMillis());
    assertExpirationCallbackDelay(pre, expiryDuration.millis(), latestValidTriggerDelay,
earliestValidTriggerDelay);
  }

  public void testPostExpirationWithNullMax() throws Exception {
```

```
        int postExpirySeconds = randomIntBetween(5, 10);
        TimeValue postExpiryDuration = TimeValue.timeValueSeconds(postExpirySeconds);
        TimeValue min = TimeValue.timeValueSeconds(postExpirySeconds - randomIntBetween(1, 3));

        final ExpirationCallback.Post post = new NoopPostExpirationCallback(min, null, timeValueMillis(10));
        long now = System.currentTimeMillis();
        assertThat(post.delay(now - postExpiryDuration.millis(), now), equalTo(TimeValue.timeValueMillis(0)));
    }

    public void testPreExpirationWithNullMin() throws Exception {
        int expirySeconds = randomIntBetween(5, 10);
        TimeValue expiryDuration = TimeValue.timeValueSeconds(expirySeconds);
        TimeValue max = TimeValue.timeValueSeconds(expirySeconds + randomIntBetween(1, 10));

        final ExpirationCallback.Pre pre = new NoopPreExpirationCallback(null, max, timeValueMillis(10));
        long now = System.currentTimeMillis();
        assertThat(pre.delay(expiryDuration.millis() + now, now), equalTo(TimeValue.timeValueMillis(0)));
    }

    public void testPreExpirationScheduleTime() throws Exception {
        TimeValue expiryDuration = TimeValue.timeValueSeconds(randomIntBetween(5, 10));
        TimeValue min = TimeValue.timeValueSeconds(1);
        TimeValue max = TimeValue.timeValueSeconds(4);
        TimeValue frequency = TimeValue.timeValueSeconds(1);
        NoopPreExpirationCallback pre = new NoopPreExpirationCallback(min, max, frequency);
        int latestValidTriggerDelay = (int) (expiryDuration.getMillis() - min.getMillis());
        int earliestValidTriggerDelay = (int) (expiryDuration.getMillis() - max.getMillis());
        assertExpirationCallbackScheduleTime(pre, expiryDuration.millis(), latestValidTriggerDelay,
earliestValidTriggerDelay);
    }

    public void testPostExpirationScheduleTime() throws Exception {
        TimeValue expiryDuration = TimeValue.timeValueSeconds(randomIntBetween(5, 10));
        TimeValue min = TimeValue.timeValueSeconds(1);
        TimeValue max = TimeValue.timeValueSeconds(4);
        TimeValue frequency = TimeValue.timeValueSeconds(1);
        NoopPostExpirationCallback pre = new NoopPostExpirationCallback(min, max, frequency);
        int latestValidTriggerDelay = (int) (expiryDuration.getMillis() + max.getMillis());
        int earliestValidTriggerDelay = (int) (expiryDuration.getMillis() + min.getMillis());
        assertExpirationCallbackScheduleTime(pre, expiryDuration.millis(), latestValidTriggerDelay,
earliestValidTriggerDelay);
    }

    private void assertExpirationCallbackDelay(ExpirationCallback expirationCallback, long expiryDuration,
                                               int latestValidTriggerDelay, int earliestValidTriggerDelay) {
        long now = System.currentTimeMillis();
        long expiryDate = now + expiryDuration;
        // bounds
```

```
    assertThat(expirationCallback.delay(expiryDate, now + earliestValidTriggerDelay),
equalTo(TimeValue.timeValueMillis(0)));
    assertThat(expirationCallback.delay(expiryDate, now + latestValidTriggerDelay),
equalTo(TimeValue.timeValueMillis(0)));
    // in match
    assertThat(expirationCallback.delay(expiryDate,
        now + randomIntBetween(earliestValidTriggerDelay, latestValidTriggerDelay)),
        equalTo(TimeValue.timeValueMillis(0)));
    // out of bounds
    int deltaBeforeEarliestMatch = between(1, earliestValidTriggerDelay);
    assertThat(expirationCallback.delay(expiryDate, now + deltaBeforeEarliestMatch),
        equalTo(TimeValue.timeValueMillis(earliestValidTriggerDelay - deltaBeforeEarliestMatch)));
    int deltaAfterLatestMatch = between(latestValidTriggerDelay + 1, Integer.MAX_VALUE); // after expiry and
after max
    assertThat(expirationCallback.delay(expiryDate, expiryDate + deltaAfterLatestMatch), nullValue());
  }

  public void assertExpirationCallbackScheduleTime(ExpirationCallback expirationCallback, long expiryDuration,
                                    int latestValidTriggerDelay, int earliestValidTriggerDelay) {
    long now = System.currentTimeMillis();
    long expiryDate = now + expiryDuration;
    int validTriggerInterval = between(earliestValidTriggerDelay, latestValidTriggerDelay);
    assertThat(expirationCallback.nextScheduledTimeForExpiry(expiryDate,
        now + validTriggerInterval, now + validTriggerInterval),
        equalTo(now + validTriggerInterval));
    assertThat(expirationCallback.nextScheduledTimeForExpiry(expiryDate, now, now + validTriggerInterval),
        equalTo(now + validTriggerInterval + expirationCallback.getFrequency()));

    int deltaBeforeEarliestMatch = between(1, earliestValidTriggerDelay - 1);
    assertThat(expirationCallback.nextScheduledTimeForExpiry(expiryDate, now, now +
deltaBeforeEarliestMatch),
        equalTo(now + deltaBeforeEarliestMatch +
            expirationCallback.delay(expiryDate, now + deltaBeforeEarliestMatch).getMillis()));
    assertThat(expirationCallback.nextScheduledTimeForExpiry(expiryDate,
        now + deltaBeforeEarliestMatch, now + deltaBeforeEarliestMatch),
        equalTo(now + deltaBeforeEarliestMatch +
            expirationCallback.delay(expiryDate, now + deltaBeforeEarliestMatch).getMillis()));

    int deltaAfterLatestMatch = between(latestValidTriggerDelay + 1, Integer.MAX_VALUE); // after expiry and
after max
    assertThat(expirationCallback.nextScheduledTimeForExpiry(expiryDate, now, now + deltaAfterLatestMatch),
equalTo(-1L));
    assertThat(expirationCallback.nextScheduledTimeForExpiry(expiryDate,
        now + deltaAfterLatestMatch, now + deltaAfterLatestMatch),
        equalTo(-1L));
  }

  private static class NoopPostExpirationCallback extends ExpirationCallback.Post {
```

```
    NoopPostExpirationCallback(TimeValue min, TimeValue max, TimeValue frequency) {
        super(min, max, frequency);
    }

    @Override
    public void on(License license) {}
  }

  private static class NoopPreExpirationCallback extends ExpirationCallback.Pre {

    NoopPreExpirationCallback(TimeValue min, TimeValue max, TimeValue frequency) {
        super(min, max, frequency);
    }

    @Override
    public void on(License license) {}
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.threadpool.TestThreadPool;
import org.elasticsearch.watcher.ResourceWatcherService;
import org.junit.After;
import org.junit.Before;

import java.io.IOException;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.StandardOpenOption;
import java.util.concurrent.CountDownLatch;
import java.util.concurrent.TimeUnit;
import java.util.concurrent.atomic.AtomicReference;

import static org.hamcrest.Matchers.equalTo;

public class OperationModeFileWatcherTests extends ESTestCase {
  private ResourceWatcherService watcherService;
  private TestThreadPool threadPool;
  private Path licenseModePath;
```

```
    private OperationModeFileWatcher operationModeFileWatcher;
    private AtomicReference<CountDownLatch> onChangeCounter;

    @Before
    public void setup() throws Exception {
        threadPool = new TestThreadPool("license mode file watcher tests");
        Settings settings = Settings.builder()
                .put("resource.reload.interval.high", "10ms")
                .build();
        watcherService = new ResourceWatcherService(settings,
                threadPool);
        watcherService.start();
        licenseModePath = createTempFile();
        onChangeCounter = new AtomicReference<>(new CountDownLatch(1));
        operationModeFileWatcher = new OperationModeFileWatcher(watcherService, licenseModePath, logger,
                () -> onChangeCounter.get().countDown());
    }

    @After
    public void shutdown() throws InterruptedException {
        terminate(threadPool);
        watcherService.stop();
    }

    public void testInit() throws Exception {
        onChangeCounter.set(new CountDownLatch(2));
        writeMode("gold");
        assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
        operationModeFileWatcher.init();
        assertTrue(onChangeCounter.get().await(5, TimeUnit.SECONDS));
        assertThat(operationModeFileWatcher.getCurrentOperationMode(), equalTo(License.OperationMode.GOLD));
    }

    public void testUpdateModeFromFile() throws Exception {
        Files.delete(licenseModePath);
        operationModeFileWatcher.init();
        assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
        writeMode("gold");
        assertTrue(onChangeCounter.get().await(5, TimeUnit.SECONDS));
        assertThat(operationModeFileWatcher.getCurrentOperationMode(), equalTo(License.OperationMode.GOLD));
        onChangeCounter.set(new CountDownLatch(1));
        writeMode("basic");
        assertTrue(onChangeCounter.get().await(5, TimeUnit.SECONDS));
        assertThat(operationModeFileWatcher.getCurrentOperationMode(), equalTo(License.OperationMode.BASIC));
    }
```

```
  public void testDeleteModeFromFile() throws Exception {
      Files.delete(licenseModePath);
      operationModeFileWatcher.init();
      writeMode("gold");
      assertTrue(onChangeCounter.get().await(5, TimeUnit.SECONDS));
      assertThat(operationModeFileWatcher.getCurrentOperationMode(), equalTo(License.OperationMode.GOLD));
      onChangeCounter.set(new CountDownLatch(1));

      Files.delete(licenseModePath);
      assertTrue(onChangeCounter.get().await(5, TimeUnit.SECONDS));
      assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
  }

  public void testInvalidModeFromFile() throws Exception {
      writeMode("invalid");
      operationModeFileWatcher.init();
      assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
      operationModeFileWatcher.onFileChanged(licenseModePath);
      assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
  }

  public void testLicenseModeFileIsDirectory() throws Exception {
      licenseModePath = createTempDir();
      operationModeFileWatcher.init();
      assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
      operationModeFileWatcher.onFileChanged(licenseModePath);
      assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
  }

  public void testLicenseModeFileCreatedAfterInit() throws Exception {
      Files.delete(licenseModePath);
      operationModeFileWatcher.init();
      assertThat(operationModeFileWatcher.getCurrentOperationMode(),
equalTo(License.OperationMode.PLATINUM));
      onChangeCounter.set(new CountDownLatch(1));
      Path tempFile = createTempFile();
      writeMode("gold", tempFile);
      licenseModePath = tempFile;
      assertTrue(onChangeCounter.get().await(5, TimeUnit.SECONDS));
      assertThat(operationModeFileWatcher.getCurrentOperationMode(), equalTo(License.OperationMode.GOLD));
  }

  private void writeMode(String mode) throws IOException {
```

```
        writeMode(mode, licenseModePath);
    }

    static void writeMode(String mode, Path file) throws IOException {
        Files.write(file, mode.getBytes(StandardCharsets.UTF_8), StandardOpenOption.CREATE);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.test.ESTestCase;
import org.elasticsearch.watcher.FileWatcher;
import org.elasticsearch.watcher.ResourceWatcherService;
import org.junit.Before;

import java.nio.file.Path;

import static org.hamcrest.Matchers.equalTo;
import static org.mockito.Matchers.any;
import static org.mockito.Matchers.eq;
import static org.mockito.Mockito.mock;
import static org.mockito.Mockito.times;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.verifyZeroInteractions;

public class LicenseOperationModeUpdateTests extends ESTestCase {

    private OperationModeFileWatcher operationModeFileWatcher;
    private Path licenseModeFile;
    private ResourceWatcherService resourceWatcherService;

    @Before
    public void init() throws Exception {
        licenseModeFile = createTempFile();
        resourceWatcherService = mock(ResourceWatcherService.class);
        operationModeFileWatcher = new OperationModeFileWatcher(resourceWatcherService, licenseModeFile,
logger, () -> {});
    }

    public void testLicenseOperationModeUpdate() throws Exception {
        String type = randomFrom("trial", "basic", "standard", "gold", "platinum");
        License license = License.builder()
                .uid("id")
                .expiryDate(0)
```

```
            .issueDate(0)
            .issuedTo("elasticsearch")
            .issuer("issuer")
            .type(type)
            .maxNodes(1)
            .build();

        assertThat(license.operationMode(), equalTo(License.OperationMode.resolve(type)));
        OperationModeFileWatcherTests.writeMode("gold", licenseModeFile);
        license.setOperationModeFileWatcher(operationModeFileWatcher);
        verifyZeroInteractions(resourceWatcherService);
        assertThat(license.operationMode(), equalTo(License.OperationMode.resolve(type)));
    }

    public void testCloudInternalLicenseOperationModeUpdate() throws Exception {
        License license = License.builder()
            .uid("id")
            .expiryDate(0)
            .issueDate(0)
            .issuedTo("elasticsearch")
            .issuer("issuer")
            .type("cloud_internal")
            .maxNodes(1)
            .build();

        assertThat(license.operationMode(), equalTo(License.OperationMode.PLATINUM));
        OperationModeFileWatcherTests.writeMode("gold", licenseModeFile);
        license.setOperationModeFileWatcher(operationModeFileWatcher);
        verify(resourceWatcherService, times(1)).add(any(FileWatcher.class),
eq(ResourceWatcherService.Frequency.HIGH));
        assertThat(license.operationMode(), equalTo(License.OperationMode.GOLD));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.bytes.BytesArray;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentHelper;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.test.ESTestCase;
```

```
import java.nio.charset.StandardCharsets;
import java.util.Collections;
import java.util.Map;
import java.util.UUID;

import static org.hamcrest.core.IsEqual.equalTo;
import static org.hamcrest.core.IsNull.notNullValue;
import static org.hamcrest.core.IsNull.nullValue;

public class LicenseSerializationTests extends ESTestCase {
  public void testSimpleIssueExpiryDate() throws Exception {
    long now = System.currentTimeMillis();
    String issueDate = TestUtils.dateMathString("now", now);
    String expiryDate = TestUtils.dateMathString("now+10d/d", now);
    String licenseSpecs = TestUtils.generateLicenseSpecString(new TestUtils.LicenseSpec(issueDate, expiryDate));
    License generatedLicense = License.fromSource(new
BytesArray(licenseSpecs.getBytes(StandardCharsets.UTF_8)), XContentType.JSON);
    assertThat(generatedLicense.issueDate(), equalTo(DateUtils.beginningOfTheDay(issueDate)));
    assertThat(generatedLicense.expiryDate(), equalTo(DateUtils.endOfTheDay(expiryDate)));
  }

  public void testLicensesFields() throws Exception {
    TestUtils.LicenseSpec randomLicenseSpec =
TestUtils.generateRandomLicenseSpec(License.VERSION_START);
    String licenseSpecsSource = TestUtils.generateLicenseSpecString(randomLicenseSpec);
    final License fromSource =
        License.fromSource(new BytesArray(licenseSpecsSource.getBytes(StandardCharsets.UTF_8)),
XContentType.JSON);
    TestUtils.assertLicenseSpec(randomLicenseSpec, fromSource);
  }

  public void testLicenseRestView() throws Exception {
    long now = System.currentTimeMillis();
    String expiredLicenseExpiryDate = TestUtils.dateMathString("now-1d/d", now);
    String validLicenseIssueDate = TestUtils.dateMathString("now-10d/d", now);
    String invalidLicenseIssueDate = TestUtils.dateMathString("now+1d/d", now);
    String validLicenseExpiryDate = TestUtils.dateMathString("now+2d/d", now);

    License license = TestUtils.generateLicenses(new TestUtils.LicenseSpec(validLicenseIssueDate,
expiredLicenseExpiryDate));
    XContentBuilder builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, new
ToXContent.MapParams(Collections.singletonMap(License.REST_VIEW_MODE, "true")));
    builder.flush();
    Map<String, Object> map = XContentHelper.convertToMap(BytesReference.bytes(builder), false,
builder.contentType()).v2();
```

```
    // should have an extra status field, human readable issue_data and expiry_date
    assertThat(map.get("status"), notNullValue());
    assertThat(map.get("issue_date"), notNullValue());
    assertThat(map.get("expiry_date"), notNullValue());
    assertThat(map.get("status"), equalTo("expired"));
    builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.flush();
    map = XContentHelper.convertToMap(BytesReference.bytes(builder), false, builder.contentType()).v2();
    assertThat(map.get("status"), nullValue());

    license = TestUtils.generateLicenses(new TestUtils.LicenseSpec(validLicenseIssueDate,
validLicenseExpiryDate));
    builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, new
ToXContent.MapParams(Collections.singletonMap(License.REST_VIEW_MODE, "true")));
    builder.flush();
    map = XContentHelper.convertToMap(BytesReference.bytes(builder), false, builder.contentType()).v2();

    // should have an extra status field, human readable issue_data and expiry_date
    assertThat(map.get("status"), notNullValue());
    assertThat(map.get("issue_date"), notNullValue());
    assertThat(map.get("expiry_date"), notNullValue());
    assertThat(map.get("status"), equalTo("active"));
    builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.flush();
    map = XContentHelper.convertToMap(BytesReference.bytes(builder), false, builder.contentType()).v2();
    assertThat(map.get("status"), nullValue());

    license = TestUtils.generateLicenses(new TestUtils.LicenseSpec(invalidLicenseIssueDate,
validLicenseExpiryDate));
    builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, new
ToXContent.MapParams(Collections.singletonMap(License.REST_VIEW_MODE, "true")));
    builder.flush();
    map = XContentHelper.convertToMap(BytesReference.bytes(builder), false, builder.contentType()).v2();

    // should have an extra status field, human readable issue_data and expiry_date
    assertThat(map.get("status"), notNullValue());
    assertThat(map.get("issue_date"), notNullValue());
    assertThat(map.get("expiry_date"), notNullValue());
    assertThat(map.get("status"), equalTo("invalid"));
    builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.flush();
    map = XContentHelper.convertToMap(BytesReference.bytes(builder), false, builder.contentType()).v2();
    assertThat(map.get("status"), nullValue());
```

```
  }

  public void testLicenseRestViewNonExpiringBasic() throws Exception {
    long now = System.currentTimeMillis();

    License.Builder specBuilder = License.builder()
        .uid(UUID.randomUUID().toString())
        .issuedTo("test")
        .maxNodes(1000)
        .issueDate(now)
        .type("basic")
        .expiryDate(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS);
    License license = SelfGeneratedLicense.create(specBuilder, License.VERSION_CURRENT);
    XContentBuilder builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, new
ToXContent.MapParams(Collections.singletonMap(License.REST_VIEW_MODE, "true")));
    builder.flush();
    Map<String, Object> map = XContentHelper.convertToMap(BytesReference.bytes(builder), false,
builder.contentType()).v2();

    // should have an extra status field, human readable issue_data and no expiry_date
    assertThat(map.get("status"), notNullValue());
    assertThat(map.get("type"), equalTo("basic"));
    assertThat(map.get("issue_date"), notNullValue());
    assertThat(map.get("expiry_date"), nullValue());
    assertThat(map.get("expiry_date_in_millis"), nullValue());
    assertThat(map.get("status"), equalTo("active"));
    builder = XContentFactory.contentBuilder(XContentType.JSON);
    license.toXContent(builder, ToXContent.EMPTY_PARAMS);
    builder.flush();
    map = XContentHelper.convertToMap(BytesReference.bytes(builder), false, builder.contentType()).v2();
    assertThat(map.get("status"), nullValue());
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchSecurityException;
import org.elasticsearch.test.ESTestCase;

import java.util.Arrays;

import static org.hamcrest.Matchers.contains;
import static org.hamcrest.Matchers.equalTo;
```

```
import static org.hamcrest.Matchers.hasSize;

public class LicenseUtilsTests extends ESTestCase {

  public void testNewExpirationException() {
    for (String feature : Arrays.asList("feature", randomAlphaOfLength(5), null, "")) {
      ElasticsearchSecurityException exception = LicenseUtils.newComplianceException(feature);
      assertNotNull(exception);
      assertThat(exception.getMetadataKeys(), contains(LicenseUtils.EXPIRED_FEATURE_METADATA));
      assertThat(exception.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA), hasSize(1));
      assertThat(exception.getMetadata(LicenseUtils.EXPIRED_FEATURE_METADATA).iterator().next(),
equalTo(feature));
    }
  }

  public void testIsLicenseExpiredException() {
    ElasticsearchSecurityException exception = LicenseUtils.newComplianceException("feature");
    assertTrue(LicenseUtils.isLicenseExpiredException(exception));

    exception = new ElasticsearchSecurityException("msg");
    assertFalse(LicenseUtils.isLicenseExpiredException(exception));
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.analysis.common.CommonAnalysisPlugin;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.test.ESIntegTestCase;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackClientPlugin;
import org.elasticsearch.xpack.core.XPackSettings;

import java.util.Arrays;
import java.util.Collection;
import java.util.concurrent.CountDownLatch;

public abstract class AbstractLicensesIntegrationTestCase extends ESIntegTestCase {
```

```
    @Override
    protected Settings nodeSettings(int nodeOrdinal) {
        return Settings.builder().put(XPackSettings.SECURITY_ENABLED.getKey(), false).build();
    }

    @Override
    protected Collection<Class<? extends Plugin>> nodePlugins() {
        return Arrays.asList(LocalStateCompositeXPackPlugin.class, CommonAnalysisPlugin.class);
    }

    @Override
    protected Collection<Class<? extends Plugin>> transportClientPlugins() {
        return Arrays.asList(XPackClientPlugin.class, CommonAnalysisPlugin.class);
    }

    @Override
    protected Settings transportClientSettings() {
        // Plugin should be loaded on the transport client as well
        return nodeSettings(0);
    }

    protected void putLicense(final License license) throws InterruptedException {
        final CountDownLatch latch = new CountDownLatch(1);
        ClusterService clusterService = internalCluster().getInstance(ClusterService.class,
internalCluster().getMasterName());
        clusterService.submitStateUpdateTask("putting license", new ClusterStateUpdateTask() {
            @Override
            public void clusterStateProcessed(String source, ClusterState oldState, ClusterState newState) {
                latch.countDown();
            }

            @Override
            public ClusterState execute(ClusterState currentState) throws Exception {
                MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
                mdBuilder.putCustom(LicensesMetaData.TYPE, new LicensesMetaData(license, null));
                return ClusterState.builder(currentState).metaData(mdBuilder).build();
            }

            @Override
            public void onFailure(String source, @Nullable Exception e) {
                logger.error("error on metaData cleanup after test", e);
            }
        });
        latch.await();
    }

    protected void putLicenseTombstone() throws InterruptedException {
```

```
        putLicense(LicensesMetaData.LICENSE_TOMBSTONE);
    }

    protected void wipeAllLicenses() throws InterruptedException {
        final CountDownLatch latch = new CountDownLatch(1);
        ClusterService clusterService = internalCluster().getInstance(ClusterService.class,
internalCluster().getMasterName());
        clusterService.submitStateUpdateTask("delete licensing metadata", new ClusterStateUpdateTask() {
            @Override
            public void clusterStateProcessed(String source, ClusterState oldState, ClusterState newState) {
                latch.countDown();
            }

            @Override
            public ClusterState execute(ClusterState currentState) throws Exception {
                MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
                mdBuilder.removeCustom(LicensesMetaData.TYPE);
                return ClusterState.builder(currentState).metaData(mdBuilder).build();
            }

            @Override
            public void onFailure(String source, @Nullable Exception e) {
                logger.error("error on metaData cleanup after test", e);
            }
        });
        latch.await();
    }

    protected void assertLicenseActive(boolean active) throws InterruptedException {
        boolean success = awaitBusy(() -> {
            for (XPackLicenseState licenseState : internalCluster().getDataNodeInstances(XPackLicenseState.class)) {
                if (licenseState.isActive() == active) {
                    return true;
                }
            }
            return false;
        });
        assertTrue(success);
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import org.elasticsearch.client.Request;
import org.elasticsearch.client.Response;
import org.elasticsearch.client.ResponseException;
import org.elasticsearch.client.RestClient;
import org.elasticsearch.common.io.Streams;
import org.elasticsearch.common.network.NetworkModule;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.test.ESIntegTestCase;
import org.elasticsearch.transport.Netty4Plugin;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackClientPlugin;

import java.io.InputStreamReader;
import java.nio.charset.StandardCharsets;
import java.util.Arrays;
import java.util.Collection;

import static org.elasticsearch.test.ESIntegTestCase.Scope.SUITE;

@ESIntegTestCase.ClusterScope(scope = SUITE)
public class StartBasicLicenseTests extends AbstractLicensesIntegrationTestCase {

  @Override
  protected Settings nodeSettings(int nodeOrdinal) {
    return Settings.builder()
          .put(super.nodeSettings(nodeOrdinal))
          .put("node.data", true)
          .put(LicenseService.SELF_GENERATED_LICENSE_TYPE.getKey(), "basic")
          .put(NetworkModule.HTTP_ENABLED.getKey(), true).build();
  }

  @Override
  protected Collection<Class<? extends Plugin>> nodePlugins() {
    return Arrays.asList(LocalStateCompositeXPackPlugin.class, Netty4Plugin.class);
  }

  @Override
  protected Collection<Class<? extends Plugin>> transportClientPlugins() {
    return Arrays.asList(XPackClientPlugin.class, Netty4Plugin.class);
  }

  public void testStartBasicLicense() throws Exception {
    LicensingClient licensingClient = new LicensingClient(client());
    License license = TestUtils.generateSignedLicense("trial",  License.VERSION_CURRENT, -1,
TimeValue.timeValueHours(24));
    licensingClient.preparePutLicense(license).get();
```

```
    assertBusy(() -> {
        GetLicenseResponse getLicenseResponse = licensingClient.prepareGetLicense().get();
        assertEquals("trial", getLicenseResponse.license().type());
    });

    // Testing that you can start a basic license when you have no license
    if (randomBoolean()) {
        licensingClient.prepareDeleteLicense().get();
        assertNull(licensingClient.prepareGetLicense().get().license());
    }

    RestClient restClient = getRestClient();
    Response response = restClient.performRequest(new Request("GET", "/_xpack/license/basic_status"));
    String body = Streams.copyToString(new InputStreamReader(response.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response.getStatusLine().getStatusCode());
    assertEquals("{\"eligible_to_start_basic\":true}", body);

    Request ackRequest = new Request("POST", "/_xpack/license/start_basic");
    ackRequest.addParameter("acknowledge", "true");
    Response response2 = restClient.performRequest(ackRequest);
    String body2 = Streams.copyToString(new InputStreamReader(response2.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response2.getStatusLine().getStatusCode());
    assertTrue(body2.contains("\"acknowledged\":true"));
    assertTrue(body2.contains("\"basic_was_started\":true"));

    assertBusy(() -> {
        GetLicenseResponse currentLicense = licensingClient.prepareGetLicense().get();
        assertEquals("basic", currentLicense.license().type());
    });

    long expirationMillis = licensingClient.prepareGetLicense().get().license().expiryDate();
    assertEquals(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS,
expirationMillis);

    Response response3 = restClient.performRequest(new Request("GET", "/_xpack/license"));
    String body3 = Streams.copyToString(new InputStreamReader(response3.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertTrue(body3.contains("\"type\" : \"basic\""));
    assertFalse(body3.contains("expiry_date"));
    assertFalse(body3.contains("expiry_date_in_millis"));

    Response response4 = restClient.performRequest(new Request("GET", "/_xpack/license/basic_status"));
    String body4 = Streams.copyToString(new InputStreamReader(response4.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response3.getStatusLine().getStatusCode());
```

```
        assertEquals("{\"eligible_to_start_basic\":false}", body4);

        ResponseException ex = expectThrows(ResponseException.class,
                () -> restClient.performRequest(new Request("POST", "/_xpack/license/start_basic")));
        Response response5 = ex.getResponse();
        String body5 = Streams.copyToString(new InputStreamReader(response5.getEntity().getContent(),
StandardCharsets.UTF_8));
        assertEquals(403, response5.getStatusLine().getStatusCode());
        assertTrue(body5.contains("\"basic_was_started\":false"));
        assertTrue(body5.contains("\"acknowledged\":true"));
        assertTrue(body5.contains("\"error_message\":\"Operation failed: Current license is basic.\""));
    }

    public void testUnacknowledgedStartBasicLicense() throws Exception {
        LicensingClient licensingClient = new LicensingClient(client());
        License license = TestUtils.generateSignedLicense("trial",  License.VERSION_CURRENT, -1,
TimeValue.timeValueHours(24));
        licensingClient.preparePutLicense(license).get();

        assertBusy(() -> {
            GetLicenseResponse getLicenseResponse = licensingClient.prepareGetLicense().get();
            assertEquals("trial", getLicenseResponse.license().type());
        });

        Response response2 = getRestClient().performRequest(new Request("POST", "/_xpack/license/start_basic"));
        String body2 = Streams.copyToString(new InputStreamReader(response2.getEntity().getContent(),
StandardCharsets.UTF_8));
        assertEquals(200, response2.getStatusLine().getStatusCode());
        assertTrue(body2.contains("\"acknowledged\":false"));
        assertTrue(body2.contains("\"basic_was_started\":false"));
        assertTrue(body2.contains("\"error_message\":\"Operation failed: Needs acknowledgement.\""));
        assertTrue(body2.contains("\"message\":\"This license update requires acknowledgement. To acknowledge the
license, " +
                "please read the following messages and call /start_basic again, this time with the
\\\"acknowledge=true\\\""));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.cluster.ClusterChangedEvent;
import org.elasticsearch.cluster.ClusterName;
import org.elasticsearch.cluster.ClusterState;
```

```
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.node.DiscoveryNode;
import org.elasticsearch.cluster.node.DiscoveryNodes;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.junit.After;
import org.junit.Before;
import org.mockito.ArgumentCaptor;

import static java.util.Collections.emptyMap;
import static java.util.Collections.emptySet;
import static org.hamcrest.Matchers.equalTo;
import static org.mockito.Matchers.any;
import static org.mockito.Mockito.times;
import static org.mockito.Mockito.verify;
import static org.mockito.Mockito.when;

public class LicenseClusterChangeTests extends AbstractLicenseServiceTestCase {

  private TestUtils.AssertingLicenseState licenseState;

  @Before
  public void setup() {
    licenseState = new TestUtils.AssertingLicenseState();
    setInitialState(null, licenseState, Settings.EMPTY);
    licenseService.start();
  }

  @After
  public void teardown() {
    licenseService.stop();
  }


  public void testNotificationOnNewLicense() throws Exception {
    ClusterState oldState = ClusterState.builder(new ClusterName("a")).build();
    final License license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(24));
    MetaData metaData = MetaData.builder().putCustom(LicensesMetaData.TYPE, new
LicensesMetaData(license, null)).build();
    ClusterState newState = ClusterState.builder(new ClusterName("a")).metaData(metaData).build();
    licenseService.clusterChanged(new ClusterChangedEvent("simulated", newState, oldState));
    assertThat(licenseState.activeUpdates.size(), equalTo(1));
    assertTrue(licenseState.activeUpdates.get(0));
  }

  public void testNoNotificationOnExistingLicense() throws Exception {
    final License license = TestUtils.generateSignedLicense(TimeValue.timeValueHours(24));
```

```
        MetaData metaData = MetaData.builder().putCustom(LicensesMetaData.TYPE, new
LicensesMetaData(license, null)).build();
        ClusterState newState = ClusterState.builder(new ClusterName("a")).metaData(metaData).build();
        ClusterState oldState = ClusterState.builder(newState).build();
        licenseService.clusterChanged(new ClusterChangedEvent("simulated", newState, oldState));
        assertThat(licenseState.activeUpdates.size(), equalTo(0));
    }

    public void testSelfGeneratedLicenseGeneration() throws Exception {
        DiscoveryNode master = new DiscoveryNode("b", buildNewFakeTransportAddress(), emptyMap(), emptySet(),
Version.CURRENT);
        ClusterState oldState = ClusterState.builder(new ClusterName("a"))
                .nodes(DiscoveryNodes.builder().masterNodeId(master.getId()).add(master)).build();
        when(discoveryNodes.isLocalNodeElectedMaster()).thenReturn(true);
        ClusterState newState = ClusterState.builder(oldState).nodes(discoveryNodes).build();

        licenseService.clusterChanged(new ClusterChangedEvent("simulated", newState, oldState));
        ArgumentCaptor<ClusterStateUpdateTask> stateUpdater =
ArgumentCaptor.forClass(ClusterStateUpdateTask.class);
        verify(clusterService, times(1)).submitStateUpdateTask(any(), stateUpdater.capture());
        ClusterState stateWithLicense = stateUpdater.getValue().execute(newState);
        LicensesMetaData licenseMetaData = stateWithLicense.metaData().custom(LicensesMetaData.TYPE);
        assertNotNull(licenseMetaData);
        assertNotNull(licenseMetaData.getLicense());
        assertEquals(licenseType, licenseMetaData.getLicense().type());
        long expiration;
        if (licenseType.equals("basic")) {
            expiration = LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS;
        } else {
            expiration = LicenseService.NON_BASIC_SELF_GENERATED_LICENSE_DURATION.millis() +
clock.millis();
        }
        assertEquals(expiration, licenseMetaData.getLicense().expiryDate());
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.client.Request;
import org.elasticsearch.client.Response;
import org.elasticsearch.client.ResponseException;
import org.elasticsearch.client.RestClient;
import org.elasticsearch.common.io.Streams;
import org.elasticsearch.common.network.NetworkModule;
```

```
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.plugins.Plugin;
import org.elasticsearch.test.ESIntegTestCase;
import org.elasticsearch.transport.Netty4Plugin;
import org.elasticsearch.xpack.core.LocalStateCompositeXPackPlugin;
import org.elasticsearch.xpack.core.XPackClientPlugin;

import java.io.InputStreamReader;
import java.nio.charset.StandardCharsets;
import java.util.Arrays;
import java.util.Collection;

import static org.elasticsearch.test.ESIntegTestCase.Scope.SUITE;

@ESIntegTestCase.ClusterScope(scope = SUITE)
public class StartTrialLicenseTests extends AbstractLicensesIntegrationTestCase {

  @Override
  protected Settings nodeSettings(int nodeOrdinal) {
    return Settings.builder()
        .put(super.nodeSettings(nodeOrdinal))
        .put("node.data", true)
        .put(LicenseService.SELF_GENERATED_LICENSE_TYPE.getKey(), "basic")
        .put(NetworkModule.HTTP_ENABLED.getKey(), true).build();
  }

  @Override
  protected Collection<Class<? extends Plugin>> nodePlugins() {
    return Arrays.asList(LocalStateCompositeXPackPlugin.class, Netty4Plugin.class);
  }

  @Override
  protected Collection<Class<? extends Plugin>> transportClientPlugins() {
    return Arrays.asList(XPackClientPlugin.class, Netty4Plugin.class);
  }

  public void testStartTrial() throws Exception {
    LicensingClient licensingClient = new LicensingClient(client());
    ensureStartingWithBasic();

    RestClient restClient = getRestClient();
    Response response = restClient.performRequest(new Request("GET", "/_xpack/license/trial_status"));
    String body = Streams.copyToString(new InputStreamReader(response.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response.getStatusLine().getStatusCode());
    assertEquals("{\"eligible_to_start_trial\":true}", body);

    // Test that starting will fail without acknowledgement
```

```
    Response response2 = restClient.performRequest(new Request("POST", "/_xpack/license/start_trial"));
    String body2 = Streams.copyToString(new InputStreamReader(response2.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response2.getStatusLine().getStatusCode());
    assertTrue(body2.contains("\"trial_was_started\":false"));
    assertTrue(body2.contains("\"error_message\":\"Operation failed: Needs acknowledgement.\""));
    assertTrue(body2.contains("\"acknowledged\":false"));

    assertBusy(() -> {
        GetLicenseResponse getLicenseResponse = licensingClient.prepareGetLicense().get();
        assertEquals("basic", getLicenseResponse.license().type());
    });

    String type = randomFrom(LicenseService.VALID_TRIAL_TYPES);

    Request ackRequest = new Request("POST", "/_xpack/license/start_trial");
    ackRequest.addParameter("acknowledge", "true");
    ackRequest.addParameter("type", type);
    Response response3 = restClient.performRequest(ackRequest);
    String body3 = Streams.copyToString(new InputStreamReader(response3.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response3.getStatusLine().getStatusCode());
    assertTrue(body3.contains("\"trial_was_started\":true"));
    assertTrue(body3.contains("\"type\":\"" + type + "\""));
    assertTrue(body3.contains("\"acknowledged\":true"));

    assertBusy(() -> {
        GetLicenseResponse postTrialLicenseResponse = licensingClient.prepareGetLicense().get();
        assertEquals(type, postTrialLicenseResponse.license().type());
    });

    Response response4 = restClient.performRequest(new Request("GET", "/_xpack/license/trial_status"));
    String body4 = Streams.copyToString(new InputStreamReader(response4.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(200, response4.getStatusLine().getStatusCode());
    assertEquals("{\"eligible_to_start_trial\":false}", body4);

    String secondAttemptType = randomFrom(LicenseService.VALID_TRIAL_TYPES);

    Request startTrialWhenStartedRequest = new Request("POST", "/_xpack/license/start_trial");
    startTrialWhenStartedRequest.addParameter("acknowledge", "true");
    startTrialWhenStartedRequest.addParameter("type", secondAttemptType);
    ResponseException ex = expectThrows(ResponseException.class, () ->
restClient.performRequest(startTrialWhenStartedRequest));
    Response response5 = ex.getResponse();
    String body5 = Streams.copyToString(new InputStreamReader(response5.getEntity().getContent(),
StandardCharsets.UTF_8));
    assertEquals(403, response5.getStatusLine().getStatusCode());
```

```
        assertTrue(body5.contains("\"trial_was_started\":false"));
        assertTrue(body5.contains("\"error_message\":\"Operation failed: Trial was already activated.\""));
    }

    public void testInvalidType() throws Exception {
        ensureStartingWithBasic();

        Request request = new Request("POST", "/_xpack/license/start_trial");
        request.addParameter("type", "basic");
        ResponseException ex = expectThrows(ResponseException.class, () ->
getRestClient().performRequest(request));
        Response response = ex.getResponse();
        String body = Streams.copyToString(new InputStreamReader(response.getEntity().getContent(),
StandardCharsets.UTF_8));
        assertEquals(400, response.getStatusLine().getStatusCode());
        assertTrue(body.contains("\"type\":\"illegal_argument_exception\""));
        assertTrue(body.contains("\"reason\":\"Cannot start trial of type [basic]. Valid trial types are ["));
    }

    private void ensureStartingWithBasic() throws Exception {
        LicensingClient licensingClient = new LicensingClient(client());
        GetLicenseResponse getLicenseResponse = licensingClient.prepareGetLicense().get();

        if ("basic".equals(getLicenseResponse.license().type()) == false) {
            licensingClient.preparePostStartBasic().setAcknowledge(true).get();
        }

        assertBusy(() -> {
            GetLicenseResponse postTrialLicenseResponse = licensingClient.prepareGetLicense().get();
            assertEquals("basic", postTrialLicenseResponse.license().type());
        });
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.apache.lucene.index.DirectoryReader;
import org.apache.lucene.util.Accountable;
import org.apache.lucene.util.RamUsageEstimator;
import org.apache.lucene.util.automaton.Automata;
import org.apache.lucene.util.automaton.Automaton;
import org.apache.lucene.util.automaton.CharacterRunAutomaton;
import org.apache.lucene.util.automaton.MinimizationOperations;
import org.apache.lucene.util.automaton.Operations;
```

```
import org.elasticsearch.ElasticsearchSecurityException;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.regex.Regex;
import org.elasticsearch.index.mapper.AllFieldMapper;
import org.elasticsearch.xpack.core.security.authz.accesscontrol.FieldSubsetReader;
import org.elasticsearch.xpack.core.security.authz.permission.FieldPermissionsDefinition.FieldGrantExcludeGroup;
import org.elasticsearch.xpack.core.security.support.Automatons;

import java.io.IOException;
import java.util.Arrays;
import java.util.HashMap;
import java.util.HashSet;
import java.util.List;
import java.util.Set;
import java.util.stream.Collectors;

import static org.apache.lucene.util.automaton.Operations.subsetOf;

/**
* Stores patterns to fields which access is granted or denied to and maintains an automaton that can be used to check
if permission is
* allowed for a specific field.
* Field permissions are configured via a list of strings that are patterns a field has to match. Two lists determine
whether or
* not a field is granted access to:
* 1. It has to match the patterns in grantedFieldsArray
* 2. it must not match the patterns in deniedFieldsArray
*/
public final class FieldPermissions implements Accountable {

  public static final FieldPermissions DEFAULT = new FieldPermissions();

  private static final long BASE_FIELD_PERM_DEF_BYTES = RamUsageEstimator.shallowSizeOf(new
FieldPermissionsDefinition(null, null));
  private static final long BASE_FIELD_GROUP_BYTES = RamUsageEstimator.shallowSizeOf(new
FieldGrantExcludeGroup(null, null));
  private static final long BASE_HASHSET_SIZE = RamUsageEstimator.shallowSizeOfInstance(HashSet.class);
  private static final long BASE_HASHSET_ENTRY_SIZE;
  static {
    HashMap<String, Object> map = new HashMap<>();
    map.put(FieldPermissions.class.getName(), new Object());
    long mapEntryShallowSize = RamUsageEstimator.shallowSizeOf(map.entrySet().iterator().next());
    // assume a load factor of 50%
    // for each entry, we need two object refs, one for the entry itself
    // and one for the free space that is due to the fact hash tables can
    // not be fully loaded
    BASE_HASHSET_ENTRY_SIZE = mapEntryShallowSize + 2 *
RamUsageEstimator.NUM_BYTES_OBJECT_REF;
```

```
    }

    private final FieldPermissionsDefinition fieldPermissionsDefinition;
    // an automaton that represents a union of one more sets of permitted and denied fields
    private final CharacterRunAutomaton permittedFieldsAutomaton;
    private final boolean permittedFieldsAutomatonIsTotal;
    private final Automaton originalAutomaton;

    private final long ramBytesUsed;

    /** Constructor that does not enable field-level security: all fields are accepted. */
    public FieldPermissions() {
        this(new FieldPermissionsDefinition(null, null), Automatons.MATCH_ALL);
    }

    /** Constructor that enables field-level security based on include/exclude rules. Exclude rules
     *  have precedence over include rules. */
    public FieldPermissions(FieldPermissionsDefinition fieldPermissionsDefinition) {
        this(fieldPermissionsDefinition, initializePermittedFieldsAutomaton(fieldPermissionsDefinition));
    }

    /** Constructor that enables field-level security based on include/exclude rules. Exclude rules
     *  have precedence over include rules. */
    FieldPermissions(FieldPermissionsDefinition fieldPermissionsDefinition, Automaton permittedFieldsAutomaton)
  {
        if (permittedFieldsAutomaton.isDeterministic() == false && permittedFieldsAutomaton.getNumStates() > 1) {
            // we only accept deterministic automata so that the CharacterRunAutomaton constructor
            // directly wraps the provided automaton
            throw new IllegalArgumentException("Only accepts deterministic automata");
        }
        this.fieldPermissionsDefinition = fieldPermissionsDefinition;
        this.originalAutomaton = permittedFieldsAutomaton;
        this.permittedFieldsAutomaton = new CharacterRunAutomaton(permittedFieldsAutomaton);
        // we cache the result of isTotal since this might be a costly operation
        this.permittedFieldsAutomatonIsTotal = Operations.isTotal(permittedFieldsAutomaton);

        long ramBytesUsed = BASE_FIELD_PERM_DEF_BYTES;

        for (FieldGrantExcludeGroup group : fieldPermissionsDefinition.getFieldGrantExcludeGroups()) {
            ramBytesUsed += BASE_FIELD_GROUP_BYTES + BASE_HASHSET_ENTRY_SIZE;
            if (group.getGrantedFields() != null) {
                ramBytesUsed += RamUsageEstimator.shallowSizeOf(group.getGrantedFields());
            }
            if (group.getExcludedFields() != null) {
                ramBytesUsed += RamUsageEstimator.shallowSizeOf(group.getExcludedFields());
            }
        }
        ramBytesUsed += permittedFieldsAutomaton.ramBytesUsed();
```

```
        ramBytesUsed += runAutomatonRamBytesUsed(permittedFieldsAutomaton);
        this.ramBytesUsed = ramBytesUsed;
    }

    /**
     * Return an estimation of the ram bytes used by a {@link CharacterRunAutomaton}
     * that wraps the given automaton.
     */
    private static long runAutomatonRamBytesUsed(Automaton a) {
        return a.getNumStates() * 5; // wild guess, better than 0
    }

    public static Automaton initializePermittedFieldsAutomaton(FieldPermissionsDefinition
fieldPermissionsDefinition) {
        Set<FieldGrantExcludeGroup> groups = fieldPermissionsDefinition.getFieldGrantExcludeGroups();
        assert groups.size() > 0 : "there must always be a single group for field inclusion/exclusion";
        List<Automaton> automatonList =
                groups.stream()
                        .map(g -> FieldPermissions.initializePermittedFieldsAutomaton(g.getGrantedFields(),
g.getExcludedFields()))
                        .collect(Collectors.toList());
        return Automatons.unionAndMinimize(automatonList);
    }

    private static Automaton initializePermittedFieldsAutomaton(final String[] grantedFields, final String[]
deniedFields) {
        Automaton grantedFieldsAutomaton;
        if (grantedFields == null || Arrays.stream(grantedFields).anyMatch(Regex::isMatchAllPattern)) {
            grantedFieldsAutomaton = Automatons.MATCH_ALL;
        } else {
            // an automaton that includes metadata fields, including join fields created by the _parent field such
            // as _parent#type
            Automaton metaFieldsAutomaton = Operations.concatenate(Automata.makeChar('_'),
Automata.makeAnyString());
            grantedFieldsAutomaton = Operations.union(Automatons.patterns(grantedFields), metaFieldsAutomaton);
        }

        Automaton deniedFieldsAutomaton;
        if (deniedFields == null || deniedFields.length == 0) {
            deniedFieldsAutomaton = Automatons.EMPTY;
        } else {
            deniedFieldsAutomaton = Automatons.patterns(deniedFields);
        }

        grantedFieldsAutomaton = MinimizationOperations.minimize(grantedFieldsAutomaton,
Operations.DEFAULT_MAX_DETERMINIZED_STATES);
        deniedFieldsAutomaton = MinimizationOperations.minimize(deniedFieldsAutomaton,
Operations.DEFAULT_MAX_DETERMINIZED_STATES);
```

```
        if (subsetOf(deniedFieldsAutomaton, grantedFieldsAutomaton) == false) {
            throw new ElasticsearchSecurityException("Exceptions for field permissions must be a subset of the " +
                    "granted fields but " + Strings.arrayToCommaDelimitedString(deniedFields) + " is not a subset of " +
                    Strings.arrayToCommaDelimitedString(grantedFields));
        }

        if (!containsAllField(grantedFields) && !containsAllField(deniedFields)) {
            // It is not explicitly stated whether _all should be allowed/denied
            // In that case we automatically disable _all, unless all fields would match
            if (Operations.isTotal(grantedFieldsAutomaton) && Operations.isEmpty(deniedFieldsAutomaton)) {
                // all fields are accepted, so using _all is fine
            } else {
                deniedFieldsAutomaton = Operations.union(deniedFieldsAutomaton,
Automata.makeString(AllFieldMapper.NAME));
            }
        }

        grantedFieldsAutomaton = Automatons.minusAndMinimize(grantedFieldsAutomaton,
deniedFieldsAutomaton);
        return grantedFieldsAutomaton;
    }

    private static boolean containsAllField(String[] fields) {
        return fields != null && Arrays.stream(fields).anyMatch(AllFieldMapper.NAME::equals);
    }

    /**
     * Returns true if this field permission policy allows access to the field and false if not.
     * fieldName can be a wildcard.
     */
    public boolean grantsAccessTo(String fieldName) {
        return permittedFieldsAutomatonIsTotal || permittedFieldsAutomaton.run(fieldName);
    }

    public FieldPermissionsDefinition getFieldPermissionsDefinition() {
        return fieldPermissionsDefinition;
    }

    /** Return whether field-level security is enabled, ie. whether any field might be filtered out. */
    public boolean hasFieldLevelSecurity() {
        return permittedFieldsAutomatonIsTotal == false;
    }

    /** Return a wrapped reader that only exposes allowed fields. */
    public DirectoryReader filter(DirectoryReader reader) throws IOException {
        if (hasFieldLevelSecurity() == false) {
            return reader;
```

```
        }
        return FieldSubsetReader.wrap(reader, permittedFieldsAutomaton);
    }

    // for testing only
    Automaton getIncludeAutomaton() {
        return originalAutomaton;
    }

    @Override
    public boolean equals(Object o) {
        if (this == o) return true;
        if (o == null || getClass() != o.getClass()) return false;

        FieldPermissions that = (FieldPermissions) o;

        if (permittedFieldsAutomatonIsTotal != that.permittedFieldsAutomatonIsTotal) return false;
        return fieldPermissionsDefinition != null ?
                fieldPermissionsDefinition.equals(that.fieldPermissionsDefinition) : that.fieldPermissionsDefinition ==
null;
    }

    @Override
    public int hashCode() {
        int result = fieldPermissionsDefinition != null ? fieldPermissionsDefinition.hashCode() : 0;
        result = 31 * result + (permittedFieldsAutomatonIsTotal ? 1 : 0);
        return result;
    }

    @Override
    public long ramBytesUsed() {
        return ramBytesUsed;
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.elasticsearch.xpack.core.security.authz.privilege.Privilege;

import java.util.function.Predicate;

/**
* A permissions that is based on a general privilege that contains patterns of users that this
* user can execute a request as
```

```
*/
public final class RunAsPermission {

  public static final RunAsPermission NONE = new RunAsPermission(Privilege.NONE);

  private final Privilege privilege;
  private final Predicate<String> predicate;

  RunAsPermission(Privilege privilege) {
      this.privilege = privilege;
      this.predicate = privilege.predicate();
  }

  public Privilege getPrivilege() {
      return privilege;
  }

  /**
   * Checks if this permission grants run as to the specified user
   */
  public boolean check(String username) {
      return predicate.test(username);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.apache.lucene.util.automaton.Automaton;
import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.common.cache.Cache;
import org.elasticsearch.common.cache.CacheBuilder;
import org.elasticsearch.common.settings.Setting;
import org.elasticsearch.common.settings.Setting.Property;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.xpack.core.security.authz.permission.FieldPermissionsDefinition.FieldGrantExcludeGroup;
import org.elasticsearch.xpack.core.security.support.Automatons;

import java.util.Collection;
import java.util.List;
import java.util.Optional;
import java.util.Set;
import java.util.concurrent.ExecutionException;
import java.util.function.Predicate;
import java.util.stream.Collectors;
```

```
import static org.elasticsearch.xpack.core.security.SecurityField.setting;

/**
* A service for managing the caching of {@link FieldPermissions} as these may often need to be combined or
created and internally they
* use an {@link org.apache.lucene.util.automaton.Automaton}, which can be costly to create once you account for
minimization
*/
public final class FieldPermissionsCache {

  public static final Setting<Long> CACHE_SIZE_SETTING = Setting.longSetting(
        setting("authz.store.roles.field_permissions.cache.max_size_in_bytes"), 100 * 1024 * 1024, -1L,
Property.NodeScope);
  private final Cache<FieldPermissionsDefinition, FieldPermissions> cache;

  public FieldPermissionsCache(Settings settings) {
     this.cache = CacheBuilder.<FieldPermissionsDefinition, FieldPermissions>builder()
            .setMaximumWeight(CACHE_SIZE_SETTING.get(settings))
            .weigher((key, fieldPermissions) -> fieldPermissions.ramBytesUsed())
            .build();
  }

  /**
   * Gets a {@link FieldPermissions} instance that corresponds to the granted and denied parameters. The instance
may come from the cache
   * or if it gets created, the instance will be cached
   */
  FieldPermissions getFieldPermissions(String[] granted, String[] denied) {
     return getFieldPermissions(new FieldPermissionsDefinition(granted, denied));
  }

  /**
   * Gets a {@link FieldPermissions} instance that corresponds to the granted and denied parameters. The instance
may come from the cache
   * or if it gets created, the instance will be cached
   */
  public FieldPermissions getFieldPermissions(FieldPermissionsDefinition fieldPermissionsDefinition) {
     try {
        return cache.computeIfAbsent(fieldPermissionsDefinition,
              (key) -> new FieldPermissions(key, FieldPermissions.initializePermittedFieldsAutomaton(key)));
     } catch (ExecutionException e) {
        throw new ElasticsearchException("unable to compute field permissions", e);
     }
  }

  /**
   * Returns a field permissions object that corresponds to the merging of the given field permissions and caches the
```

```
instance if one was
    * not found in the cache.
    */
   FieldPermissions getFieldPermissions(Collection<FieldPermissions> fieldPermissionsCollection) {
      Optional<FieldPermissions> allowAllFieldPermissions = fieldPermissionsCollection.stream()
            .filter(((Predicate<FieldPermissions>) (FieldPermissions::hasFieldLevelSecurity)).negate())
            .findFirst();
      return allowAllFieldPermissions.orElseGet(() -> {
         final Set<FieldGrantExcludeGroup> fieldGrantExcludeGroups = fieldPermissionsCollection.stream()
               .flatMap(fieldPermission ->
fieldPermission.getFieldPermissionsDefinition().getFieldGrantExcludeGroups().stream())
               .collect(Collectors.toSet());
         final FieldPermissionsDefinition combined = new FieldPermissionsDefinition(fieldGrantExcludeGroups);
         try {
            return cache.computeIfAbsent(combined, (key) -> {
               List<Automaton> automatonList = fieldPermissionsCollection.stream()
                           .map(FieldPermissions::getIncludeAutomaton)
                           .collect(Collectors.toList());
               return new FieldPermissions(key, Automatons.unionAndMinimize(automatonList));
            });
         } catch (ExecutionException e) {
            throw new ElasticsearchException("unable to compute field permissions", e);
         }
      });
   }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.transport.TransportRequest;
import org.elasticsearch.xpack.core.security.authz.privilege.ClusterPrivilege;
import org.elasticsearch.xpack.core.security.authz.privilege.ConditionalClusterPrivilege;

import java.util.Collection;
import java.util.Collections;
import java.util.List;
import java.util.Set;
import java.util.function.Predicate;
import java.util.stream.Collectors;

/**
* A permission that is based on privileges for cluster wide actions, with the optional ability to inspect the request
object
```

```
*/
public abstract class ClusterPermission {
  private final ClusterPrivilege privilege;

  ClusterPermission(ClusterPrivilege privilege) {
    this.privilege = privilege;
  }

  public ClusterPrivilege privilege() {
    return privilege;
  }

  public abstract boolean check(String action, TransportRequest request);

  public abstract List<Tuple<ClusterPrivilege, ConditionalClusterPrivilege>> privileges();

  /**
   * A permission that is based solely on cluster privileges and does not consider request state
   */
  public static class SimpleClusterPermission extends ClusterPermission {

    public static final SimpleClusterPermission NONE = new SimpleClusterPermission(ClusterPrivilege.NONE);

    private final Predicate<String> predicate;

    SimpleClusterPermission(ClusterPrivilege privilege) {
      super(privilege);
      this.predicate = privilege.predicate();
    }

    @Override
    public boolean check(String action, TransportRequest request) {
      return predicate.test(action);
    }

    @Override
    public List<Tuple<ClusterPrivilege, ConditionalClusterPrivilege>> privileges() {
      return Collections.singletonList(new Tuple<>(super.privilege, null));
    }
  }

  /**
   * A permission that makes use of both cluster privileges and request inspection
   */
  public static class ConditionalClusterPermission extends ClusterPermission {
    private final ConditionalClusterPrivilege conditionalPrivilege;

    public ConditionalClusterPermission(ConditionalClusterPrivilege conditionalPrivilege) {
```

```
        super(conditionalPrivilege.getPrivilege());
        this.conditionalPrivilege = conditionalPrivilege;
    }

    @Override
    public boolean check(String action, TransportRequest request) {
        return super.privilege.predicate().test(action) && conditionalPrivilege.getRequestPredicate().test(request);
    }

    @Override
    public List<Tuple<ClusterPrivilege, ConditionalClusterPrivilege>> privileges() {
        return Collections.singletonList(new Tuple<>(super.privilege, conditionalPrivilege));
    }
  }

  /**
   * A permission that composes a number of other cluster permissions
   */
  public static class CompositeClusterPermission extends ClusterPermission {
    private final Collection<ClusterPermission> children;

    public CompositeClusterPermission(Collection<ClusterPermission> children) {
        super(buildPrivilege(children));
        this.children = children;
    }

    private static ClusterPrivilege buildPrivilege(Collection<ClusterPermission> children) {
        final Set<String> names = children.stream()
            .map(ClusterPermission::privilege)
            .map(ClusterPrivilege::name)
            .flatMap(Set::stream)
            .collect(Collectors.toSet());
        return ClusterPrivilege.get(names);
    }

    @Override
    public List<Tuple<ClusterPrivilege, ConditionalClusterPrivilege>> privileges() {
        return children.stream().map(ClusterPermission::privileges).flatMap(List::stream).collect(Collectors.toList());
    }

    @Override
    public boolean check(String action, TransportRequest request) {
        return children.stream().anyMatch(p -> p.check(action, request));
    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
```

```
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.elasticsearch.common.Strings;

import java.util.Arrays;
import java.util.Collections;
import java.util.Set;

/**
* Represents the definition of a {@link FieldPermissions}. Field permissions are defined as a
* collections of grant and exclude definitions where the exclude definition must be a subset of
* the grant definition.
*/
public final class FieldPermissionsDefinition {

  private final Set<FieldGrantExcludeGroup> fieldGrantExcludeGroups;

  public FieldPermissionsDefinition(String[] grant, String[] exclude) {
     this(Collections.singleton(new FieldGrantExcludeGroup(grant, exclude)));
  }

  public FieldPermissionsDefinition(Set<FieldGrantExcludeGroup> fieldGrantExcludeGroups) {
     this.fieldGrantExcludeGroups = Collections.unmodifiableSet(fieldGrantExcludeGroups);
  }

  public Set<FieldGrantExcludeGroup> getFieldGrantExcludeGroups() {
     return fieldGrantExcludeGroups;
  }

  @Override
  public boolean equals(Object o) {
     if (this == o) return true;
     if (o == null || getClass() != o.getClass()) return false;

     FieldPermissionsDefinition that = (FieldPermissionsDefinition) o;

     return fieldGrantExcludeGroups != null ?
           fieldGrantExcludeGroups.equals(that.fieldGrantExcludeGroups) :
           that.fieldGrantExcludeGroups == null;
  }

  @Override
  public int hashCode() {
     return fieldGrantExcludeGroups != null ? fieldGrantExcludeGroups.hashCode() : 0;
  }
```

```
  public static final class FieldGrantExcludeGroup {
    private final String[] grantedFields;
    private final String[] excludedFields;

    public FieldGrantExcludeGroup(String[] grantedFields, String[] excludedFields) {
      this.grantedFields = grantedFields;
      this.excludedFields = excludedFields;
    }

    public String[] getGrantedFields() {
      return grantedFields;
    }

    public String[] getExcludedFields() {
      return excludedFields;
    }

    @Override
    public boolean equals(Object o) {
      if (this == o) return true;
      if (o == null || getClass() != o.getClass()) return false;

      FieldGrantExcludeGroup that = (FieldGrantExcludeGroup) o;

      if (!Arrays.equals(grantedFields, that.grantedFields)) return false;
      return Arrays.equals(excludedFields, that.excludedFields);
    }

    @Override
    public int hashCode() {
      int result = Arrays.hashCode(grantedFields);
      result = 31 * result + Arrays.hashCode(excludedFields);
      return result;
    }

    @Override
    public String toString() {
      return getClass().getSimpleName()
        + "[grant=" + Strings.arrayToCommaDelimitedString(grantedFields)
        + "; exclude=" + Strings.arrayToCommaDelimitedString(excludedFields)
        + "]";
    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
```

```
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.util.set.Sets;
import org.elasticsearch.xpack.core.security.authz.RoleDescriptor;
import org.elasticsearch.xpack.core.security.authz.accesscontrol.IndicesAccessControl;
import org.elasticsearch.xpack.core.security.authz.privilege.ApplicationPrivilege;
import org.elasticsearch.xpack.core.security.authz.privilege.ClusterPrivilege;
import org.elasticsearch.xpack.core.security.authz.privilege.ConditionalClusterPrivilege;
import org.elasticsearch.xpack.core.security.authz.privilege.IndexPrivilege;
import org.elasticsearch.xpack.core.security.authz.privilege.Privilege;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.Map;
import java.util.Objects;
import java.util.Set;

public final class Role {

  public static final Role EMPTY = Role.builder("__empty").build();

  private final String[] names;
  private final ClusterPermission cluster;
  private final IndicesPermission indices;
  private final ApplicationPermission application;
  private final RunAsPermission runAs;

  Role(String[] names, ClusterPermission cluster, IndicesPermission indices, ApplicationPermission application,
RunAsPermission runAs) {
      this.names = names;
      this.cluster = Objects.requireNonNull(cluster);
      this.indices = Objects.requireNonNull(indices);
      this.application = Objects.requireNonNull(application);
      this.runAs = Objects.requireNonNull(runAs);
  }

  public String[] names() {
      return names;
  }
```

```
    public ClusterPermission cluster() {
        return cluster;
    }

    public IndicesPermission indices() {
        return indices;
    }

    public ApplicationPermission application() {
        return application;
    }

    public RunAsPermission runAs() {
        return runAs;
    }

    public static Builder builder(String... names) {
        return new Builder(names, null);
    }

    public static Builder builder(String[] names, FieldPermissionsCache fieldPermissionsCache) {
        return new Builder(names, fieldPermissionsCache);
    }

    public static Builder builder(RoleDescriptor rd, FieldPermissionsCache fieldPermissionsCache) {
        return new Builder(rd, fieldPermissionsCache);
    }

    /**
     * Returns whether at least one group encapsulated by this indices permissions is authorized to execute the
     * specified action with the requested indices/aliases. At the same time if field and/or document level security
     * is configured for any group also the allowed fields and role queries are resolved.
     */
    public IndicesAccessControl authorize(String action, Set<String> requestedIndicesOrAliases, MetaData
metaData,
                                          FieldPermissionsCache fieldPermissionsCache) {
        Map<String, IndicesAccessControl.IndexAccessControl> indexPermissions = indices.authorize(
            action, requestedIndicesOrAliases, metaData, fieldPermissionsCache
        );

        // At least one role / indices permission set need to match with all the requested indices/aliases:
        boolean granted = true;
        for (Map.Entry<String, IndicesAccessControl.IndexAccessControl> entry : indexPermissions.entrySet()) {
            if (!entry.getValue().isGranted()) {
                granted = false;
                break;
            }
        }
```

```
        return new IndicesAccessControl(granted, indexPermissions);
    }

    public static class Builder {

        private final String[] names;
        private ClusterPermission cluster = ClusterPermission.SimpleClusterPermission.NONE;
        private RunAsPermission runAs = RunAsPermission.NONE;
        private List<IndicesPermission.Group> groups = new ArrayList<>();
        private FieldPermissionsCache fieldPermissionsCache = null;
        private List<Tuple<ApplicationPrivilege, Set<String>>> applicationPrivs = new ArrayList<>();

        private Builder(String[] names, FieldPermissionsCache fieldPermissionsCache) {
            this.names = names;
            this.fieldPermissionsCache = fieldPermissionsCache;
        }

        private Builder(RoleDescriptor rd, @Nullable FieldPermissionsCache fieldPermissionsCache) {
            this.names = new String[] { rd.getName() };
            this.fieldPermissionsCache = fieldPermissionsCache;
            cluster(Sets.newHashSet(rd.getClusterPrivileges()), Arrays.asList(rd.getConditionalClusterPrivileges()));
            groups.addAll(convertFromIndicesPrivileges(rd.getIndicesPrivileges(), fieldPermissionsCache));

            final RoleDescriptor.ApplicationResourcePrivileges[] applicationPrivileges = rd.getApplicationPrivileges();
            for (int i = 0; i < applicationPrivileges.length; i++) {
                applicationPrivs.add(convertApplicationPrivilege(rd.getName(), i, applicationPrivileges[i]));
            }

            String[] rdRunAs = rd.getRunAs();
            if (rdRunAs != null && rdRunAs.length > 0) {
                this.runAs(new Privilege(Sets.newHashSet(rdRunAs), rdRunAs));
            }
        }

        public Builder cluster(Set<String> privilegeNames, Iterable<ConditionalClusterPrivilege>
conditionalClusterPrivileges) {
            List<ClusterPermission> clusterPermissions = new ArrayList<>();
            if (privilegeNames.isEmpty() == false) {
                clusterPermissions.add(new
ClusterPermission.SimpleClusterPermission(ClusterPrivilege.get(privilegeNames)));
            }
            for (ConditionalClusterPrivilege ccp : conditionalClusterPrivileges) {
                clusterPermissions.add(new ClusterPermission.ConditionalClusterPermission(ccp));
            }
            if (clusterPermissions.isEmpty()) {
                this.cluster = ClusterPermission.SimpleClusterPermission.NONE;
            } else if (clusterPermissions.size() == 1) {
                this.cluster = clusterPermissions.get(0);
```

```
        } else {
            this.cluster = new ClusterPermission.CompositeClusterPermission(clusterPermissions);
        }
        return this;
    }

    /**
     * @deprecated Use {@link #cluster(Set, Iterable)}
     */
    @Deprecated
    public Builder cluster(ClusterPrivilege privilege) {
        cluster = new ClusterPermission.SimpleClusterPermission(privilege);
        return this;
    }

    public Builder runAs(Privilege privilege) {
        runAs = new RunAsPermission(privilege);
        return this;
    }

    public Builder add(IndexPrivilege privilege, String... indices) {
        groups.add(new IndicesPermission.Group(privilege, FieldPermissions.DEFAULT, null, indices));
        return this;
    }

    public Builder add(FieldPermissions fieldPermissions, Set<BytesReference> query, IndexPrivilege privilege,
String... indices) {
        groups.add(new IndicesPermission.Group(privilege, fieldPermissions, query, indices));
        return this;
    }

    public Builder addApplicationPrivilege(ApplicationPrivilege privilege, Set<String> resources) {
        applicationPrivs.add(new Tuple<>(privilege, resources));
        return this;
    }

    public Role build() {
        IndicesPermission indices = groups.isEmpty() ? IndicesPermission.NONE :
            new IndicesPermission(groups.toArray(new IndicesPermission.Group[groups.size()]));
        final ApplicationPermission applicationPermission
            = applicationPrivs.isEmpty() ? ApplicationPermission.NONE : new
ApplicationPermission(applicationPrivs);
        return new Role(names, cluster, indices, applicationPermission, runAs);
    }

    static List<IndicesPermission.Group> convertFromIndicesPrivileges(RoleDescriptor.IndicesPrivileges[]
indicesPrivileges,
                                                                      @Nullable FieldPermissionsCache fieldPermissionsCache) {
```

```
        List<IndicesPermission.Group> list = new ArrayList<>(indicesPrivileges.length);
        for (RoleDescriptor.IndicesPrivileges privilege : indicesPrivileges) {
            final FieldPermissions fieldPermissions;
            if (fieldPermissionsCache != null) {
                fieldPermissions = fieldPermissionsCache.getFieldPermissions(privilege.getGrantedFields(),
privilege.getDeniedFields());
            } else {
                fieldPermissions = new FieldPermissions(
                    new FieldPermissionsDefinition(privilege.getGrantedFields(), privilege.getDeniedFields()));
            }
            final Set<BytesReference> query = privilege.getQuery() == null ? null :
Collections.singleton(privilege.getQuery());
            list.add(new IndicesPermission.Group(IndexPrivilege.get(Sets.newHashSet(privilege.getPrivileges())),
                fieldPermissions,
                query,
                privilege.getIndices()));

        }
        return list;
    }

    static Tuple<ApplicationPrivilege, Set<String>> convertApplicationPrivilege(String role, int index,
                                                        RoleDescriptor.ApplicationResourcePrivileges arp) {
        return new Tuple<>(new ApplicationPrivilege(arp.getApplication(),
            Sets.newHashSet(arp.getPrivileges()),
            arp.getPrivileges()
        ), Sets.newHashSet(arp.getResources()));
    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.apache.lucene.util.automaton.Automaton;
import org.apache.lucene.util.automaton.TooComplexToDeterminizeException;
import org.elasticsearch.ElasticsearchSecurityException;
import org.elasticsearch.cluster.metadata.AliasOrIndex;
import org.elasticsearch.cluster.metadata.IndexMetaData;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.logging.Loggers;
import org.elasticsearch.xpack.core.security.authz.accesscontrol.IndicesAccessControl;
```

```
import org.elasticsearch.xpack.core.security.authz.privilege.IndexPrivilege;
import org.elasticsearch.xpack.core.security.support.Automatons;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.HashMap;
import java.util.HashSet;
import java.util.Iterator;
import java.util.List;
import java.util.Map;
import java.util.Objects;
import java.util.Set;
import java.util.SortedMap;
import java.util.concurrent.ConcurrentHashMap;
import java.util.function.Function;
import java.util.function.Predicate;

import static java.util.Collections.unmodifiableMap;
import static java.util.Collections.unmodifiableSet;

/**
* A permission that is based on privileges for index related actions executed
* on specific indices
*/
public final class IndicesPermission implements Iterable<IndicesPermission.Group> {

  public static final IndicesPermission NONE = new IndicesPermission();

  private final Function<String, Predicate<String>> loadingFunction;

  private final ConcurrentHashMap<String, Predicate<String>> allowedIndicesMatchersForAction = new
ConcurrentHashMap<>();

  private final Group[] groups;

  public IndicesPermission(Group... groups) {
    this.groups = groups;
    loadingFunction = (action) -> {
      List<String> indices = new ArrayList<>();
      for (Group group : groups) {
        if (group.actionMatcher.test(action)) {
          indices.addAll(Arrays.asList(group.indices));
        }
      }
      return indexMatcher(indices);
    };
  }
```

```
    static Predicate<String> indexMatcher(List<String> indices) {
        try {
            return Automatons.predicate(indices);
        } catch (TooComplexToDeterminizeException e) {
            Loggers.getLogger(IndicesPermission.class).debug("Index pattern automaton [{}] is too complex", indices);
            String description = Strings.collectionToCommaDelimitedString(indices);
            if (description.length() > 80) {
                description = Strings.cleanTruncate(description, 80) + "...";
            }
            throw new ElasticsearchSecurityException("The set of permitted index patterns [{}] is too complex to
evaluate", e, description);
        }
    }

    @Override
    public Iterator<Group> iterator() {
        return Arrays.asList(groups).iterator();
    }

    public Group[] groups() {
        return groups;
    }

    /**
     * @return A predicate that will match all the indices that this permission
     * has the privilege for executing the given action on.
     */
    public Predicate<String> allowedIndicesMatcher(String action) {
        return allowedIndicesMatchersForAction.computeIfAbsent(action, loadingFunction);
    }

    /**
     * Checks if the permission matches the provided action, without looking at indices.
     * To be used in very specific cases where indices actions need to be authorized regardless of their indices.
     * The usecase for this is composite actions that are initially only authorized based on the action name (indices are
not
     * checked on the coordinating node), and properly authorized later at the shard level checking their indices as
well.
     */
    public boolean check(String action) {
        for (Group group : groups) {
            if (group.check(action)) {
                return true;
            }
        }
        return false;
    }
```

```
  public Automaton allowedActionsMatcher(String index) {
    List<Automaton> automatonList = new ArrayList<>();
    for (Group group : groups) {
      if (group.indexNameMatcher.test(index)) {
        automatonList.add(group.privilege.getAutomaton());
      }
    }
    return automatonList.isEmpty() ? Automatons.EMPTY : Automatons.unionAndMinimize(automatonList);
  }

  /**
   * Authorizes the provided action against the provided indices, given the current cluster metadata
   */
  public Map<String, IndicesAccessControl.IndexAccessControl> authorize(String action, Set<String>
requestedIndicesOrAliases,
                                                MetaData metaData, FieldPermissionsCache fieldPermissionsCache) {
    // now... every index that is associated with the request, must be granted
    // by at least one indices permission group

    SortedMap<String, AliasOrIndex> allAliasesAndIndices = metaData.getAliasAndIndexLookup();
    Map<String, Set<FieldPermissions>> fieldPermissionsByIndex = new HashMap<>();
    Map<String, DocumentLevelPermissions> roleQueriesByIndex = new HashMap<>();
    Map<String, Boolean> grantedBuilder = new HashMap<>();

    for (String indexOrAlias : requestedIndicesOrAliases) {
      boolean granted = false;
      Set<String> concreteIndices = new HashSet<>();
      AliasOrIndex aliasOrIndex = allAliasesAndIndices.get(indexOrAlias);
      if (aliasOrIndex != null) {
        for (IndexMetaData indexMetaData : aliasOrIndex.getIndices()) {
          concreteIndices.add(indexMetaData.getIndex().getName());
        }
      }

      for (Group group : groups) {
        if (group.check(action, indexOrAlias)) {
          granted = true;
          for (String index : concreteIndices) {
            Set<FieldPermissions> fieldPermissions = fieldPermissionsByIndex.computeIfAbsent(index, (k) ->
new HashSet<>());
            fieldPermissionsByIndex.put(indexOrAlias, fieldPermissions);
            fieldPermissions.add(group.getFieldPermissions());
            DocumentLevelPermissions permissions =
                  roleQueriesByIndex.computeIfAbsent(index, (k) -> new DocumentLevelPermissions());
            roleQueriesByIndex.putIfAbsent(indexOrAlias, permissions);
            if (group.hasQuery()) {
              permissions.addAll(group.getQuery());
            } else {
```

```
                // if more than one permission matches for a concrete index here and if
                // a single permission doesn't have a role query then DLS will not be
                // applied even when other permissions do have a role query
                permissions.setAllowAll(true);
            }
        }
    }
}

if (concreteIndices.isEmpty()) {
    grantedBuilder.put(indexOrAlias, granted);
} else {
    grantedBuilder.put(indexOrAlias, granted);
    for (String concreteIndex : concreteIndices) {
        grantedBuilder.put(concreteIndex, granted);
    }
}
}

Map<String, IndicesAccessControl.IndexAccessControl> indexPermissions = new HashMap<>();
for (Map.Entry<String, Boolean> entry : grantedBuilder.entrySet()) {
    String index = entry.getKey();
    DocumentLevelPermissions permissions = roleQueriesByIndex.get(index);
    final Set<BytesReference> roleQueries;
    if (permissions != null && permissions.isAllowAll() == false) {
        roleQueries = unmodifiableSet(permissions.queries);
    } else {
        roleQueries = null;
    }

    final FieldPermissions fieldPermissions;
    final Set<FieldPermissions> indexFieldPermissions = fieldPermissionsByIndex.get(index);
    if (indexFieldPermissions != null && indexFieldPermissions.isEmpty() == false) {
        fieldPermissions = indexFieldPermissions.size() == 1 ? indexFieldPermissions.iterator().next() :
                fieldPermissionsCache.getFieldPermissions(indexFieldPermissions);
    } else {
        fieldPermissions = FieldPermissions.DEFAULT;
    }
    indexPermissions.put(index, new IndicesAccessControl.IndexAccessControl(entry.getValue(),
fieldPermissions, roleQueries));
}
return unmodifiableMap(indexPermissions);
}

public static class Group {
    private final IndexPrivilege privilege;
    private final Predicate<String> actionMatcher;
    private final String[] indices;
```

```
        private final Predicate<String> indexNameMatcher;

        public FieldPermissions getFieldPermissions() {
            return fieldPermissions;
        }

        private final FieldPermissions fieldPermissions;
        private final Set<BytesReference> query;

        public Group(IndexPrivilege privilege, FieldPermissions fieldPermissions, @Nullable Set<BytesReference>
query, String... indices) {
            assert indices.length != 0;
            this.privilege = privilege;
            this.actionMatcher = privilege.predicate();
            this.indices = indices;
            this.indexNameMatcher = indexMatcher(Arrays.asList(indices));
            this.fieldPermissions = Objects.requireNonNull(fieldPermissions);
            this.query = query;
        }

        public IndexPrivilege privilege() {
            return privilege;
        }

        public String[] indices() {
            return indices;
        }

        @Nullable
        public Set<BytesReference> getQuery() {
            return query;
        }

        private boolean check(String action) {
            return actionMatcher.test(action);
        }

        private boolean check(String action, String index) {
            assert index != null;
            return check(action) && indexNameMatcher.test(index);
        }

        boolean hasQuery() {
            return query != null;
        }
    }

    private static class DocumentLevelPermissions {
```

```
        private Set<BytesReference> queries = null;
        private boolean allowAll = false;

        private void addAll(Set<BytesReference> query) {
            if (allowAll == false) {
                if (queries == null) {
                    queries = new HashSet<>();
                }
                queries.addAll(query);
            }
        }

        private boolean isAllowAll() {
            return allowAll;
        }

        private void setAllowAll(boolean allowAll) {
            this.allowAll = allowAll;
        }
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.xpack.core.security.authz.permission;

import org.apache.logging.log4j.LogManager;
import org.apache.logging.log4j.Logger;
import org.apache.lucene.util.automaton.Automaton;
import org.apache.lucene.util.automaton.Operations;
import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.util.set.Sets;
import org.elasticsearch.xpack.core.security.authz.privilege.ApplicationPrivilege;
import org.elasticsearch.xpack.core.security.support.Automatons;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Set;
import java.util.function.Predicate;
import java.util.stream.Collectors;
```

```
/**
* A permission that is based on privileges for application (non elasticsearch) capabilities
*/
public final class ApplicationPermission {

  public static final ApplicationPermission NONE = new ApplicationPermission(Collections.emptyList());

  private final Logger logger;
  private final List<PermissionEntry> permissions;

  /**
   * @param privilegesAndResources A list of (privilege, resources). Each element in the {@link List} is a {@link
Tuple} containing
   *                              a single {@link ApplicationPrivilege} and the {@link Set} of resources to which that
privilege is
   *                              applied. The resources are treated as a wildcard {@link Automatons#pattern}.
   */
  ApplicationPermission(List<Tuple<ApplicationPrivilege, Set<String>>> privilegesAndResources) {
    this.logger = LogManager.getLogger(getClass());
    Map<ApplicationPrivilege, PermissionEntry> permissionsByPrivilege = new HashMap<>();
    privilegesAndResources.forEach(tup -> permissionsByPrivilege.compute(tup.v1(), (appPriv, existing) -> {
      final Set<String> resourceNames = tup.v2();
      final Automaton patterns = Automatons.patterns(resourceNames);
      if (existing == null) {
        return new PermissionEntry(appPriv, resourceNames, patterns);
      } else {
        return new PermissionEntry(appPriv, Sets.union(existing.resourceNames, resourceNames),
          Automatons.unionAndMinimize(Arrays.asList(existing.resourceAutomaton, patterns)));
      }
    }));
    this.permissions = Collections.unmodifiableList(new ArrayList<>(permissionsByPrivilege.values()));
  }

  /**
   * Determines whether this permission grants the specified privilege on the given resource.
   * <p>
   * An {@link ApplicationPermission} consists of a sequence of permission entries, where each entry contains a
single
   * {@link ApplicationPrivilege} and one or more resource patterns.
   * </p>
   * <p>
   * This method returns {@code true} if, one or more of those entries meet the following criteria
   * </p>
   * <ul>
   * <li>The entry's application, when interpreted as an {@link Automaton} {@link Automatons#pattern(String)
pattern} matches the
   * application given in the argument (interpreted as a raw string)
   * </li>
```

```
    * <li>The {@link ApplicationPrivilege#getAutomaton automaton that defines the entry's actions} entirely covers
the
    * automaton given in the argument (that is, the argument is a subset of the entry's automaton)
    * </li>
    * <li>The entry's resources, when interpreted as an {@link Automaton} {@link Automatons#patterns(String...)}
set of patterns} entirely
    * covers the resource given in the argument (also interpreted as an {@link Automaton} {@link
Automatons#pattern(String) pattern}.
    * </li>
    * </ul>
    */
   public boolean grants(ApplicationPrivilege other, String resource) {
       Automaton resourceAutomaton = Automatons.patterns(resource);
       final boolean matched = permissions.stream().anyMatch(e -> e.grants(other, resourceAutomaton));
       logger.trace("Permission [{}] {} grant [{} , {}]", this, matched ? "does" : "does not", other, resource);
       return matched;
   }

   @Override
   public String toString() {
       return getClass().getSimpleName() + "{privileges=" + permissions + "}";
   }

   public Set<String> getApplicationNames() {
       return permissions.stream()
           .map(e -> e.privilege.getApplication())
           .collect(Collectors.toSet());
   }

   public Set<ApplicationPrivilege> getPrivileges(String application) {
       return permissions.stream()
           .filter(e -> application.equals(e.privilege.getApplication()))
           .map(e -> e.privilege)
           .collect(Collectors.toSet());
   }

   /**
    * Returns a set of resource patterns that are permitted for the provided privilege.
    * The returned set may include patterns that overlap (e.g. "object/*" and "object/1") and may
    * also include patterns that are defined again a more permissive privilege.
    * e.g. If a permission grants
    * <ul>
    *     <li>"my-app", "read", [ "user/*" ]</li>
    *     <li>"my-app", "all", [ "user/kimchy", "config/*" ]</li>
    * </ul>
    * Then <code>getResourcePatterns( myAppRead )</code> would return <code>"user/*", "user/kimchy",
"config/*"</code>.
    */
```

```
  public Set<String> getResourcePatterns(ApplicationPrivilege privilege) {
    return permissions.stream()
      .filter(e -> e.matchesPrivilege(privilege))
      .map(e -> e.resourceNames)
      .flatMap(Set::stream)
      .collect(Collectors.toSet());
  }

  private static class PermissionEntry {
    private final ApplicationPrivilege privilege;
    private final Predicate<String> application;
    private final Set<String> resourceNames;
    private final Automaton resourceAutomaton;

    private PermissionEntry(ApplicationPrivilege privilege, Set<String> resourceNames, Automaton
resourceAutomaton) {
      this.privilege = privilege;
      this.application = Automatons.predicate(privilege.getApplication());
      this.resourceNames = resourceNames;
      this.resourceAutomaton = resourceAutomaton;
    }

    private boolean grants(ApplicationPrivilege other, Automaton resource) {
      return matchesPrivilege(other) && Operations.subsetOf(resource, this.resourceAutomaton);
    }

    private boolean matchesPrivilege(ApplicationPrivilege other) {
      if (this.privilege.equals(other)) {
        return true;
      }
      if (this.application.test(other.getApplication()) == false) {
        return false;
      }
      if (Operations.isTotal(privilege.getAutomaton())) {
        return true;
      }
      return Operations.isEmpty(privilege.getAutomaton()) == false
        && Operations.isEmpty(other.getAutomaton()) == false
        && Operations.subsetOf(other.getAutomaton(), privilege.getAutomaton());
    }

    @Override
    public String toString() {
      return privilege.toString() + ":" + resourceNames;
    }
  }
}
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.MasterNodeReadRequest;


public class GetLicenseRequest extends MasterNodeReadRequest<GetLicenseRequest> {

  public GetLicenseRequest() {
  }

  @Override
  public ActionRequestValidationException validate() {
     return null;
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

/**
* Request and Response objects for the default distribution's License
* APIs.
*/
package org.elasticsearch.protocol.xpack.license;
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.AcknowledgedRequest;

public class PutLicenseRequest extends AcknowledgedRequest<PutLicenseRequest> {

  private String licenseDefinition;
  private boolean acknowledge = false;

  public PutLicenseRequest() {
```

```
    }

    @Override
    public ActionRequestValidationException validate() {
        return null;
    }

    public void setLicenseDefinition(String licenseDefinition) {
        this.licenseDefinition = licenseDefinition;
    }

    public String getLicenseDefinition() {
        return licenseDefinition;
    }

    public void setAcknowledge(boolean acknowledge) {
        this.acknowledge = acknowledge;
    }

    public boolean isAcknowledge() {
        return acknowledge;
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import org.elasticsearch.action.support.master.AcknowledgedResponse;
import org.elasticsearch.common.ParseField;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.collect.Tuple;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.common.xcontent.ConstructingObjectParser;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentParseException;
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.protocol.xpack.common.ProtocolUtils;

import java.io.IOException;
import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.List;
```

```
import java.util.Map;
import java.util.Objects;

import static org.elasticsearch.common.xcontent.ConstructingObjectParser.constructorArg;
import static org.elasticsearch.common.xcontent.ConstructingObjectParser.optionalConstructorArg;

public class PutLicenseResponse extends AcknowledgedResponse {

  private static final ConstructingObjectParser<PutLicenseResponse, Void> PARSER = new
ConstructingObjectParser<>(
      "put_license_response", true, (a, v) -> {
      boolean acknowledged = (Boolean) a[0];
      LicensesStatus licensesStatus = LicensesStatus.fromString((String) a[1]);
      @SuppressWarnings("unchecked") Tuple<String, Map<String, String[]>> acknowledgements = (Tuple<String,
Map<String, String[]>>) a[2];
      if (acknowledgements == null) {
         return new PutLicenseResponse(acknowledged, licensesStatus);
      } else {
         return new PutLicenseResponse(acknowledged, licensesStatus, acknowledgements.v1(),
acknowledgements.v2());
      }

  });

  static {
      PARSER.declareBoolean(constructorArg(), new ParseField("acknowledged"));
      PARSER.declareString(constructorArg(), new ParseField("license_status"));
      PARSER.declareObject(optionalConstructorArg(), (parser, v) -> {
            Map<String, String[]> acknowledgeMessages = new HashMap<>();
            String message = null;
            XContentParser.Token token;
            String currentFieldName = null;
            while ((token = parser.nextToken()) != XContentParser.Token.END_OBJECT) {
               if (token == XContentParser.Token.FIELD_NAME) {
                  currentFieldName = parser.currentName();
               } else {
                  if (currentFieldName == null) {
                     throw new XContentParseException(parser.getTokenLocation(), "expected message header or
acknowledgement");
                  }
                  if ("message".equals(currentFieldName)) {
                     if (token != XContentParser.Token.VALUE_STRING) {
                        throw new XContentParseException(parser.getTokenLocation(), "unexpected message header
type");
                     }
                     message = parser.text();
                  } else {
                     if (token != XContentParser.Token.START_ARRAY) {
```

```
                    throw new XContentParseException(parser.getTokenLocation(), "unexpected acknowledgement
type");
                }
                List<String> acknowledgeMessagesList = new ArrayList<>();
                while ((token = parser.nextToken()) != XContentParser.Token.END_ARRAY) {
                    if (token != XContentParser.Token.VALUE_STRING) {
                        throw new XContentParseException(parser.getTokenLocation(), "unexpected
acknowledgement text");
                    }
                    acknowledgeMessagesList.add(parser.text());
                }
                acknowledgeMessages.put(currentFieldName, acknowledgeMessagesList.toArray(new String[0]));
            }
        }
    }
    return new Tuple<>(message, acknowledgeMessages);
},
new ParseField("acknowledge"));
}

private LicensesStatus status;
private Map<String, String[]> acknowledgeMessages;
private String acknowledgeHeader;

public PutLicenseResponse() {
}

public PutLicenseResponse(boolean acknowledged, LicensesStatus status) {
    this(acknowledged, status, null, Collections.<String, String[]>emptyMap());
}

public PutLicenseResponse(boolean acknowledged, LicensesStatus status, String acknowledgeHeader,
                          Map<String, String[]> acknowledgeMessages) {
    super(acknowledged);
    this.status = status;
    this.acknowledgeHeader = acknowledgeHeader;
    this.acknowledgeMessages = acknowledgeMessages;
}

public LicensesStatus status() {
    return status;
}

public Map<String, String[]> acknowledgeMessages() {
    return acknowledgeMessages;
}

public String acknowledgeHeader() {
```

```
    return acknowledgeHeader;
}

@Override
public void readFrom(StreamInput in) throws IOException {
    super.readFrom(in);
    status = LicensesStatus.fromId(in.readVInt());
    acknowledgeHeader = in.readOptionalString();
    int size = in.readVInt();
    Map<String, String[]> acknowledgeMessages = new HashMap<>(size);
    for (int i = 0; i < size; i++) {
        String feature = in.readString();
        int nMessages = in.readVInt();
        String[] messages = new String[nMessages];
        for (int j = 0; j < nMessages; j++) {
            messages[j] = in.readString();
        }
        acknowledgeMessages.put(feature, messages);
    }
    this.acknowledgeMessages = acknowledgeMessages;
}

@Override
public void writeTo(StreamOutput out) throws IOException {
    super.writeTo(out);
    out.writeVInt(status.id());
    out.writeOptionalString(acknowledgeHeader);
    out.writeVInt(acknowledgeMessages.size());
    for (Map.Entry<String, String[]> entry : acknowledgeMessages.entrySet()) {
        out.writeString(entry.getKey());
        out.writeVInt(entry.getValue().length);
        for (String message : entry.getValue()) {
            out.writeString(message);
        }
    }
}

@Override
protected void addCustomFields(XContentBuilder builder, Params params) throws IOException {
    builder.field("license_status", status.toString());
    if (!acknowledgeMessages.isEmpty()) {
        builder.startObject("acknowledge");
        builder.field("message", acknowledgeHeader);
        for (Map.Entry<String, String[]> entry : acknowledgeMessages.entrySet()) {
            builder.startArray(entry.getKey());
            for (String message : entry.getValue()) {
                builder.value(message);
            }
```

```
            builder.endArray();
        }
        builder.endObject();
    }
  }

  @Override
  public String toString() {
    return Strings.toString(this, true, true);
  }

  public static PutLicenseResponse fromXContent(XContentParser parser) throws IOException {
    return PARSER.parse(parser, null);
  }

  @Override
  public boolean equals(Object o) {
    if (this == o) return true;
    if (o == null || getClass() != o.getClass()) return false;
    if (!super.equals(o)) return false;
    PutLicenseResponse that = (PutLicenseResponse) o;

    return status == that.status &&
        ProtocolUtils.equals(acknowledgeMessages, that.acknowledgeMessages) &&
        Objects.equals(acknowledgeHeader, that.acknowledgeHeader);
  }

  @Override
  public int hashCode() {
    return Objects.hash(super.hashCode(), status, ProtocolUtils.hashCode(acknowledgeMessages),
acknowledgeHeader);
  }


}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import org.elasticsearch.action.ActionResponse;

public class GetLicenseResponse extends ActionResponse {

  private String license;
```

```
    GetLicenseResponse() {
    }

    public GetLicenseResponse(String license) {
        this.license = license;
    }

    public String getLicenseDefinition() {
        return license;
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import java.io.IOException;

import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.common.io.stream.Writeable;

/**
* Status of an X-Pack license.
*/
public enum LicenseStatus implements Writeable {

    ACTIVE("active"),
    INVALID("invalid"),
    EXPIRED("expired");

    private final String label;

    LicenseStatus(String label) {
        this.label = label;
    }

    public String label() {
        return label;
    }

    @Override
    public void writeTo(StreamOutput out) throws IOException {
        out.writeString(label);
    }
```

```
  public static LicenseStatus readFrom(StreamInput in) throws IOException {
    return fromString(in.readString());
  }

  public static LicenseStatus fromString(String value) {
    switch (value) {
      case "active":
        return ACTIVE;
      case "invalid":
        return INVALID;
      case "expired":
        return EXPIRED;
      default:
        throw new IllegalArgumentException("unknown license status [" + value + "]");
    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.AcknowledgedRequest;


public class DeleteLicenseRequest extends AcknowledgedRequest<DeleteLicenseRequest> {

  @Override
  public ActionRequestValidationException validate() {
    return null;
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.protocol.xpack.license;

import java.util.Locale;

public enum LicensesStatus {
  VALID((byte) 0),
  INVALID((byte) 1),
```

```
    EXPIRED((byte) 2);

    private final byte id;

    LicensesStatus(byte id) {
        this.id = id;
    }

    public int id() {
        return id;
    }

    public static LicensesStatus fromId(int id) {
        if (id == 0) {
            return VALID;
        } else if (id == 1) {
            return INVALID;
        } else if (id == 2) {
            return EXPIRED;
        } else {
            throw new IllegalStateException("no valid LicensesStatus for id=" + id);
        }
    }


    @Override
    public String toString() {
        return this.name().toLowerCase(Locale.ROOT);
    }

    public static LicensesStatus fromString(String value) {
        switch (value) {
            case "valid":
                return VALID;
            case "invalid":
                return INVALID;
            case "expired":
                return EXPIRED;
            default:
                throw new IllegalArgumentException("unknown licenses status [" + value + "]");
        }
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
package org.elasticsearch.license;

import org.elasticsearch.action.support.master.AcknowledgedRequestBuilder;
import org.elasticsearch.action.support.master.AcknowledgedResponse;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;

public class DeleteLicenseRequestBuilder extends AcknowledgedRequestBuilder<DeleteLicenseRequest,
AcknowledgedResponse,
      DeleteLicenseRequestBuilder> {

   public DeleteLicenseRequestBuilder(ElasticsearchClient client) {
      this(client, DeleteLicenseAction.INSTANCE);
   }

   /**
    * Creates new get licenses request builder
    *
    * @param client elasticsearch client
    */
   public DeleteLicenseRequestBuilder(ElasticsearchClient client, DeleteLicenseAction action) {
      super(client, action, new DeleteLicenseRequest());
   }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;


import org.apache.logging.log4j.Logger;
import org.apache.logging.log4j.message.ParameterizedMessage;
import org.apache.logging.log4j.util.Supplier;
import org.apache.lucene.util.BytesRef;
import org.elasticsearch.license.License.OperationMode;
import org.elasticsearch.watcher.FileChangesListener;
import org.elasticsearch.watcher.FileWatcher;
import org.elasticsearch.watcher.ResourceWatcherService;

import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.util.concurrent.atomic.AtomicBoolean;

/**
* File based watcher for license {@link OperationMode}
```

```
* Watches for changes in <code>licenseModePath</code>, use
* {@link #getCurrentOperationMode()} to access the latest mode
*
* In case of failure to read a valid operation mode from <code>licenseModePath</code>,
* the operation mode will default to PLATINUM
*/
public final class OperationModeFileWatcher implements FileChangesListener {
  private final ResourceWatcherService resourceWatcherService;
  private final Path licenseModePath;
  private final AtomicBoolean initialized = new AtomicBoolean();
  private final OperationMode defaultOperationMode = OperationMode.PLATINUM;
  private volatile OperationMode currentOperationMode = defaultOperationMode;
  private final Logger logger;
  private final Runnable onChange;

  public OperationModeFileWatcher(ResourceWatcherService resourceWatcherService, Path licenseModePath,
                                  Logger logger, Runnable onChange) {
    this.resourceWatcherService = resourceWatcherService;
    this.licenseModePath = licenseModePath;
    this.logger = logger;
    this.onChange = onChange;
  }

  public void init() {
    if (initialized.compareAndSet(false, true)) {
      final FileWatcher watcher = new FileWatcher(licenseModePath);
      watcher.addListener(this);
      try {
        resourceWatcherService.add(watcher, ResourceWatcherService.Frequency.HIGH);
        if (Files.exists(licenseModePath)) {
          onChange(licenseModePath);
        }
      } catch (IOException e) {
        logger.error("couldn't initialize watching license mode file", e);
      }
    }
  }

  /**
   * Returns the current operation mode based on license mode file.
   * Defaults to {@link OperationMode#PLATINUM}
   */
  public OperationMode getCurrentOperationMode() {
    return currentOperationMode;
  }

  @Override
  public void onFileInit(Path file) {
```

```
        onChange(file);
    }

    @Override
    public void onFileCreated(Path file) {
        onChange(file);
    }

    @Override
    public void onFileDeleted(Path file) {
        onChange(file);
    }

    @Override
    public void onFileChanged(Path file) {
        onChange(file);
    }

    private synchronized void onChange(Path file) {
        if (file.equals(licenseModePath)) {
            OperationMode newOperationMode = defaultOperationMode;
            try {
                if (Files.exists(licenseModePath)
                        && Files.isReadable(licenseModePath)) {
                    final byte[] content;
                    try {
                        content = Files.readAllBytes(licenseModePath);
                    } catch (IOException e) {
                        logger.error(
                                (Supplier<?>) () -> new ParameterizedMessage(
                                        "couldn't read operation mode from [{}]", licenseModePath.toAbsolutePath()), e);
                        return;
                    }
                    // this UTF-8 conversion is much pickier than java String
                    final String operationMode = new BytesRef(content).utf8ToString();
                    try {
                        newOperationMode = OperationMode.resolve(operationMode);
                    } catch (IllegalArgumentException e) {
                        logger.error(
                                (Supplier<?>) () -> new ParameterizedMessage(
                                        "invalid operation mode in [{}]", licenseModePath.toAbsolutePath()), e);
                        return;
                    }
                }
            } finally {
                // set this after the fact to prevent that we are jumping back and forth first setting to defautl and then
reading the
                // actual op mode resetting it.
```

```
            this.currentOperationMode = newOperationMode;
        }
        onChange.run();
    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequest;
import org.elasticsearch.action.ActionResponse;
import org.elasticsearch.cluster.NamedDiff;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.node.DiscoveryNodes;
import org.elasticsearch.common.ParseField;
import org.elasticsearch.common.io.stream.NamedWriteableRegistry;
import org.elasticsearch.common.settings.ClusterSettings;
import org.elasticsearch.common.settings.IndexScopedSettings;
import org.elasticsearch.common.settings.Setting;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.settings.SettingsFilter;
import org.elasticsearch.common.xcontent.NamedXContentRegistry;
import org.elasticsearch.plugins.ActionPlugin;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestHandler;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.function.Supplier;

import static org.elasticsearch.xpack.core.XPackClientActionPlugin.isTribeNode;
import static org.elasticsearch.xpack.core.XPackPlugin.transportClientMode;

public class Licensing implements ActionPlugin {

  public static final String NAME = "license";
  protected final Settings settings;
  private final boolean isTransportClient;
  private final boolean isTribeNode;

  // Until this is moved out to its own plugin (its currently in XPackPlugin.java, we need to make sure that any edits
```

```
to this file
  // are also carried out in XPackClientPlugin.java
  public List<NamedWriteableRegistry.Entry> getNamedWriteables() {
     List<NamedWriteableRegistry.Entry> entries = new ArrayList<>();
     entries.add(new NamedWriteableRegistry.Entry(MetaData.Custom.class, LicensesMetaData.TYPE,
LicensesMetaData::new));
     entries.add(new NamedWriteableRegistry.Entry(NamedDiff.class, LicensesMetaData.TYPE,
LicensesMetaData::readDiffFrom));
     return entries;
  }

  // Until this is moved out to its own plugin (its currently in XPackPlugin.java, we need to make sure that any edits
to this file
  // are also carried out in XPackClientPlugin.java
  public List<NamedXContentRegistry.Entry> getNamedXContent() {
     List<NamedXContentRegistry.Entry> entries = new ArrayList<>();
     // Metadata
     entries.add(new NamedXContentRegistry.Entry(MetaData.Custom.class, new
ParseField(LicensesMetaData.TYPE),
           LicensesMetaData::fromXContent));
     return entries;
  }

  public Licensing(Settings settings) {
     this.settings = settings;
     isTransportClient = transportClientMode(settings);
     isTribeNode = isTribeNode(settings);
  }

  @Override
  public List<ActionHandler<? extends ActionRequest, ? extends ActionResponse>> getActions() {
     if (isTribeNode) {
        return Collections.singletonList(new ActionHandler<>(GetLicenseAction.INSTANCE,
TransportGetLicenseAction.class));
     }
     return Arrays.asList(new ActionHandler<>(PutLicenseAction.INSTANCE, TransportPutLicenseAction.class),
           new ActionHandler<>(GetLicenseAction.INSTANCE, TransportGetLicenseAction.class),
           new ActionHandler<>(DeleteLicenseAction.INSTANCE, TransportDeleteLicenseAction.class),
           new ActionHandler<>(PostStartTrialAction.INSTANCE, TransportPostStartTrialAction.class),
           new ActionHandler<>(GetTrialStatusAction.INSTANCE, TransportGetTrialStatusAction.class),
           new ActionHandler<>(PostStartBasicAction.INSTANCE, TransportPostStartBasicAction.class),
           new ActionHandler<>(GetBasicStatusAction.INSTANCE, TransportGetBasicStatusAction.class));
  }

  @Override
  public List<RestHandler> getRestHandlers(Settings settings, RestController restController, ClusterSettings
clusterSettings,
        IndexScopedSettings indexScopedSettings, SettingsFilter settingsFilter, IndexNameExpressionResolver
```

```
indexNameExpressionResolver,
        Supplier<DiscoveryNodes> nodesInCluster) {
    List<RestHandler> handlers = new ArrayList<>();
    handlers.add(new RestGetLicenseAction(settings, restController));
    if (false == isTribeNode) {
        handlers.add(new RestPutLicenseAction(settings, restController));
        handlers.add(new RestDeleteLicenseAction(settings, restController));
        handlers.add(new RestGetTrialStatus(settings, restController));
        handlers.add(new RestGetBasicStatus(settings, restController));
        handlers.add(new RestPostStartTrialLicense(settings, restController));
        handlers.add(new RestPostStartBasicLicense(settings, restController));
    }
    return handlers;
  }

  // Until this is moved out to its own plugin (its currently in XPackPlugin.java, we need to make sure that any edits
to this file
  // are also carried out in XPackClientPlugin.java
  public List<Setting<?>> getSettings() {
    // TODO convert this wildcard to a real setting
    return Collections.singletonList(Setting.groupSetting("license.", Setting.Property.NodeScope));
  }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestBuilder;
import org.elasticsearch.client.ElasticsearchClient;

class PostStartBasicRequestBuilder extends ActionRequestBuilder<PostStartBasicRequest,
    PostStartBasicResponse, PostStartBasicRequestBuilder> {

  PostStartBasicRequestBuilder(ElasticsearchClient client, PostStartBasicAction action) {
    super(client, action, new PostStartBasicRequest());
  }

  public PostStartBasicRequestBuilder setAcknowledge(boolean acknowledge) {
    request.acknowledge(acknowledge);
    return this;
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
```

```
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.cluster.node.DiscoveryNodes;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.xcontent.LoggingDeprecationHandler;
import org.elasticsearch.common.xcontent.NamedXContentRegistry;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.common.xcontent.XContentType;

import java.io.IOException;
import java.nio.ByteBuffer;
import java.util.Base64;
import java.util.Collections;

import static org.elasticsearch.license.CryptUtils.encryptV3Format;
import static org.elasticsearch.license.CryptUtils.encrypt;
import static org.elasticsearch.license.CryptUtils.decryptV3Format;
import static org.elasticsearch.license.CryptUtils.decrypt;

class SelfGeneratedLicense {

  public static License create(License.Builder specBuilder, DiscoveryNodes currentNodes) {
    return create(specBuilder, LicenseUtils.compatibleLicenseVersion(currentNodes));
  }

  public static License create(License.Builder specBuilder, int version) {
    License spec = specBuilder
          .issuer("elasticsearch")
          .version(version)
          .build();
    final String signature;
    try {
      XContentBuilder contentBuilder = XContentFactory.contentBuilder(XContentType.JSON);
      spec.toXContent(contentBuilder, new
ToXContent.MapParams(Collections.singletonMap(License.LICENSE_SPEC_VIEW_MODE, "true")));
      byte[] encrypt;
      if (version < License.VERSION_CRYPTO_ALGORITHMS) {
        encrypt = encryptV3Format(BytesReference.toBytes(BytesReference.bytes(contentBuilder)));
      } else {
        encrypt = encrypt(BytesReference.toBytes(BytesReference.bytes(contentBuilder)));
      }
      byte[] bytes = new byte[4 + 4 + encrypt.length];
```

```
            ByteBuffer byteBuffer = ByteBuffer.wrap(bytes);
            // Set -version in signature
            byteBuffer.putInt(-version)
                    .putInt(encrypt.length)
                    .put(encrypt);
            signature = Base64.getEncoder().encodeToString(bytes);
        } catch (IOException e) {
            throw new IllegalStateException(e);
        }
        return License.builder().fromLicenseSpec(spec, signature).build();
    }

    public static boolean verify(final License license) {
        try {
            byte[] signatureBytes = Base64.getDecoder().decode(license.signature());
            ByteBuffer byteBuffer = ByteBuffer.wrap(signatureBytes);
            int version = byteBuffer.getInt();
            int contentLen = byteBuffer.getInt();
            byte[] content = new byte[contentLen];
            byteBuffer.get(content);
            final License expectedLicense;
            // Version in signature is -version, so check for -(-version) < 4
            byte[] decryptedContent = (-version < License.VERSION_CRYPTO_ALGORITHMS) ?
decryptV3Format(content) : decrypt(content);
            // EMPTY is safe here because we don't call namedObject
            try (XContentParser parser = XContentFactory.xContent(XContentType.JSON)
                    .createParser(NamedXContentRegistry.EMPTY, LoggingDeprecationHandler.INSTANCE,
decryptedContent)) {
                parser.nextToken();
                expectedLicense = License.builder().fromLicenseSpec(License.fromXContent(parser),
                        license.signature()).version(-version).build();
            }
            return license.equals(expectedLicense);
        } catch (IOException e) {
            throw new IllegalStateException(e);
        }
    }

    public static boolean validSelfGeneratedType(String type) {
        return "basic".equals(type) || "trial".equals(type);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.SecretKey;
import javax.crypto.SecretKeyFactory;
import javax.crypto.spec.PBEKeySpec;
import javax.crypto.spec.SecretKeySpec;
import java.nio.charset.StandardCharsets;
import java.security.GeneralSecurityException;
import java.security.KeyFactory;
import java.security.MessageDigest;
import java.security.PrivateKey;
import java.security.PublicKey;
import java.security.SecureRandom;
import java.security.NoSuchAlgorithmException;
import java.security.InvalidKeyException;
import java.security.spec.InvalidKeySpecException;
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;
import java.util.Base64;

public class CryptUtils {
  // SALT must be at least 128bits for FIPS 140-2 compliance
  private static final byte[] SALT = {
        (byte) 0x74, (byte) 0x68, (byte) 0x69, (byte) 0x73,
        (byte) 0x69, (byte) 0x73, (byte) 0x74, (byte) 0x68,
        (byte) 0x65, (byte) 0x73, (byte) 0x61, (byte) 0x6C,
        (byte) 0x74, (byte) 0x77, (byte) 0x65, (byte) 0x75
  };
  private static final String KEY_ALGORITHM = "RSA";
  private static final char[] DEFAULT_PASS_PHRASE = "elasticsearch-license".toCharArray();
  private static final String KDF_ALGORITHM = "PBKDF2WithHmacSHA512";
  private static final int KDF_ITERATION_COUNT = 10000;
  private static final String CIPHER_ALGORITHM = "AES";
  // This can be changed to 256 once Java 9 is the minimum version
  // http://www.oracle.com/technetwork/java/javase/terms/readme/jdk9-readme-3852447.html#jce
  private static final int ENCRYPTION_KEY_LENGTH = 128;
  private static final SecureRandom RANDOM = new SecureRandom();

  /**
   * Read encrypted private key file content with default pass phrase
   */
  public static PrivateKey readEncryptedPrivateKey(byte[] fileContents) {
     return readEncryptedPrivateKey(fileContents, DEFAULT_PASS_PHRASE, false);
  }
```

```
/**
 * Returns encrypted private key file content with default pass phrase
 */
public static byte[] writeEncryptedPrivateKey(PrivateKey privateKey) {
    return writeEncryptedPrivateKey(privateKey, DEFAULT_PASS_PHRASE);
}

/**
 * Read encrypted private key file content with provided <code>passPhrase</code>
 */
public static PrivateKey readEncryptedPrivateKey(byte[] fileContents, char[] passPhrase, boolean preV4) {
    byte[] keyBytes = preV4 ? decryptV3Format(fileContents) : decrypt(fileContents, passPhrase);
    PKCS8EncodedKeySpec privateKeySpec = new PKCS8EncodedKeySpec(keyBytes);
    try {
        return KeyFactory.getInstance(KEY_ALGORITHM).generatePrivate(privateKeySpec);
    } catch (NoSuchAlgorithmException | InvalidKeySpecException e) {
        throw new IllegalStateException(e);
    }
}

/**
 * Read public key file content
 */
public static PublicKey readPublicKey(byte[] fileContents) {
    X509EncodedKeySpec publicKeySpec = new X509EncodedKeySpec(fileContents);
    try {
        return KeyFactory.getInstance(CryptUtils.KEY_ALGORITHM).generatePublic(publicKeySpec);
    } catch (NoSuchAlgorithmException | InvalidKeySpecException e) {
        throw new IllegalStateException(e);
    }
}

/**
 * Returns encrypted public key file content with provided <code>passPhrase</code>
 */
public static byte[] writeEncryptedPublicKey(PublicKey publicKey) {
    X509EncodedKeySpec encodedKeySpec = new X509EncodedKeySpec(publicKey.getEncoded());
    return encrypt(encodedKeySpec.getEncoded(), DEFAULT_PASS_PHRASE);
}

/**
 * Returns encrypted private key file content with provided <code>passPhrase</code>
 */
public static byte[] writeEncryptedPrivateKey(PrivateKey privateKey, char[] passPhrase) {
    PKCS8EncodedKeySpec encodedKeySpec = new PKCS8EncodedKeySpec(privateKey.getEncoded());
    return encrypt(encodedKeySpec.getEncoded(), passPhrase);
}
```

```
/**
 * Encrypts provided <code>data</code> with <code>DEFAULT_PASS_PHRASE</code>
 */
static byte[] encrypt(byte[] data) {
    return encrypt(data, DEFAULT_PASS_PHRASE);
}

/**
 * Decrypts provided <code>encryptedData</code> with <code>DEFAULT_PASS_PHRASE</code>
 */
static byte[] decrypt(byte[] encryptedData) {
    return decrypt(encryptedData, DEFAULT_PASS_PHRASE);
}

/**
 * Encrypts provided <code>data</code> with <code>passPhrase</code>
 */
private static byte[] encrypt(byte[] data, char[] passPhrase) {
    try {
        final Cipher encryptionCipher = getEncryptionCipher(deriveSecretKey(passPhrase));
        return encryptionCipher.doFinal(data);
    } catch (IllegalBlockSizeException | BadPaddingException e) {
        throw new IllegalStateException(e);
    }
}

/**
 * Decrypts provided <code>encryptedData</code> with <code>passPhrase</code>
 */
private static byte[] decrypt(byte[] encryptedData, char[] passPhrase) {
    try {
        final Cipher cipher = getDecryptionCipher(deriveSecretKey(passPhrase));
        return cipher.doFinal(encryptedData);
    } catch (IllegalBlockSizeException | BadPaddingException e) {
        throw new IllegalStateException(e);
    }
}

static byte[] encryptV3Format(byte[] data) {
    try {
        SecretKey encryptionKey = getV3Key();
        final Cipher encryptionCipher = getEncryptionCipher(encryptionKey);
        return encryptionCipher.doFinal(pad(data, 20));
    } catch (GeneralSecurityException e) {
        throw new IllegalStateException(e);
    }
}
```

```
static byte[] decryptV3Format(byte[] data) {
    try {
        SecretKey decryptionKey = getV3Key();
        final Cipher decryptionCipher = getDecryptionCipher(decryptionKey);
        return unPad(decryptionCipher.doFinal(data));
    } catch (GeneralSecurityException e) {
        throw new IllegalStateException(e);
    }
}

private static SecretKey getV3Key() throws NoSuchAlgorithmException, InvalidKeySpecException {
    final byte[] salt = {
            (byte) 0xA9, (byte) 0xA2, (byte) 0xB5, (byte) 0xDE,
            (byte) 0x2A, (byte) 0x8A, (byte) 0x9A, (byte) 0xE6
    };
    final byte[] passBytes = "elasticsearch-license".getBytes(StandardCharsets.UTF_8);
    final byte[] digest = MessageDigest.getInstance("SHA-512").digest(passBytes);
    final char[] hashedPassphrase = Base64.getEncoder().encodeToString(digest).toCharArray();
    PBEKeySpec keySpec = new PBEKeySpec(hashedPassphrase, salt, 1024, 128);
    byte[] shortKey = SecretKeyFactory.getInstance("PBEWithSHA1AndDESede").
            generateSecret(keySpec).getEncoded();
    byte[] intermediaryKey = new byte[16];
    for (int i = 0, j = 0; i < 16; i++) {
        intermediaryKey[i] = shortKey[j];
        if (++j == shortKey.length)
            j = 0;
    }
    return new SecretKeySpec(intermediaryKey, "AES");
}

private static SecretKey deriveSecretKey(char[] passPhrase) {
    try {
        PBEKeySpec keySpec = new PBEKeySpec(passPhrase, SALT, KDF_ITERATION_COUNT,
ENCRYPTION_KEY_LENGTH);

        SecretKey secretKey = SecretKeyFactory.getInstance(KDF_ALGORITHM).
                generateSecret(keySpec);
        return new SecretKeySpec(secretKey.getEncoded(), CIPHER_ALGORITHM);
    } catch (NoSuchAlgorithmException | InvalidKeySpecException e) {
        throw new IllegalStateException(e);
    }
}

private static Cipher getEncryptionCipher(SecretKey secretKey) {
    return getCipher(Cipher.ENCRYPT_MODE, secretKey);
}
```

```
private static Cipher getDecryptionCipher(SecretKey secretKey) {
    return getCipher(Cipher.DECRYPT_MODE, secretKey);
}

private static Cipher getCipher(int mode, SecretKey secretKey) {
    try {
        Cipher cipher = Cipher.getInstance(CIPHER_ALGORITHM);
        cipher.init(mode, secretKey, RANDOM);
        return cipher;
    } catch (NoSuchAlgorithmException | InvalidKeyException | NoSuchPaddingException e) {
        throw new IllegalStateException(e);
    }
}

private static byte[] pad(byte[] bytes, int length) {
    if (bytes.length >= length) {
        byte[] out = new byte[bytes.length + 1];
        System.arraycopy(bytes, 0, out, 0, bytes.length);
        out[bytes.length] = (byte) 1;
        return out;
    }

    byte[] out = new byte[length + 1];

    int i = 0;
    for (; i < bytes.length; i++)
        out[i] = bytes[i];

    int padded = length - i;

    // fill the rest with random bytes
    byte[] fill = new byte[padded - 1];
    RANDOM.nextBytes(fill);
    System.arraycopy(fill, 0, out, i, padded - 1);

    out[length] = (byte) (padded + 1);

    return out;
}

private static byte[] unPad(byte[] bytes) {
    int padded = (int) bytes[bytes.length - 1];
    int targetLength = bytes.length - padded;

    byte[] out = new byte[targetLength];

    System.arraycopy(bytes, 0, out, 0, targetLength);
```

```
        return out;
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.rest.BytesRestResponse;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.RestResponse;
import org.elasticsearch.rest.action.RestBuilderListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;

import java.io.IOException;
import java.util.Map;

import static org.elasticsearch.rest.RestRequest.Method.POST;

public class RestPostStartTrialLicense extends XPackRestHandler {

    RestPostStartTrialLicense(Settings settings, RestController controller) {
        super(settings);
        controller.registerHandler(POST, URI_BASE + "/license/start_trial", this);
    }

    @Override
    protected RestChannelConsumer doPrepareRequest(RestRequest request, XPackClient client) throws IOException
{
        PostStartTrialRequest startTrialRequest = new PostStartTrialRequest();
        startTrialRequest.setType(request.param("type", "trial"));
        startTrialRequest.acknowledge(request.paramAsBoolean("acknowledge", false));
        return channel -> client.licensing().postStartTrial(startTrialRequest,
                new RestBuilderListener<PostStartTrialResponse>(channel) {
                    @Override
                    public RestResponse buildResponse(PostStartTrialResponse response, XContentBuilder builder) throws
Exception {
                        PostStartTrialResponse.Status status = response.getStatus();
                        builder.startObject();
                        builder.field("acknowledged", startTrialRequest.isAcknowledged());
                        if (status.isTrialStarted()) {
                            builder.field("trial_was_started", true);
```

```
                builder.field("type", startTrialRequest.getType());
            } else {
                builder.field("trial_was_started", false);
                builder.field("error_message", status.getErrorMessage());
            }

            Map<String, String[]> acknowledgementMessages = response.getAcknowledgementMessages();
            if (acknowledgementMessages.isEmpty() == false) {
                builder.startObject("acknowledge");
                builder.field("message", response.getAcknowledgementMessage());
                for (Map.Entry<String, String[]> entry : acknowledgementMessages.entrySet()) {
                    builder.startArray(entry.getKey());
                    for (String message : entry.getValue()) {
                        builder.value(message);
                    }
                    builder.endArray();
                }
                builder.endObject();
            }
            builder.endObject();
            return new BytesRestResponse(status.getRestStatus(), builder);
        }
    });
  }

  @Override
  public String getName() {
    return "xpack_upgrade_to_trial_action";
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.apache.logging.log4j.Logger;
import org.apache.logging.log4j.LogManager;
import org.apache.logging.log4j.message.ParameterizedMessage;
import org.apache.logging.log4j.util.Supplier;
import org.elasticsearch.Version;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.common.settings.Settings;
```

```
import org.elasticsearch.xpack.core.XPackPlugin;

import java.time.Clock;
import java.util.UUID;

public class StartupSelfGeneratedLicenseTask extends ClusterStateUpdateTask {
  private static final Logger logger = LogManager.getLogger(StartupSelfGeneratedLicenseTask.class);

  /**
   * Max number of nodes licensed by generated trial license
   */
  private int selfGeneratedLicenseMaxNodes = 1000;

  private final Settings settings;
  private final Clock clock;
  private final ClusterService clusterService;

  public StartupSelfGeneratedLicenseTask(Settings settings, Clock clock, ClusterService clusterService) {
     this.settings = settings;
     this.clock = clock;
     this.clusterService = clusterService;
  }

  @Override
  public void clusterStateProcessed(String source, ClusterState oldState, ClusterState newState) {
     LicensesMetaData licensesMetaData = newState.metaData().custom(LicensesMetaData.TYPE);
     if (logger.isDebugEnabled()) {
        logger.debug("registered self generated license: {}", licensesMetaData);
     }
  }

  @Override
  public ClusterState execute(ClusterState currentState) throws Exception {
     XPackPlugin.checkReadyForXPackCustomMetadata(currentState);
     final MetaData metaData = currentState.metaData();
     final LicensesMetaData currentLicensesMetaData = metaData.custom(LicensesMetaData.TYPE);
     // do not generate a license if any license is present
     if (currentLicensesMetaData == null) {
        String type = LicenseService.SELF_GENERATED_LICENSE_TYPE.get(settings);
        if (SelfGeneratedLicense.validSelfGeneratedType(type) == false) {
           throw new IllegalArgumentException("Illegal self generated license type [" + type +
                 "]. Must be trial or basic.");
        }
        return updateWithLicense(currentState, type);
     } else if (LicenseUtils.signatureNeedsUpdate(currentLicensesMetaData.getLicense(), currentState.nodes())) {
        return updateLicenseSignature(currentState, currentLicensesMetaData);
     } else if (LicenseUtils.licenseNeedsExtended(currentLicensesMetaData.getLicense())) {
        return extendBasic(currentState, currentLicensesMetaData);
```

```
        } else {
            return currentState;
        }
    }

    private ClusterState updateLicenseSignature(ClusterState currentState, LicensesMetaData
currentLicenseMetaData) {
        License license = currentLicenseMetaData.getLicense();
        MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
        String type = license.type();
        long issueDate = license.issueDate();
        long expiryDate = license.expiryDate();
        // extend the basic license expiration date if needed since extendBasic will not be called now
        if ("basic".equals(type) &&  expiryDate !=
LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS) {
            expiryDate = LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS;
        }
        License.Builder specBuilder = License.builder()
                .uid(license.uid())
                .issuedTo(license.issuedTo())
                .maxNodes(selfGeneratedLicenseMaxNodes)
                .issueDate(issueDate)
                .type(type)
                .expiryDate(expiryDate);
        License selfGeneratedLicense = SelfGeneratedLicense.create(specBuilder, currentState.nodes());
        Version trialVersion = currentLicenseMetaData.getMostRecentTrialVersion();
        LicensesMetaData newLicenseMetadata = new LicensesMetaData(selfGeneratedLicense, trialVersion);
        mdBuilder.putCustom(LicensesMetaData.TYPE, newLicenseMetadata);
        logger.info("Updating existing license to the new version.\n\nOld license:\n {}\n\n New license:\n{}",
            license, newLicenseMetadata.getLicense());
        return ClusterState.builder(currentState).metaData(mdBuilder).build();
    }

    @Override
    public void onFailure(String source, @Nullable Exception e) {
        logger.error((Supplier<?>) () -> new ParameterizedMessage("unexpected failure during [{}]", source), e);
    }

    private ClusterState extendBasic(ClusterState currentState, LicensesMetaData currentLicenseMetadata) {
        License license = currentLicenseMetadata.getLicense();
        MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
        LicensesMetaData newLicenseMetadata = createBasicLicenseFromExistingLicense(currentLicenseMetadata);
        mdBuilder.putCustom(LicensesMetaData.TYPE, newLicenseMetadata);
        logger.info("Existing basic license has an expiration. Basic licenses no longer expire." +
                "Regenerating license.\n\nOld license:\n {}\n\n New license:\n{}", license,
newLicenseMetadata.getLicense());
        return ClusterState.builder(currentState).metaData(mdBuilder).build();
    }
```

```
  private LicensesMetaData createBasicLicenseFromExistingLicense(LicensesMetaData currentLicenseMetadata) {
    License currentLicense = currentLicenseMetadata.getLicense();
    License.Builder specBuilder = License.builder()
            .uid(currentLicense.uid())
            .issuedTo(currentLicense.issuedTo())
            .maxNodes(selfGeneratedLicenseMaxNodes)
            .issueDate(currentLicense.issueDate())
            .type("basic")
            .expiryDate(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS);
    License selfGeneratedLicense = SelfGeneratedLicense.create(specBuilder, currentLicense.version());
    Version trialVersion = currentLicenseMetadata.getMostRecentTrialVersion();
    return new LicensesMetaData(selfGeneratedLicense, trialVersion);
  }

  private ClusterState updateWithLicense(ClusterState currentState, String type) {
    long issueDate = clock.millis();
    MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
    long expiryDate;
    if ("basic".equals(type)) {
      expiryDate = LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS;
    } else {
      expiryDate = issueDate +
LicenseService.NON_BASIC_SELF_GENERATED_LICENSE_DURATION.getMillis();
    }
    License.Builder specBuilder = License.builder()
            .uid(UUID.randomUUID().toString())
            .issuedTo(clusterService.getClusterName().value())
            .maxNodes(selfGeneratedLicenseMaxNodes)
            .issueDate(issueDate)
            .type(type)
            .expiryDate(expiryDate);
    License selfGeneratedLicense = SelfGeneratedLicense.create(specBuilder, currentState.nodes());
    LicensesMetaData licensesMetaData;
    if ("trial".equals(type)) {
      licensesMetaData = new LicensesMetaData(selfGeneratedLicense, Version.CURRENT);
    } else {
      licensesMetaData = new LicensesMetaData(selfGeneratedLicense, null);
    }
    mdBuilder.putCustom(LicensesMetaData.TYPE, licensesMetaData);
    return ClusterState.builder(currentState).metaData(mdBuilder).build();
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.cluster.AbstractNamedDiffable;
import org.elasticsearch.cluster.MergableCustomMetaData;
import org.elasticsearch.cluster.NamedDiff;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.license.License.OperationMode;
import org.elasticsearch.xpack.core.XPackPlugin;

import java.io.IOException;
import java.util.EnumSet;

/**
* Contains metadata about registered licenses
*/
public class LicensesMetaData extends AbstractNamedDiffable<MetaData.Custom> implements
XPackPlugin.XPackMetaDataCustom,
      MergableCustomMetaData<LicensesMetaData> {

   public static final String TYPE = "licenses";

   /**
    * When license is explicitly removed by a user, LICENSE_TOMBSTONE
    * is used as a placeholder in the license metadata. This enables
    * us to distinguish between the scenario when a cluster never
    * had a license (null) and when a license was removed explicitly
    * (LICENSE_TOMBSTONE).
    * We rely on this to decide whether to generate a unsigned trial
    * license or not. we should only generate a license if no license
    * ever existed in the cluster state
    */
   public static final License LICENSE_TOMBSTONE = License.builder()
         .type("trial")
         .issuer("elasticsearch")
         .uid("TOMBSTONE")
         .issuedTo("")
         .maxNodes(0)
         .issueDate(0)
         .expiryDate(0)
         .build();

   private License license;
```

```
// This field describes the version of x-pack for which this cluster has exercised a trial. If the field
// is null, then no trial has been exercised. We keep the version to leave open the possibility that we
// may eventually allow a cluster to exercise a trial every time they upgrade to a new major version.
@Nullable
private Version trialVersion;

LicensesMetaData(License license, Version trialVersion) {
    this.license = license;
    this.trialVersion = trialVersion;
}

public License getLicense() {
    return license;
}

boolean isEligibleForTrial() {
    if (trialVersion == null) {
        return true;
    }
    return Version.CURRENT.major > trialVersion.major;
}

Version getMostRecentTrialVersion() {
    return trialVersion;
}

@Override
public String toString() {
    return "LicensesMetaData{" +
            "license=" + license +
            ", trialVersion=" + trialVersion +
            '}';
}

@Override
public boolean equals(Object o) {
    if (this == o) return true;
    if (o == null || getClass() != o.getClass()) return false;

    LicensesMetaData that = (LicensesMetaData) o;

    if (license != null ? !license.equals(that.license) : that.license != null) return false;
    return trialVersion != null ? trialVersion.equals(that.trialVersion) : that.trialVersion == null;
}

@Override
public int hashCode() {
```

```
    int result = license != null ? license.hashCode() : 0;
    result = 31 * result + (trialVersion != null ? trialVersion.hashCode() : 0);
    return result;
}

@Override
public String getWriteableName() {
    return TYPE;
}

@Override
public Version getMinimalSupportedVersion() {
    return Version.CURRENT.minimumCompatibilityVersion();
}

@Override
public EnumSet<MetaData.XContentContext> context() {
    return EnumSet.of(MetaData.XContentContext.GATEWAY);
}

public static LicensesMetaData fromXContent(XContentParser parser) throws IOException {
    License license = LICENSE_TOMBSTONE;
    Version trialLicense = null;
    XContentParser.Token token;
    while ((token = parser.nextToken()) != XContentParser.Token.END_OBJECT) {
        if (token == XContentParser.Token.FIELD_NAME) {
            String fieldName = parser.currentName();
            if (fieldName != null) {
                if (fieldName.equals(Fields.LICENSE)) {
                    token = parser.nextToken();
                    if (token == XContentParser.Token.START_OBJECT) {
                        license = License.fromXContent(parser);
                    } else if (token == XContentParser.Token.VALUE_NULL) {
                        license = LICENSE_TOMBSTONE;
                    }
                } else if (fieldName.equals(Fields.TRIAL_LICENSE)) {
                    parser.nextToken();
                    trialLicense = Version.fromString(parser.text());
                }
            }
        }
    }
    return new LicensesMetaData(license, trialLicense);
}

@Override
public XContentBuilder toXContent(XContentBuilder builder, Params params) throws IOException {
    if (license == LICENSE_TOMBSTONE) {
```

```
        builder.nullField(Fields.LICENSE);
    } else {
        builder.startObject(Fields.LICENSE);
        license.toInnerXContent(builder, params);
        builder.endObject();
    }
    if (trialVersion != null) {
        builder.field(Fields.TRIAL_LICENSE, trialVersion.toString());
    }
    return builder;
}

@Override
public void writeTo(StreamOutput streamOutput) throws IOException {
    if (license == LICENSE_TOMBSTONE) {
        streamOutput.writeBoolean(false); // no license
    } else {
        streamOutput.writeBoolean(true); // has a license
        license.writeTo(streamOutput);
    }
    if (streamOutput.getVersion().onOrAfter(Version.V_6_1_0)) {
        if (trialVersion == null) {
            streamOutput.writeBoolean(false);
        } else {
            streamOutput.writeBoolean(true);
            Version.writeVersion(trialVersion, streamOutput);
        }
    }
}

public LicensesMetaData(StreamInput streamInput) throws IOException {
    if (streamInput.readBoolean()) {
        license = License.readLicense(streamInput);
    } else {
        license = LICENSE_TOMBSTONE;
    }
    if (streamInput.getVersion().onOrAfter(Version.V_6_1_0)) {
        boolean hasExercisedTrial = streamInput.readBoolean();
        if (hasExercisedTrial) {
            this.trialVersion = Version.readVersion(streamInput);
        }
    }
}

public static NamedDiff<MetaData.Custom> readDiffFrom(StreamInput streamInput) throws IOException {
    return readDiffFrom(MetaData.Custom.class, TYPE, streamInput);
}
```

```
    public static License extractLicense(LicensesMetaData licensesMetaData) {
        if (licensesMetaData != null) {
            License license = licensesMetaData.getLicense();
            if (license == LicensesMetaData.LICENSE_TOMBSTONE) {
                return null;
            } else {
                return license;
            }
        }
        return null;
    }

    @Override
    public LicensesMetaData merge(LicensesMetaData other) {
        if (other.license == null) {
            return this;
        } else if (license == null
                || OperationMode.compare(other.license.operationMode(), license.operationMode()) > 0) {
            return other;
        }
        return this;
    }

    private static final class Fields {
        private static final String LICENSE = "license";
        private static final String TRIAL_LICENSE = "trial_license";
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.protocol.xpack.license.GetLicenseRequest;
import org.elasticsearch.rest.BytesRestResponse;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.RestResponse;
import org.elasticsearch.rest.action.RestBuilderListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;
```

```
import java.io.IOException;
import java.util.HashMap;
import java.util.Map;

import static org.elasticsearch.rest.RestRequest.Method.GET;
import static org.elasticsearch.rest.RestStatus.NOT_FOUND;
import static org.elasticsearch.rest.RestStatus.OK;

public class RestGetLicenseAction extends XPackRestHandler {

  @Inject
  public RestGetLicenseAction(Settings settings, RestController controller) {
    super(settings);
    // @deprecated Remove deprecations in 6.0
    controller.registerWithDeprecatedHandler(GET,  URI_BASE + "/license", this,
                                  GET, "/_license", deprecationLogger);

    // Remove _licenses support entirely in 6.0
    controller.registerAsDeprecatedHandler(GET, "/_licenses", this,
                                  "[GET /_licenses] is deprecated! Use " +
                                  "[GET /_xpack/license] instead.",
                                  deprecationLogger);
  }

  @Override
  public String getName() {
    return "xpack_get_license_action";
  }

  /**
   * There will be only one license displayed per feature, the selected license will have the latest expiry_date
   * out of all other licenses for the feature.
   * <p>
   * The licenses are sorted by latest issue_date
   */
  @Override
  public RestChannelConsumer doPrepareRequest(final RestRequest request, final XPackClient client) throws
IOException {
    final Map<String, String> overrideParams = new HashMap<>(2);
    overrideParams.put(License.REST_VIEW_MODE, "true");
    overrideParams.put(License.LICENSE_VERSION_MODE, String.valueOf(License.VERSION_CURRENT));
    final ToXContent.Params params = new ToXContent.DelegatingMapParams(overrideParams, request);
    GetLicenseRequest getLicenseRequest = new GetLicenseRequest();
    getLicenseRequest.local(request.paramAsBoolean("local", getLicenseRequest.local()));
    return channel -> client.es().admin().cluster().execute(GetLicenseAction.INSTANCE, getLicenseRequest,
        new RestBuilderListener<GetLicenseResponse>(channel) {
          @Override
          public RestResponse buildResponse(GetLicenseResponse response, XContentBuilder builder) throws
```

```
Exception {
                // Default to pretty printing, but allow ?pretty=false to disable
                if (!request.hasParam("pretty")) {
                    builder.prettyPrint().lfAtEnd();
                }
                boolean hasLicense = response.license() != null;
                builder.startObject();
                if (hasLicense) {
                    builder.startObject("license");
                    response.license().toInnerXContent(builder, params);
                    builder.endObject();
                }
                builder.endObject();
                return new BytesRestResponse(hasLicense ? OK : NOT_FOUND, builder);
            }
        });
    }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

/**
* Licensing for xpack.
*
* A {@link org.elasticsearch.license.License} is a signed set of json properties that determine what features
* are available in a running cluster. Licenses are registered through a
* {@link org.elasticsearch.license.PutLicenseRequest}. This action is handled by the master node, which places
* the signed license into the cluster state. Each node listens for cluster state updates via the
* {@link org.elasticsearch.license.LicenseService}, and updates its local copy of the license when it detects
* changes in the cluster state.
*
* The logic for which features are available given the current license is handled by
* {@link org.elasticsearch.license.XPackLicenseState}, which is updated by the
* {@link org.elasticsearch.license.LicenseService} when the license changes.
*/
package org.elasticsearch.license;
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.support.master.AcknowledgedResponse;
```

```
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.rest.RestStatus;

import java.io.IOException;
import java.util.Collections;
import java.util.HashMap;
import java.util.Map;

import static org.elasticsearch.license.PostStartBasicResponse.Status.NEED_ACKNOWLEDGEMENT;

class PostStartBasicResponse extends AcknowledgedResponse {

  private Map<String, String[]> acknowledgeMessages;
  private String acknowledgeMessage;

  enum Status {
    GENERATED_BASIC(true, null, RestStatus.OK),
    ALREADY_USING_BASIC(false, "Operation failed: Current license is basic.", RestStatus.FORBIDDEN),
    NEED_ACKNOWLEDGEMENT(false, "Operation failed: Needs acknowledgement.", RestStatus.OK);

    private final boolean isBasicStarted;
    private final String errorMessage;
    private final RestStatus restStatus;

    Status(boolean isBasicStarted, String errorMessage, RestStatus restStatus) {
      this.isBasicStarted = isBasicStarted;
      this.errorMessage = errorMessage;
      this.restStatus = restStatus;
    }

    boolean isBasicStarted() {
      return isBasicStarted;
    }

    String getErrorMessage() {
      return errorMessage;
    }

    RestStatus getRestStatus() {
      return restStatus;
    }
  }

  private Status status;

  PostStartBasicResponse() {
```

```
    }

    PostStartBasicResponse(Status status) {
        this(status, Collections.emptyMap(), null);
    }

    PostStartBasicResponse(Status status, Map<String, String[]> acknowledgeMessages, String
acknowledgeMessage) {
        super(status != NEED_ACKNOWLEDGEMENT);
        this.status = status;
        this.acknowledgeMessages = acknowledgeMessages;
        this.acknowledgeMessage = acknowledgeMessage;
    }

    public Status getStatus() {
        return status;
    }

    @Override
    public void readFrom(StreamInput in) throws IOException {
        super.readFrom(in);
        status = in.readEnum(Status.class);
        acknowledgeMessage = in.readOptionalString();
        int size = in.readVInt();
        Map<String, String[]> acknowledgeMessages = new HashMap<>(size);
        for (int i = 0; i < size; i++) {
            String feature = in.readString();
            int nMessages = in.readVInt();
            String[] messages = new String[nMessages];
            for (int j = 0; j < nMessages; j++) {
                messages[j] = in.readString();
            }
            acknowledgeMessages.put(feature, messages);
        }
        this.acknowledgeMessages = acknowledgeMessages;
    }

    @Override
    public void writeTo(StreamOutput out) throws IOException {
        super.writeTo(out);
        out.writeEnum(status);
        out.writeOptionalString(acknowledgeMessage);
        out.writeVInt(acknowledgeMessages.size());
        for (Map.Entry<String, String[]> entry : acknowledgeMessages.entrySet()) {
            out.writeString(entry.getKey());
            out.writeVInt(entry.getValue().length);
            for (String message : entry.getValue()) {
                out.writeString(message);
```

```
        }
    }
  }

  @Override
  protected void addCustomFields(XContentBuilder builder, Params params) throws IOException {
    if (status.isBasicStarted()) {
      builder.field("basic_was_started", true);
    } else {
      builder.field("basic_was_started", false);
      builder.field("error_message", status.getErrorMessage());
    }
    if (acknowledgeMessages.isEmpty() == false) {
      builder.startObject("acknowledge");
      builder.field("message", acknowledgeMessage);
      for (Map.Entry<String, String[]> entry : acknowledgeMessages.entrySet()) {
        builder.startArray(entry.getKey());
        for (String message : entry.getValue()) {
          builder.value(message);
        }
        builder.endArray();
      }
      builder.endObject();
    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
import org.elasticsearch.action.support.master.TransportMasterNodeAction;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.transport.TransportService;

public class TransportPostStartBasicAction extends TransportMasterNodeAction<PostStartBasicRequest,
PostStartBasicResponse> {
```

```
    private final LicenseService licenseService;

    @Inject
    public TransportPostStartBasicAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                         LicenseService licenseService, ThreadPool threadPool, ActionFilters actionFilters,
                                         IndexNameExpressionResolver indexNameExpressionResolver) {
        super(settings, PostStartBasicAction.NAME, transportService, clusterService, threadPool, actionFilters,
                indexNameExpressionResolver, PostStartBasicRequest::new);
        this.licenseService = licenseService;
    }

    @Override
    protected String executor() {
        return ThreadPool.Names.SAME;
    }

    @Override
    protected PostStartBasicResponse newResponse() {
        return new PostStartBasicResponse();
    }

    @Override
    protected void masterOperation(PostStartBasicRequest request, ClusterState state,
                                   ActionListener<PostStartBasicResponse> listener) throws Exception {
        licenseService.startBasicLicense(request, listener);
    }

    @Override
    protected ClusterBlockException checkBlock(PostStartBasicRequest request, ClusterState state) {
        return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_WRITE);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.protocol.xpack.license.GetLicenseRequest;

public class GetLicenseAction extends Action<GetLicenseRequest, GetLicenseResponse,
GetLicenseRequestBuilder> {
```

```
    public static final GetLicenseAction INSTANCE = new GetLicenseAction();
    public static final String NAME = "cluster:monitor/xpack/license/get";

    private GetLicenseAction() {
        super(NAME);
    }

    @Override
    public GetLicenseResponse newResponse() {
        return new GetLicenseResponse();
    }

    @Override
    public GetLicenseRequestBuilder newRequestBuilder(ElasticsearchClient client) {
        return new GetLicenseRequestBuilder(client, this);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.action.support.master.AcknowledgedResponse;

public class DeleteLicenseAction extends Action<DeleteLicenseRequest, AcknowledgedResponse,
DeleteLicenseRequestBuilder> {

    public static final DeleteLicenseAction INSTANCE = new DeleteLicenseAction();
    public static final String NAME = "cluster:admin/xpack/license/delete";

    private DeleteLicenseAction() {
        super(NAME);
    }

    @Override
    public AcknowledgedResponse newResponse() {
        return new AcknowledgedResponse();
    }

    @Override
    public DeleteLicenseRequestBuilder newRequestBuilder(ElasticsearchClient client) {
        return new DeleteLicenseRequestBuilder(client, this);
    }
```

```
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.MasterNodeRequest;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;

import java.io.IOException;

public class PostStartTrialRequest extends MasterNodeRequest<PostStartTrialRequest> {

  private boolean acknowledge = false;
  private String type;

  @Override
  public ActionRequestValidationException validate() {
     return null;
  }

  public PostStartTrialRequest setType(String type) {
     this.type = type;
     return this;
  }

  public String getType() {
     return type;
  }

  public PostStartTrialRequest acknowledge(boolean acknowledge) {
     this.acknowledge = acknowledge;
     return this;
  }

  public boolean isAcknowledged() {
     return acknowledge;
  }

  @Override
  public void readFrom(StreamInput in) throws IOException {
     super.readFrom(in);
     if (in.getVersion().onOrAfter(Version.V_6_3_0)) {
```

```
            type = in.readString();
            acknowledge = in.readBoolean();
        } else {
            type = "trial";
            acknowledge = true;
        }
    }

    @Override
    public void writeTo(StreamOutput out) throws IOException {
        Version version = Version.V_6_3_0;
        if (out.getVersion().onOrAfter(version)) {
            super.writeTo(out);
            out.writeString(type);
            out.writeBoolean(acknowledge);
        } else {
            if ("trial".equals(type) == false) {
                throw new IllegalArgumentException("All nodes in cluster must be version [" + version
                        + "] or newer to start trial with a different type than 'trial'. Attempting to write to " +
                        "a node with version [" + out.getVersion() + "] with trial type [" + type + "].");
            } else if (acknowledge == false) {
                throw new IllegalArgumentException("Request must be acknowledged to send to a node with a version " +
                        "prior to [" + version + "]. Attempting to send request to node with version [" + out.getVersion() + "] "
+
                        "without acknowledgement.");
            } else {
                super.writeTo(out);
            }
        }
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import java.io.IOException;
import java.io.InputStream;
import java.nio.ByteBuffer;
import java.util.ArrayList;
import java.util.Base64;
import java.util.Comparator;
import java.util.List;
import java.util.Locale;

import org.apache.lucene.util.CollectionUtil;
```

```
import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.ElasticsearchParseException;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.common.xcontent.LoggingDeprecationHandler;
import org.elasticsearch.common.xcontent.NamedXContentRegistry;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.ToXContentObject;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentParser;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.protocol.xpack.license.LicenseStatus;

/**
* Data structure for license. Use {@link Builder} to build a license.
* Provides serialization/deserialization & validation methods for license object
*/
public class License implements ToXContentObject {
   public static final int VERSION_START = 1;
   public static final int VERSION_NO_FEATURE_TYPE = 2;
   public static final int VERSION_START_DATE = 3;
   public static final int VERSION_CRYPTO_ALGORITHMS = 4;
   public static final int VERSION_CURRENT = VERSION_CRYPTO_ALGORITHMS;

   /**
    * XContent param name to deserialize license(s) with
    * an additional <code>status</code> field, indicating whether a
    * particular license is 'active' or 'expired' and no signature
    * and in a human readable format
    */
   public static final String REST_VIEW_MODE = "rest_view";
   /**
    * XContent param name to deserialize license(s) with
    * no signature
    */
   public static final String LICENSE_SPEC_VIEW_MODE = "license_spec_view";
   /**
    * XContent param name to deserialize licenses according
    * to a specific license version
    */
   public static final String LICENSE_VERSION_MODE = "license_version";

   public static final Comparator<License> LATEST_ISSUE_DATE_FIRST =
Comparator.comparing(License::issueDate).reversed();
```

```
    private final int version;
    private final String uid;
    private final String issuer;
    private final String issuedTo;
    private final long issueDate;
    private final String type;
    private final String subscriptionType;
    private final String feature;
    private final String signature;
    private final long expiryDate;
    private final long startDate;
    private final int maxNodes;
    private final OperationMode operationMode;

    /**
     * Decouples operation mode of a license from the license type value.
     * <p>
     * Note: The mode indicates features that should be made available, but it does not indicate whether the license is
active!
     *
     * The id byte is used for ordering operation modes (used for merging license md in tribe node)
     */
    public enum OperationMode {
        MISSING((byte) 0),
        TRIAL((byte) 1),
        BASIC((byte) 2),
        STANDARD((byte) 3),
        GOLD((byte) 4),
        PLATINUM((byte) 5);

        private final byte id;

        OperationMode(byte id) {
            this.id = id;
        }

        /** Returns non-zero positive number when <code>opMode1</code> is greater than <code>opMode2</code>
*/
        public static int compare(OperationMode opMode1, OperationMode opMode2) {
            return Integer.compare(opMode1.id, opMode2.id);
        }

        public static OperationMode resolve(String type) {
            switch (type.toLowerCase(Locale.ROOT)) {
                case "missing":
                    return MISSING;
                case "trial":
                case "none": // bwc for 1.x subscription_type field
```

```
            case "dev": // bwc for 1.x subscription_type field
            case "development": // bwc for 1.x subscription_type field
                return TRIAL;
            case "basic":
                return BASIC;
            case "standard":
                return STANDARD;
            case "silver":
            case "gold":
                return GOLD;
            case "platinum":
            case "cloud_internal":
            case "internal": // bwc for 1.x subscription_type field
                return PLATINUM;
            default:
                throw new IllegalArgumentException("unknown type [" + type + "]");
        }
    }
}

private License(int version, String uid, String issuer, String issuedTo, long issueDate, String type,
                String subscriptionType, String feature, String signature, long expiryDate, int maxNodes, long startDate)
{
    this.version = version;
    this.uid = uid;
    this.issuer = issuer;
    this.issuedTo = issuedTo;
    this.issueDate = issueDate;
    this.type = type;
    this.subscriptionType = subscriptionType;
    this.feature = feature;
    this.signature = signature;
    // We will validate that only a basic license can have the
BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS
    // in the validate() method.
    if (expiryDate == -1) {
        this.expiryDate = LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS;
    } else {
        this.expiryDate = expiryDate;
    }
    this.maxNodes = maxNodes;
    this.startDate = startDate;
    if (version == VERSION_START) {
        // in 1.x: the acceptable values for 'subscription_type': none | dev | silver | gold | platinum
        this.operationMode = OperationMode.resolve(subscriptionType);
    } else {
        // in 2.x: the acceptable values for 'type': trial | basic | silver | dev | gold | platinum
        this.operationMode = OperationMode.resolve(type);
```

```
    }
    validate();
}

/**
 * @return version of the license
 */
public int version() {
    return version;
}

/**
 * @return a unique identifier for a license
 */
public String uid() {
    return uid;
}

/**
 * @return type of the license [trial, subscription, internal]
 */
public String type() {
    return type;
}

/**
 * @return the issueDate in milliseconds
 */
public long issueDate() {
    return issueDate;
}

/**
 * @return the startDate in milliseconds
 */
public long startDate() {
    return startDate;
}

/**
 * @return the expiry date in milliseconds
 */
public long expiryDate() {
    return expiryDate;
}

/**
 * @return the maximum number of nodes this license has been issued for
```

```
 */
public int maxNodes() {
    return maxNodes;
}

/**
 * @return a string representing the entity this licenses has been issued to
 */
public String issuedTo() {
    return issuedTo;
}

/**
 * @return a string representing the entity responsible for issuing this license (internal)
 */
public String issuer() {
    return issuer;
}

/**
 * @return a string representing the signature of the license used for license verification
 */
public String signature() {
    return signature;
}

/**
 * @return the operation mode of the license as computed from the license type or from
 * the license mode file
 */
public OperationMode operationMode() {
    synchronized (this) {
        if (canReadOperationModeFromFile() && operationModeFileWatcher != null) {
            return operationModeFileWatcher.getCurrentOperationMode();
        }
    }
    return operationMode;
}

private boolean canReadOperationModeFromFile() {
    return type.equals("cloud_internal");
}

private volatile OperationModeFileWatcher operationModeFileWatcher;

/**
 * Sets the operation mode file watcher for the license and initializes the
 * file watcher when the license type allows to override operation mode from file
```

```
 */
public synchronized void setOperationModeFileWatcher(final OperationModeFileWatcher
operationModeFileWatcher) {
    this.operationModeFileWatcher = operationModeFileWatcher;
    if (canReadOperationModeFromFile()) {
        this.operationModeFileWatcher.init();
    }
}

/**
 * Removes operation mode file watcher, so unused license objects can be gc'ed
 */
public synchronized void removeOperationModeFileWatcher() {
    this.operationModeFileWatcher = null;
}

/**
 * @return the current license's status
 */
public LicenseStatus status() {
    long now = System.currentTimeMillis();
    if (issueDate > now) {
        return LicenseStatus.INVALID;
    } else if (expiryDate < now) {
        return LicenseStatus.EXPIRED;
    }
    return LicenseStatus.ACTIVE;
}

private void validate() {
    if (issuer == null) {
        throw new IllegalStateException("issuer can not be null");
    } else if (issuedTo == null) {
        throw new IllegalStateException("issuedTo can not be null");
    } else if (issueDate == -1) {
        throw new IllegalStateException("issueDate has to be set");
    } else if (type == null) {
        throw new IllegalStateException("type can not be null");
    } else if (subscriptionType == null && version == VERSION_START) {
        throw new IllegalStateException("subscriptionType can not be null");
    } else if (uid == null) {
        throw new IllegalStateException("uid can not be null");
    } else if (feature == null && version == VERSION_START) {
        throw new IllegalStateException("feature can not be null");
    } else if (maxNodes == -1) {
        throw new IllegalStateException("maxNodes has to be set");
    } else if (expiryDate == -1) {
        throw new IllegalStateException("expiryDate has to be set");
```

```
        } else if (expiryDate == LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS
&& "basic".equals(type) == false) {
            throw new IllegalStateException("only basic licenses are allowed to have no expiration");
        }
    }

    public static License readLicense(StreamInput in) throws IOException {
        int version = in.readVInt(); // Version for future extensibility
        if (version > VERSION_CURRENT) {
            throw new ElasticsearchException("Unknown license version found, please upgrade all nodes to the latest
elasticsearch-license" +
                    " plugin");
        }
        Builder builder = builder();
        builder.version(version);
        builder.uid(in.readString());
        builder.type(in.readString());
        if (version == VERSION_START) {
            builder.subscriptionType(in.readString());
        }
        builder.issueDate(in.readLong());
        if (version == VERSION_START) {
            builder.feature(in.readString());
        }
        builder.expiryDate(in.readLong());
        builder.maxNodes(in.readInt());
        builder.issuedTo(in.readString());
        builder.issuer(in.readString());
        builder.signature(in.readOptionalString());
        if (version >= VERSION_START_DATE) {
            builder.startDate(in.readLong());
        }
        return builder.build();
    }

    public void writeTo(StreamOutput out) throws IOException {
        out.writeVInt(version);
        out.writeString(uid);
        out.writeString(type);
        if (version == VERSION_START) {
            out.writeString(subscriptionType);
        }
        out.writeLong(issueDate);
        if (version == VERSION_START) {
            out.writeString(feature);
        }
        out.writeLong(expiryDate);
        out.writeInt(maxNodes);
```

```
        out.writeString(issuedTo);
        out.writeString(issuer);
        out.writeOptionalString(signature);
        if (version >= VERSION_START_DATE) {
            out.writeLong(startDate);
        }
    }

    @Override
    public String toString() {
        try {
            final XContentBuilder builder = XContentFactory.jsonBuilder();
            toXContent(builder, ToXContent.EMPTY_PARAMS);
            return Strings.toString(builder);
        } catch (IOException e) {
            return "";
        }
    }

    @Override
    public XContentBuilder toXContent(XContentBuilder builder, Params params) throws IOException {
        builder.startObject();
        toInnerXContent(builder, params);
        builder.endObject();
        return builder;
    }

    public XContentBuilder toInnerXContent(XContentBuilder builder, Params params) throws IOException {
        boolean licenseSpecMode = params.paramAsBoolean(LICENSE_SPEC_VIEW_MODE, false);
        boolean restViewMode = params.paramAsBoolean(REST_VIEW_MODE, false);
        boolean previouslyHumanReadable = builder.humanReadable();
        if (licenseSpecMode && restViewMode) {
            throw new IllegalArgumentException("can have either " + REST_VIEW_MODE + " or " +
LICENSE_SPEC_VIEW_MODE);
        } else if (restViewMode) {
            if (!previouslyHumanReadable) {
                builder.humanReadable(true);
            }
        }
        final int version;
        if (params.param(LICENSE_VERSION_MODE) != null && restViewMode) {
            version = Integer.parseInt(params.param(LICENSE_VERSION_MODE));
        } else {
            version = this.version;
        }
        if (restViewMode) {
            builder.field(Fields.STATUS, status().label());
        }
```

```
    builder.field(Fields.UID, uid);
    builder.field(Fields.TYPE, type);
    if (version == VERSION_START) {
        builder.field(Fields.SUBSCRIPTION_TYPE, subscriptionType);
    }
    builder.timeField(Fields.ISSUE_DATE_IN_MILLIS, Fields.ISSUE_DATE, issueDate);
    if (version == VERSION_START) {
        builder.field(Fields.FEATURE, feature);
    }

    if (expiryDate != LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS) {
        builder.timeField(Fields.EXPIRY_DATE_IN_MILLIS, Fields.EXPIRY_DATE, expiryDate);
    }
    builder.field(Fields.MAX_NODES, maxNodes);
    builder.field(Fields.ISSUED_TO, issuedTo);
    builder.field(Fields.ISSUER, issuer);
    if (!licenseSpecMode && !restViewMode && signature != null) {
        builder.field(Fields.SIGNATURE, signature);
    }
    if (restViewMode) {
        builder.humanReadable(previouslyHumanReadable);
    }
    if (version >= VERSION_START_DATE) {
        builder.timeField(Fields.START_DATE_IN_MILLIS, Fields.START_DATE, startDate);
    }
    return builder;
}

public static License fromXContent(XContentParser parser) throws IOException {
    Builder builder = new Builder();
    XContentParser.Token token;
    while ((token = parser.nextToken()) != XContentParser.Token.END_OBJECT) {
        if (token == XContentParser.Token.FIELD_NAME) {
            String currentFieldName = parser.currentName();
            token = parser.nextToken();
            if (token.isValue()) {
                if (Fields.UID.equals(currentFieldName)) {
                    builder.uid(parser.text());
                } else if (Fields.TYPE.equals(currentFieldName)) {
                    builder.type(parser.text());
                } else if (Fields.SUBSCRIPTION_TYPE.equals(currentFieldName)) {
                    builder.subscriptionType(parser.text());
                } else if (Fields.ISSUE_DATE.equals(currentFieldName)) {
                    builder.issueDate(parseDate(parser, "issue", false));
                } else if (Fields.ISSUE_DATE_IN_MILLIS.equals(currentFieldName)) {
                    builder.issueDate(parser.longValue());
                } else if (Fields.FEATURE.equals(currentFieldName)) {
                    builder.feature(parser.text());
```

```
                } else if (Fields.EXPIRY_DATE.equals(currentFieldName)) {
                    builder.expiryDate(parseDate(parser, "expiration", true));
                } else if (Fields.EXPIRY_DATE_IN_MILLIS.equals(currentFieldName)) {
                    builder.expiryDate(parser.longValue());
                } else if (Fields.START_DATE.equals(currentFieldName)) {
                    builder.startDate(parseDate(parser, "start", false));
                } else if (Fields.START_DATE_IN_MILLIS.equals(currentFieldName)) {
                    builder.startDate(parser.longValue());
                } else if (Fields.MAX_NODES.equals(currentFieldName)) {
                    builder.maxNodes(parser.intValue());
                } else if (Fields.ISSUED_TO.equals(currentFieldName)) {
                    builder.issuedTo(parser.text());
                } else if (Fields.ISSUER.equals(currentFieldName)) {
                    builder.issuer(parser.text());
                } else if (Fields.SIGNATURE.equals(currentFieldName)) {
                    builder.signature(parser.text());
                } else if (Fields.VERSION.equals(currentFieldName)) {
                    builder.version(parser.intValue());
                }
                // Ignore unknown elements - might be new version of license
            } else if (token == XContentParser.Token.START_ARRAY) {
                // It was probably created by newer version - ignoring
                parser.skipChildren();
            } else if (token == XContentParser.Token.START_OBJECT) {
                // It was probably created by newer version - ignoring
                parser.skipChildren();
            }
        }
    }
    // not a license spec
    if (builder.signature != null) {
        byte[] signatureBytes = Base64.getDecoder().decode(builder.signature);
        ByteBuffer byteBuffer = ByteBuffer.wrap(signatureBytes);
        int version = byteBuffer.getInt();
        // we take the absolute version, because negative versions
        // mean that the license was generated by the cluster (see TrialLicense)
        // and positive version means that the license was signed
        if (version < 0) {
            version *= -1;
        }
        if (version == 0) {
            throw new ElasticsearchException("malformed signature for license [" + builder.uid + "]");
        } else if (version > VERSION_CURRENT) {
            throw new ElasticsearchException("Unknown license version found, please upgrade all nodes to the latest
" +
                    "elasticsearch-license plugin");
        }
        // signature version is the source of truth
```

```
        builder.version(version);
    }
    return builder.build();
}

/**
 * Returns true if the license was auto-generated (by license plugin),
 * false otherwise
 */
public static boolean isAutoGeneratedLicense(String signature) {
    try {
        byte[] signatureBytes = Base64.getDecoder().decode(signature);
        ByteBuffer byteBuffer = ByteBuffer.wrap(signatureBytes);
        return byteBuffer.getInt() < 0;
    } catch (IllegalArgumentException e) {
        throw new IllegalStateException(e);
    }
}

public static License fromSource(BytesReference bytes, XContentType xContentType) throws IOException {
    if (bytes == null || bytes.length() == 0) {
        throw new ElasticsearchParseException("failed to parse license - no content provided");
    }
    if (xContentType == null) {
        throw new ElasticsearchParseException("failed to parse license - no content-type provided");
    }
    // EMPTY is safe here because we don't call namedObject
    try (InputStream byteStream = bytes.streamInput();
         XContentParser parser = xContentType.xContent()
            .createParser(NamedXContentRegistry.EMPTY, LoggingDeprecationHandler.INSTANCE, byteStream))
    {
        License license = null;
        if (parser.nextToken() == XContentParser.Token.START_OBJECT) {
            if (parser.nextToken() == XContentParser.Token.FIELD_NAME) {
                String currentFieldName = parser.currentName();
                if (Fields.LICENSES.equals(currentFieldName)) {
                    final List<License> pre20Licenses = new ArrayList<>();
                    if (parser.nextToken() == XContentParser.Token.START_ARRAY) {
                        while (parser.nextToken() != XContentParser.Token.END_ARRAY) {
                            pre20Licenses.add(License.fromXContent(parser));
                        }
                        // take the latest issued unexpired license
                        CollectionUtil.timSort(pre20Licenses, LATEST_ISSUE_DATE_FIRST);
                        long now = System.currentTimeMillis();
                        for (License oldLicense : pre20Licenses) {
                            if (oldLicense.expiryDate() > now) {
                                license = oldLicense;
                                break;
```

```
                }
            }
            if (license == null && !pre20Licenses.isEmpty()) {
                license = pre20Licenses.get(0);
            }
        } else {
            throw new ElasticsearchParseException("failed to parse licenses expected an array of licenses");
        }
    } else if (Fields.LICENSE.equals(currentFieldName)) {
        license = License.fromXContent(parser);
    }
    // Ignore all other fields - might be created with new version
  } else {
    throw new ElasticsearchParseException("failed to parse licenses expected field");
  }
} else {
  throw new ElasticsearchParseException("failed to parse licenses expected start object");
}
return license;
}
}

@Override
public boolean equals(Object o) {
    if (this == o) return true;
    if (o == null || getClass() != o.getClass()) return false;

    License license = (License) o;

    if (issueDate != license.issueDate) return false;
    if (expiryDate != license.expiryDate) return false;
    if (startDate!= license.startDate) return false;
    if (maxNodes != license.maxNodes) return false;
    if (version != license.version) return false;
    if (uid != null ? !uid.equals(license.uid) : license.uid != null) return false;
    if (issuer != null ? !issuer.equals(license.issuer) : license.issuer != null) return false;
    if (issuedTo != null ? !issuedTo.equals(license.issuedTo) : license.issuedTo != null) return false;
    if (type != null ? !type.equals(license.type) : license.type != null) return false;
    if (subscriptionType != null ? !subscriptionType.equals(license.subscriptionType) : license.subscriptionType !=
null)
        return false;
    if (feature != null ? !feature.equals(license.feature) : license.feature != null) return false;
    return !(signature != null ? !signature.equals(license.signature) : license.signature != null);

}

@Override
public int hashCode() {
```

```
        int result = uid != null ? uid.hashCode() : 0;
        result = 31 * result + (issuer != null ? issuer.hashCode() : 0);
        result = 31 * result + (issuedTo != null ? issuedTo.hashCode() : 0);
        result = 31 * result + (int) (issueDate ^ (issueDate >>> 32));
        result = 31 * result + (type != null ? type.hashCode() : 0);
        result = 31 * result + (subscriptionType != null ? subscriptionType.hashCode() : 0);
        result = 31 * result + (feature != null ? feature.hashCode() : 0);
        result = 31 * result + (signature != null ? signature.hashCode() : 0);
        result = 31 * result + (int) (expiryDate ^ (expiryDate >>> 32));
        result = 31 * result + (int) (startDate ^ (startDate>>> 32));
        result = 31 * result + maxNodes;
        result = 31 * result + version;
        return result;
    }

    public static final class Fields {
        public static final String STATUS = "status";
        public static final String UID = "uid";
        public static final String TYPE = "type";
        public static final String SUBSCRIPTION_TYPE = "subscription_type";
        public static final String ISSUE_DATE_IN_MILLIS = "issue_date_in_millis";
        public static final String ISSUE_DATE = "issue_date";
        public static final String FEATURE = "feature";
        public static final String EXPIRY_DATE_IN_MILLIS = "expiry_date_in_millis";
        public static final String EXPIRY_DATE = "expiry_date";
        public static final String START_DATE_IN_MILLIS = "start_date_in_millis";
        public static final String START_DATE = "start_date";
        public static final String MAX_NODES = "max_nodes";
        public static final String ISSUED_TO = "issued_to";
        public static final String ISSUER = "issuer";
        public static final String VERSION = "version";
        public static final String SIGNATURE = "signature";

        public static final String LICENSES = "licenses";
        public static final String LICENSE = "license";

    }

    private static long parseDate(XContentParser parser, String description, boolean endOfTheDay) throws
IOException {
        if (parser.currentToken() == XContentParser.Token.VALUE_NUMBER) {
            return parser.longValue();
        } else {
            try {
                if (endOfTheDay) {
                    return DateUtils.endOfTheDay(parser.text());
                } else {
                    return DateUtils.beginningOfTheDay(parser.text());
```

```
            }
        } catch (IllegalArgumentException ex) {
            throw new ElasticsearchParseException("invalid " + description + " date format " + parser.text());
        }
    }
}

public static Builder builder() {
    return new Builder();
}

public static class Builder {
    private int version = License.VERSION_CURRENT;
    private String uid;
    private String issuer;
    private String issuedTo;
    private long issueDate = -1;
    private String type;
    private String subscriptionType;
    private String feature;
    private String signature;
    private long expiryDate = -1;
    private long startDate = -1;
    private int maxNodes = -1;

    public Builder uid(String uid) {
        this.uid = uid;
        return this;
    }

    public Builder version(int version) {
        this.version = version;
        return this;
    }

    public Builder issuer(String issuer) {
        this.issuer = issuer;
        return this;
    }

    public Builder issuedTo(String issuedTo) {
        this.issuedTo = issuedTo;
        return this;
    }

    public Builder issueDate(long issueDate) {
        this.issueDate = issueDate;
        return this;
```

```
}

public Builder type(String type) {
    this.type = type;
    return this;
}

public Builder subscriptionType(String subscriptionType) {
    this.subscriptionType = subscriptionType;
    return this;
}

public Builder feature(String feature) {
    this.feature = feature;
    return this;
}

public Builder expiryDate(long expiryDate) {
    this.expiryDate = expiryDate;
    return this;
}

public Builder maxNodes(int maxNodes) {
    this.maxNodes = maxNodes;
    return this;
}

public Builder signature(String signature) {
    if (signature != null) {
        this.signature = signature;
    }
    return this;
}

public Builder startDate(long startDate) {
    this.startDate = startDate;
    return this;
}

public Builder fromLicenseSpec(License license, String signature) {
    return uid(license.uid())
            .version(license.version())
            .issuedTo(license.issuedTo())
            .issueDate(license.issueDate())
            .startDate(license.startDate())
            .type(license.type())
            .subscriptionType(license.subscriptionType)
            .feature(license.feature)
```

```
            .maxNodes(license.maxNodes())
            .expiryDate(license.expiryDate())
            .issuer(license.issuer())
            .signature(signature);
    }

    /**
     * Returns a builder that converts pre 2.0 licenses
     * to the new license format
     */
    public Builder fromPre20LicenseSpec(License pre20License) {
        return uid(pre20License.uid())
            .issuedTo(pre20License.issuedTo())
            .issueDate(pre20License.issueDate())
            .maxNodes(pre20License.maxNodes())
            .expiryDate(pre20License.expiryDate());
    }

    public License build() {
        return new License(version, uid, issuer, issuedTo, issueDate, type,
            subscriptionType, feature, signature, expiryDate, maxNodes, startDate);
    }

    public Builder validate() {
        if (issuer == null) {
            throw new IllegalStateException("issuer can not be null");
        } else if (issuedTo == null) {
            throw new IllegalStateException("issuedTo can not be null");
        } else if (issueDate == -1) {
            throw new IllegalStateException("issueDate has to be set");
        } else if (type == null) {
            throw new IllegalStateException("type can not be null");
        } else if (uid == null) {
            throw new IllegalStateException("uid can not be null");
        } else if (signature == null) {
            throw new IllegalStateException("signature can not be null");
        } else if (maxNodes == -1) {
            throw new IllegalStateException("maxNodes has to be set");
        } else if (expiryDate == -1) {
            throw new IllegalStateException("expiryDate has to be set");
        }
        return this;
    }
}

/**
 * Returns <code>true</code> iff the license is a production licnese
 */
```

```
  public boolean isProductionLicense() {
    switch (operationMode()) {
      case MISSING:
      case TRIAL:
      case BASIC:
        return false;
      case STANDARD:
      case GOLD:
      case PLATINUM:
        return true;
      default:
        throw new AssertionError("unknown operation mode: " + operationMode());

    }
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.common.Strings;
import org.elasticsearch.common.logging.LoggerMessageFormat;
import org.elasticsearch.common.logging.Loggers;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.license.License.OperationMode;
import org.elasticsearch.xpack.core.XPackField;
import org.elasticsearch.xpack.core.XPackSettings;
import org.elasticsearch.xpack.core.monitoring.MonitoringField;

import java.util.Collections;
import java.util.LinkedHashMap;
import java.util.List;
import java.util.Map;
import java.util.Objects;
import java.util.concurrent.CopyOnWriteArrayList;
import java.util.function.BiFunction;

/**
* A holder for the current state of the license for all xpack features.
*/
public class XPackLicenseState {

  /** Messages for each feature which are printed when the license expires. */
```

```
static final Map<String, String[]> EXPIRATION_MESSAGES;
static {
    Map<String, String[]> messages = new LinkedHashMap<>();
    messages.put(XPackField.SECURITY, new String[] {
        "Cluster health, cluster stats and indices stats operations are blocked",
        "All data operations (read and write) continue to work"
    });
    messages.put(XPackField.WATCHER, new String[] {
        "PUT / GET watch APIs are disabled, DELETE watch API continues to work",
        "Watches execute and write to the history",
        "The actions of the watches don't execute"
    });
    messages.put(XPackField.MONITORING, new String[] {
        "The agent will stop collecting cluster and indices metrics",
        "The agent will stop automatically cleaning indices older than [xpack.monitoring.history.duration]"
    });
    messages.put(XPackField.GRAPH, new String[] {
        "Graph explore APIs are disabled"
    });
    messages.put(XPackField.MACHINE_LEARNING, new String[] {
        "Machine learning APIs are disabled"
    });
    messages.put(XPackField.LOGSTASH, new String[] {
        "Logstash will continue to poll centrally-managed pipelines"
    });
    messages.put(XPackField.BEATS, new String[] {
        "Beats will continue to poll centrally-managed configuration"
    });
    messages.put(XPackField.DEPRECATION, new String[] {
        "Deprecation APIs are disabled"
    });
    messages.put(XPackField.UPGRADE, new String[] {
        "Upgrade API is disabled"
    });
    messages.put(XPackField.SQL, new String[] {
        "SQL support is disabled"
    });
    messages.put(XPackField.ROLLUP, new String[] {
        "Creating and Starting rollup jobs will no longer be allowed.",
        "Stopping/Deleting existing jobs, RollupCaps API and RollupSearch continue to function."
    });
    EXPIRATION_MESSAGES = Collections.unmodifiableMap(messages);
}

/**
 * Messages for each feature which are printed when the license type changes.
 * The value is a function taking the old and new license type, and returns the messages for that feature.
 */
```

```
  static final Map<String, BiFunction<OperationMode, OperationMode, String[]>>
ACKNOWLEDGMENT_MESSAGES;
  static {
    Map<String, BiFunction<OperationMode, OperationMode, String[]>> messages = new LinkedHashMap<>();
    messages.put(XPackField.SECURITY, XPackLicenseState::securityAcknowledgementMessages);
    messages.put(XPackField.WATCHER, XPackLicenseState::watcherAcknowledgementMessages);
    messages.put(XPackField.MONITORING, XPackLicenseState::monitoringAcknowledgementMessages);
    messages.put(XPackField.GRAPH, XPackLicenseState::graphAcknowledgementMessages);
    messages.put(XPackField.MACHINE_LEARNING,
XPackLicenseState::machineLearningAcknowledgementMessages);
    messages.put(XPackField.LOGSTASH, XPackLicenseState::logstashAcknowledgementMessages);
    messages.put(XPackField.BEATS, XPackLicenseState::beatsAcknowledgementMessages);
    messages.put(XPackField.SQL, XPackLicenseState::sqlAcknowledgementMessages);
    ACKNOWLEDGMENT_MESSAGES = Collections.unmodifiableMap(messages);
  }

  private static String[] securityAcknowledgementMessages(OperationMode currentMode, OperationMode
newMode) {
    switch (newMode) {
      case BASIC:
        switch (currentMode) {
          case TRIAL:
          case STANDARD:
          case GOLD:
          case PLATINUM:
            return new String[] {
              "The following X-Pack security functionality will be disabled: authentication, authorization, " +
                "ip filtering, and auditing. Please restart your node after applying the license.",
              "Field and document level access control will be disabled.",
              "Custom realms will be ignored."
            };
        }
        break;
      case GOLD:
        switch (currentMode) {
          case BASIC:
          case STANDARD:
            // ^^ though technically it was already disabled, it's not bad to remind them
          case TRIAL:
          case PLATINUM:
            return new String[] {
              "Field and document level access control will be disabled.",
              "Custom realms will be ignored."
            };
        }
        break;
      case STANDARD:
        switch (currentMode) {
```

```
            case BASIC:
                // ^^ though technically it was already disabled, it's not bad to remind them
            case GOLD:
            case PLATINUM:
            case TRIAL:
                return new String[] {
                    "Authentication will be limited to the native realms.",
                    "IP filtering and auditing will be disabled.",
                    "Field and document level access control will be disabled.",
                    "Custom realms will be ignored."
                };
            }
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] watcherAcknowledgementMessages(OperationMode currentMode, OperationMode
newMode) {
        switch (newMode) {
            case BASIC:
                switch (currentMode) {
                    case TRIAL:
                    case STANDARD:
                    case GOLD:
                    case PLATINUM:
                        return new String[] { "Watcher will be disabled" };
                }
                break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] monitoringAcknowledgementMessages(OperationMode currentMode, OperationMode
newMode) {
        switch (newMode) {
            case BASIC:
                switch (currentMode) {
                    case TRIAL:
                    case STANDARD:
                    case GOLD:
                    case PLATINUM:
                        return new String[] {
                            LoggerMessageFormat.format(
                                "Multi-cluster support is disabled for clusters with [{}] license. If you are\n" +
                                    "running multiple clusters, users won't be able to access the clusters with\n" +
                                    "[{}] licenses from within a single X-Pack Kibana instance. You will have to deploy a\n" +
                                    "separate and dedicated X-pack Kibana instance for each [{}] cluster you wish to monitor.",
                                newMode, newMode, newMode),
```

```
                    LoggerMessageFormat.format(
                        "Automatic index cleanup is locked to {} days for clusters with [{}] license.",
                        MonitoringField.HISTORY_DURATION.getDefault(Settings.EMPTY).days(), newMode)
                };
            }
            break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] graphAcknowledgementMessages(OperationMode currentMode, OperationMode newMode)
{
        switch (newMode) {
            case BASIC:
            case STANDARD:
            case GOLD:
                switch (currentMode) {
                    case TRIAL:
                    case PLATINUM:
                        return new String[] { "Graph will be disabled" };
                }
                break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] machineLearningAcknowledgementMessages(OperationMode currentMode,
OperationMode newMode) {
        switch (newMode) {
            case BASIC:
            case STANDARD:
            case GOLD:
                switch (currentMode) {
                    case TRIAL:
                    case PLATINUM:
                        return new String[] { "Machine learning will be disabled" };
                }
                break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] logstashAcknowledgementMessages(OperationMode currentMode, OperationMode
newMode) {
        switch (newMode) {
            case BASIC:
                if (isBasic(currentMode) == false) {
                    return new String[] { "Logstash will no longer poll for centrally-managed pipelines" };
```

```
            }
            break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] beatsAcknowledgementMessages(OperationMode currentMode, OperationMode newMode)
{
        switch (newMode) {
            case BASIC:
                if (isBasic(currentMode) == false) {
                    return new String[] { "Beats will no longer be able to use centrally-managed configuration" };
                }
                break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static String[] sqlAcknowledgementMessages(OperationMode currentMode, OperationMode newMode) {
        switch (newMode) {
            case BASIC:
            case STANDARD:
            case GOLD:
                switch (currentMode) {
                    case TRIAL:
                    case PLATINUM:
                        return new String[] {
                                "JDBC and ODBC support will be disabled, but you can continue to use SQL CLI and REST
endpoint" };
                }
                break;
        }
        return Strings.EMPTY_ARRAY;
    }

    private static boolean isBasic(OperationMode mode) {
        return mode == OperationMode.BASIC;
    }

    /** A wrapper for the license mode and state, to allow atomically swapping. */
    private static class Status {

        /** The current "mode" of the license (ie license type). */
        final OperationMode mode;

        /** True if the license is active, or false if it is expired. */
        final boolean active;
```

```
        Status(OperationMode mode, boolean active) {
            this.mode = mode;
            this.active = active;
        }
    }

    private final List<LicenseStateListener> listeners;
    private final boolean isSecurityEnabled;
    private final boolean isSecurityExplicitlyEnabled;

    private Status status = new Status(OperationMode.TRIAL, true);
    private boolean isSecurityEnabledByTrialVersion;

    public XPackLicenseState(Settings settings) {
        this.listeners = new CopyOnWriteArrayList<>();
        this.isSecurityEnabled = XPackSettings.SECURITY_ENABLED.get(settings);
        // 6.0+ requires TLS for production licenses, so if TLS is enabled and security is enabled
        // we can interpret this as an explicit enabling of security if the security enabled
        // setting is not explicitly set
        this.isSecurityExplicitlyEnabled = isSecurityEnabled &&
            (settings.hasValue(XPackSettings.SECURITY_ENABLED.getKey()) ||
XPackSettings.TRANSPORT_SSL_ENABLED.get(settings));
        this.isSecurityEnabledByTrialVersion = false;
    }

    private XPackLicenseState(XPackLicenseState xPackLicenseState) {
        this.listeners = xPackLicenseState.listeners;
        this.isSecurityEnabled = xPackLicenseState.isSecurityEnabled;
        this.isSecurityExplicitlyEnabled = xPackLicenseState.isSecurityExplicitlyEnabled;
        this.status = xPackLicenseState.status;
        this.isSecurityEnabledByTrialVersion = xPackLicenseState.isSecurityEnabledByTrialVersion;
    }

    /**
     * Updates the current state of the license, which will change what features are available.
     *
     * @param mode   The mode (type) of the current license.
     * @param active True if the current license exists and is within its allowed usage period; false if it is expired or
missing.
     * @param mostRecentTrialVersion If this cluster has, at some point commenced a trial, the most recent version
on which they did that.
     *                               May be {@code null} if they have never generated a trial license on this cluster, or the most
recent
     *                               trial was prior to this metadata being tracked (6.1)
     */
    void update(OperationMode mode, boolean active, @Nullable Version mostRecentTrialVersion) {
        synchronized (this) {
            status = new Status(mode, active);
```

```
            if (isSecurityEnabled == true && isSecurityExplicitlyEnabled == false && mode == OperationMode.TRIAL
                && isSecurityEnabledByTrialVersion == false) {
                // Before 6.3, Trial licenses would default having security enabled.
                // If this license was generated before that version, then treat it as if security is explicitly enabled
                if (mostRecentTrialVersion == null || mostRecentTrialVersion.before(Version.V_6_3_0)) {
                    Loggers.getLogger(getClass()).info("Automatically enabling security for older trial license ({})",
                        mostRecentTrialVersion == null ? "[pre 6.1.0]" : mostRecentTrialVersion.toString());
                    isSecurityEnabledByTrialVersion = true;
                }
            }
        }
        listeners.forEach(LicenseStateListener::licenseStateChanged);
    }

    /** Add a listener to be notified on license change */
    public void addListener(final LicenseStateListener listener) {
        listeners.add(Objects.requireNonNull(listener));
    }

    /** Remove a listener */
    public void removeListener(final LicenseStateListener listener) {
        listeners.remove(Objects.requireNonNull(listener));
    }

    /** Return the current license type. */
    public synchronized OperationMode getOperationMode() {
        return status.mode;
    }

    /** Return true if the license is currently within its time boundaries, false otherwise. */
    public synchronized boolean isActive() {
        return status.active;
    }

    /**
     * @return true if authentication and authorization should be enabled. this does not indicate what realms are
available
     * @see #allowedRealmType() for the enabled realms
     */
    public synchronized boolean isAuthAllowed() {
        OperationMode mode = status.mode;
        final boolean isSecurityCurrentlyEnabled =
            isSecurityEnabled(mode, isSecurityExplicitlyEnabled, isSecurityEnabledByTrialVersion, isSecurityEnabled);
        return isSecurityCurrentlyEnabled && (mode == OperationMode.STANDARD || mode ==
OperationMode.GOLD
            || mode == OperationMode.PLATINUM || mode == OperationMode.TRIAL);
    }
```

```
    /**
     * @return true if IP filtering should be enabled
     */
    public synchronized boolean isIpFilteringAllowed() {
        OperationMode mode = status.mode;
        final boolean isSecurityCurrentlyEnabled =
            isSecurityEnabled(mode, isSecurityExplicitlyEnabled, isSecurityEnabledByTrialVersion, isSecurityEnabled);
        return isSecurityCurrentlyEnabled && (mode == OperationMode.GOLD || mode ==
OperationMode.PLATINUM || mode == OperationMode.TRIAL);
    }

    /**
     * @return true if auditing should be enabled
     */
    public synchronized boolean isAuditingAllowed() {
        OperationMode mode = status.mode;
        final boolean isSecurityCurrentlyEnabled =
            isSecurityEnabled(mode, isSecurityExplicitlyEnabled, isSecurityEnabledByTrialVersion, isSecurityEnabled);
        return isSecurityCurrentlyEnabled && (mode == OperationMode.GOLD || mode ==
OperationMode.PLATINUM || mode == OperationMode.TRIAL);
    }

    /**
     * Indicates whether the stats and health API calls should be allowed. If a license is expired and past the grace
     * period then we deny these calls.
     *
     * @return true if the license allows for the stats and health APIs to be used.
     */
    public synchronized boolean isStatsAndHealthAllowed() {
        return status.active;
    }

    /**
     * Determine if Document Level Security (DLS) and Field Level Security (FLS) should be enabled.
     * <p>
     * DLS and FLS are only disabled when the mode is not:
     * <ul>
     * <li>{@link OperationMode#PLATINUM}</li>
     * <li>{@link OperationMode#TRIAL}</li>
     * </ul>
     * Note: This does not consider the <em>state</em> of the license so that Security does not suddenly leak
information!
     *
     * @return {@code true} to enable DLS and FLS. Otherwise {@code false}.
     */
    public synchronized boolean isDocumentAndFieldLevelSecurityAllowed() {
        OperationMode mode = status.mode;
        final boolean isSecurityCurrentlyEnabled =
```

```
        isSecurityEnabled(mode, isSecurityExplicitlyEnabled, isSecurityEnabledByTrialVersion, isSecurityEnabled);
    return isSecurityCurrentlyEnabled && (mode == OperationMode.TRIAL || mode ==
OperationMode.PLATINUM);
  }

  /** Classes of realms that may be available based on the license type. */
  public enum AllowedRealmType {
    NONE,
    NATIVE,
    DEFAULT,
    ALL
  }

  /**
   * @return the type of realms that are enabled based on the license {@link OperationMode}
   */
  public synchronized AllowedRealmType allowedRealmType() {
    final boolean isSecurityCurrentlyEnabled =
        isSecurityEnabled(status.mode, isSecurityExplicitlyEnabled, isSecurityEnabledByTrialVersion,
isSecurityEnabled);
    if (isSecurityCurrentlyEnabled) {
      switch (status.mode) {
        case PLATINUM:
        case TRIAL:
          return AllowedRealmType.ALL;
        case GOLD:
          return AllowedRealmType.DEFAULT;
        case STANDARD:
          return AllowedRealmType.NATIVE;
        default:
          return AllowedRealmType.NONE;
      }
    } else {
      return AllowedRealmType.NONE;
    }
  }

  /**
   * @return whether custom role providers are allowed based on the license {@link OperationMode}
   */
  public synchronized boolean isCustomRoleProvidersAllowed() {
    final boolean isSecurityCurrentlyEnabled =
        isSecurityEnabled(status.mode, isSecurityExplicitlyEnabled, isSecurityEnabledByTrialVersion,
isSecurityEnabled);
    return isSecurityCurrentlyEnabled && (status.mode == OperationMode.PLATINUM || status.mode ==
OperationMode.TRIAL)
            && status.active;
  }
```

```
/**
 * @return whether "authorization_realms" are allowed based on the license {@link OperationMode}
 * @see org.elasticsearch.xpack.core.security.authc.support.DelegatedAuthorizationSettings
 */
public boolean isAuthorizationRealmAllowed() {
    final Status localStatus = status;
    return (localStatus.mode == OperationMode.PLATINUM || localStatus.mode == OperationMode.TRIAL)
        && localStatus.active;
}

/**
 * Determine if Watcher is available based on the current license.
 * <p>
 * Watcher is available if the license is active (hasn't expired) and of one of the following types:
 * <ul>
 * <li>{@link OperationMode#STANDARD}</li>
 * <li>{@link OperationMode#PLATINUM}</li>
 * <li>{@link OperationMode#GOLD}</li>
 * <li>{@link OperationMode#TRIAL}</li>
 * </ul>
 *
 * @return {@code true} as long as the license is valid. Otherwise {@code false}.
 */
public synchronized boolean isWatcherAllowed() {
    Status localStatus = status;

    if (localStatus.active == false) {
        return false;
    }

    switch (localStatus.mode) {
        case TRIAL:
        case GOLD:
        case PLATINUM:
        case STANDARD:
            return true;
        default:
            return false;
    }
}

/**
 * Monitoring is always available as long as there is a valid license
 *
 * @return true if the license is active
 */
public synchronized boolean isMonitoringAllowed() {
```

```
    return status.active;
}

/**
 * Monitoring Cluster Alerts requires the equivalent license to use Watcher.
 *
 * @return {@link #isWatcherAllowed()}
 * @see #isWatcherAllowed()
 */
public synchronized boolean isMonitoringClusterAlertsAllowed() {
    return isWatcherAllowed();
}

/**
 * Determine if the current license allows the retention of indices to be modified.
 * <p>
 * Only users with a non-{@link OperationMode#BASIC} license can update the retention period.
 * <p>
 * Note: This does not consider the <em>state</em> of the license so that any change is remembered for when
they fix their license.
 *
 * @return {@code true} if the user is allowed to modify the retention. Otherwise {@code false}.
 */
public synchronized boolean isUpdateRetentionAllowed() {
    final OperationMode mode = status.mode;
    return mode != OperationMode.BASIC && mode != OperationMode.MISSING;
}

/**
 * Determine if Graph Exploration should be enabled.
 * <p>
 * Exploration is only disabled when the license has expired or if the mode is not:
 * <ul>
 * <li>{@link OperationMode#PLATINUM}</li>
 * <li>{@link OperationMode#TRIAL}</li>
 * </ul>
 *
 * @return {@code true} as long as the license is valid. Otherwise {@code false}.
 */
public synchronized boolean isGraphAllowed() {
    Status localStatus = status;
    OperationMode operationMode = localStatus.mode;

    boolean licensed = operationMode == OperationMode.TRIAL || operationMode ==
OperationMode.PLATINUM;

    return licensed && localStatus.active;
}
```

```
/**
 * Determine if Machine Learning should be enabled.
 * <p>
 * Machine Learning is only disabled when the license has expired or if the
 * mode is not:
 * <ul>
 * <li>{@link OperationMode#PLATINUM}</li>
 * <li>{@link OperationMode#TRIAL}</li>
 * </ul>
 *
 * @return {@code true} as long as the license is valid. Otherwise
 *         {@code false}.
 */
public synchronized boolean isMachineLearningAllowed() {
    final Status currentStatus = status;
    return currentStatus.active && isMachineLearningAllowedForOperationMode(currentStatus.mode);
}

public static boolean isMachineLearningAllowedForOperationMode(final OperationMode operationMode) {
    return isPlatinumOrTrialOperationMode(operationMode);
}

/**
 * Rollup is always available as long as there is a valid license
 *
 * @return true if the license is active
 */
public synchronized boolean isRollupAllowed() {
    return status.active;
}

/**
 * Logstash is allowed as long as there is an active license of type TRIAL, STANDARD, GOLD or PLATINUM
 * @return {@code true} as long as there is a valid license
 */
public synchronized boolean isLogstashAllowed() {
    Status localStatus = status;
    return localStatus.active && (isBasic(localStatus.mode) == false);
}

/**
 * Beats is allowed as long as there is an active license of type TRIAL, STANDARD, GOLD or PLATINUM
 * @return {@code true} as long as there is a valid license
 */
public boolean isBeatsAllowed() {
    Status localStatus = status;
    return localStatus.active && (isBasic(localStatus.mode) == false);
```

```
    }

    /**
     * Deprecation APIs are always allowed as long as there is an active license
     * @return {@code true} as long as there is a valid license
     */
    public synchronized boolean isDeprecationAllowed() {
        return status.active;
    }

    /**
     * Determine if Upgrade API should be enabled.
     * <p>
     *  Upgrade API is not available in for all license types except {@link OperationMode#MISSING}
     *
     * @return {@code true} as long as the license is valid. Otherwise
     *         {@code false}.
     */
    public synchronized boolean isUpgradeAllowed() {
        // Should work on all active licenses
        return status.active;
    }

    /**
     * Determine if SQL support should be enabled.
     * <p>
     *  SQL is available for all license types except {@link OperationMode#MISSING}
     */
    public synchronized boolean isSqlAllowed() {
        return status.active;
    }

    /**
     * Determine if JDBC support should be enabled.
     * <p>
     *  JDBC is available only in for {@link OperationMode#PLATINUM} and {@link OperationMode#TRIAL}
licences
     */
    public synchronized boolean isJdbcAllowed() {
        Status localStatus = status;
        OperationMode operationMode = localStatus.mode;

        boolean licensed = operationMode == OperationMode.TRIAL || operationMode ==
OperationMode.PLATINUM;

        return licensed && localStatus.active;
    }
```

```
    /**
     * Determine if ODBC support should be enabled.
     * <p>
     * ODBC is available only in for {@link OperationMode#PLATINUM} and {@link OperationMode#TRIAL}
licences
     */
    public synchronized boolean isOdbcAllowed() {
        Status localStatus = status;
        OperationMode operationMode = localStatus.mode;

        boolean licensed = operationMode == OperationMode.TRIAL || operationMode ==
OperationMode.PLATINUM;

        return licensed && localStatus.active;
    }

    public synchronized boolean isTrialLicense() {
        return status.mode == OperationMode.TRIAL;
    }

    /**
     * @return true if security is available to be used with the current license type
     */
    public synchronized boolean isSecurityAvailable() {
        OperationMode mode = status.mode;
        return mode == OperationMode.GOLD || mode == OperationMode.PLATINUM || mode ==
OperationMode.STANDARD ||
                mode == OperationMode.TRIAL;
    }

    /**
     * @return true if security has been disabled by a trial license which is the case of the
     *        default distribution post 6.3.0. The conditions necessary for this are:
     *        <ul>
     *            <li>A trial license generated in 6.3.0+</li>
     *            <li>xpack.security.enabled not specified as a setting</li>
     *        </ul>
     */
    public synchronized boolean isSecurityDisabledByTrialLicense() {
        return status.mode == OperationMode.TRIAL && isSecurityEnabled
            && isSecurityExplicitlyEnabled == false
            && isSecurityEnabledByTrialVersion == false;
    }

    private static boolean isSecurityEnabled(final OperationMode mode, final boolean isSecurityExplicitlyEnabled,
                                             final boolean isSecurityEnabledByTrialVersion, final boolean isSecurityEnabled) {
        return mode == OperationMode.TRIAL ? (isSecurityExplicitlyEnabled || isSecurityEnabledByTrialVersion) :
```

```
isSecurityEnabled;
  }

  /**
   * Determine if cross-cluster replication should be enabled.
   * <p>
   * Cross-cluster replication is only disabled when the license has expired or if the mode is not:
   * <ul>
   * <li>{@link OperationMode#PLATINUM}</li>
   * <li>{@link OperationMode#TRIAL}</li>
   * </ul>
   *
   * @return true is the license is compatible, otherwise false
   */
  public synchronized boolean isCcrAllowed() {
      final Status currentStatus = status;
      return currentStatus.active && isCcrAllowedForOperationMode(currentStatus.mode);
  }

  public static boolean isCcrAllowedForOperationMode(final OperationMode operationMode) {
      return isPlatinumOrTrialOperationMode(operationMode);
  }

  public static boolean isPlatinumOrTrialOperationMode(final OperationMode operationMode) {
      return operationMode == OperationMode.PLATINUM || operationMode == OperationMode.TRIAL;
  }

  /**
   * Creates a copy of this object based on the state at the time the method was called. The
   * returned object will not be modified by a license update/expiration so it can be used to
   * make multiple method calls on the license state safely. This object should not be long
   * lived but instead used within a method when a consistent view of the license state
   * is needed for multiple interactions with the license state.
   */
  public synchronized XPackLicenseState copyCurrentLicenseState() {
      return new XPackLicenseState(this);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.ValidateActions;
import org.elasticsearch.action.support.master.AcknowledgedRequest;
```

```
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.common.xcontent.XContentType;

import java.io.IOException;


public class PutLicenseRequest extends AcknowledgedRequest<PutLicenseRequest> {

  private License license;
  private boolean acknowledge = false;

  public PutLicenseRequest() {
  }

  @Override
  public ActionRequestValidationException validate() {
    return (license == null) ? ValidateActions.addValidationError("license is missing", null) : null;
  }

  /**
   * Parses license from json format to an instance of {@link License}
   *
   * @param licenseDefinition licenses definition
   * @param xContentType the content type of the license
   */
  public PutLicenseRequest license(BytesReference licenseDefinition, XContentType xContentType) {
    try {
      return license(License.fromSource(licenseDefinition, xContentType));
    } catch (IOException e) {
      throw new IllegalArgumentException("failed to parse license source", e);
    }
  }

  public PutLicenseRequest license(License license) {
    this.license = license;
    return this;
  }

  public License license() {
    return license;
  }

  public PutLicenseRequest acknowledge(boolean acknowledge) {
    this.acknowledge = acknowledge;
    return this;
  }
```

```
    public boolean acknowledged() {
        return acknowledge;
    }

    @Override
    public void readFrom(StreamInput in) throws IOException {
        super.readFrom(in);
        license = License.readLicense(in);
        acknowledge = in.readBoolean();
    }

    @Override
    public void writeTo(StreamOutput out) throws IOException {
        super.writeTo(out);
        license.writeTo(out);
        out.writeBoolean(acknowledge);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.support.master.AcknowledgedRequestBuilder;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;

/**
* Register license request builder
*/
public class PutLicenseRequestBuilder extends AcknowledgedRequestBuilder<PutLicenseRequest,
PutLicenseResponse, PutLicenseRequestBuilder> {

    public PutLicenseRequestBuilder(ElasticsearchClient client) {
        this(client, PutLicenseAction.INSTANCE);
    }

    /**
     * Constructs register license request
     *
     * @param client elasticsearch client
     */
    public PutLicenseRequestBuilder(ElasticsearchClient client, PutLicenseAction action) {
```

```
        super(client, action, new PutLicenseRequest());
    }

    /**
     * Sets the license
     *
     * @param license license
     * @return this builder
     */
    public PutLicenseRequestBuilder setLicense(License license) {
        request.license(license);
        return this;
    }

    public PutLicenseRequestBuilder setLicense(BytesReference licenseSource, XContentType xContentType) {
        request.license(licenseSource, xContentType);
        return this;
    }

    public PutLicenseRequestBuilder setAcknowledge(boolean acknowledge) {
        request.acknowledge(acknowledge);
        return this;
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchSecurityException;
import org.elasticsearch.Version;
import org.elasticsearch.cluster.node.DiscoveryNodes;
import org.elasticsearch.rest.RestStatus;

import java.util.stream.StreamSupport;

public class LicenseUtils {

    public static final String EXPIRED_FEATURE_METADATA = "es.license.expired.feature";

    /**
     * Exception to be thrown when a feature action requires a valid license, but license
     * has expired
     *
     * <code>feature</code> accessible through {@link #EXPIRED_FEATURE_METADATA} in the
     * exception's rest header
```

```
     */
    public static ElasticsearchSecurityException newComplianceException(String feature) {
        ElasticsearchSecurityException e = new ElasticsearchSecurityException("current license is non-compliant for
[{}]",
                RestStatus.FORBIDDEN, feature);
        e.addMetadata(EXPIRED_FEATURE_METADATA, feature);
        return e;
    }

    /**
     * Checks if a given {@link ElasticsearchSecurityException} refers to a feature that
     * requires a valid license, but the license has expired.
     */
    public static boolean isLicenseExpiredException(ElasticsearchSecurityException exception) {
        return (exception != null) && (exception.getMetadata(EXPIRED_FEATURE_METADATA) != null);
    }

    public static boolean licenseNeedsExtended(License license) {
        return "basic".equals(license.type()) && license.expiryDate() !=
LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS;
    }

    /**
     * Checks if the signature of a self generated license with older version needs to be
     * recreated with the new key
     */
    public static boolean signatureNeedsUpdate(License license, DiscoveryNodes currentNodes) {
        assert License.VERSION_CRYPTO_ALGORITHMS == License.VERSION_CURRENT : "update this method
when adding a new version";

        return ("basic".equals(license.type()) || "trial".equals(license.type())) &&
                // only upgrade signature when all nodes are ready to deserialize the new signature
                (license.version() < License.VERSION_CRYPTO_ALGORITHMS &&
                    compatibleLicenseVersion(currentNodes) == License.VERSION_CRYPTO_ALGORITHMS
                );
    }

    public static int compatibleLicenseVersion(DiscoveryNodes currentNodes) {
        assert License.VERSION_CRYPTO_ALGORITHMS == License.VERSION_CURRENT : "update this method
when adding a new version";

        if (StreamSupport.stream(currentNodes.spliterator(), false)
            .allMatch(node -> node.getVersion().onOrAfter(Version.V_6_4_0))) {
            // License.VERSION_CRYPTO_ALGORITHMS was introduced in 6.4.0
            return License.VERSION_CRYPTO_ALGORITHMS;
        } else {
            return License.VERSION_START_DATE;
        }
```

```
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.rest.BytesRestResponse;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.RestResponse;
import org.elasticsearch.rest.RestStatus;
import org.elasticsearch.rest.action.RestBuilderListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;

import java.io.IOException;

import static org.elasticsearch.rest.RestRequest.Method.GET;

public class RestGetBasicStatus extends XPackRestHandler {

    RestGetBasicStatus(Settings settings, RestController controller) {
        super(settings);
        controller.registerHandler(GET, URI_BASE + "/license/basic_status", this);
    }

    @Override
    protected RestChannelConsumer doPrepareRequest(RestRequest request, XPackClient client) throws IOException
{
        return channel -> client.licensing().prepareGetStartBasic().execute(
                new RestBuilderListener<GetBasicStatusResponse>(channel) {
                    @Override
                    public RestResponse buildResponse(GetBasicStatusResponse response, XContentBuilder builder)
throws Exception {
                        builder.startObject();
                        builder.field("eligible_to_start_basic", response.isEligibleToStartBasic());
                        builder.endObject();
                        return new BytesRestResponse(RestStatus.OK, builder);
                    }
                });
    }

    @Override
```

```
  public String getName() {
    return "xpack_basic_status_action";
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.action.RestToXContentListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;

import java.io.IOException;

import static org.elasticsearch.rest.RestRequest.Method.DELETE;

public class RestDeleteLicenseAction extends XPackRestHandler {
  public RestDeleteLicenseAction(Settings settings, RestController controller) {
    super(settings);
    // @deprecated Remove deprecations in 6.0
    controller.registerWithDeprecatedHandler(DELETE, URI_BASE + "/license", this,
                                             DELETE, "/_license", deprecationLogger);

    // Remove _licenses support entirely in 6.0
    controller.registerAsDeprecatedHandler(DELETE, "/_licenses", this,
                                           "[DELETE /_licenses] is deprecated! Use " +
                                           "[DELETE /_xpack/license] instead.",
                                           deprecationLogger);
  }

  @Override
  public String getName() {
    return "xpack_delete_license_action";
  }

  @Override
  public RestChannelConsumer doPrepareRequest(final RestRequest request, final XPackClient client) throws
IOException {
    DeleteLicenseRequest deleteLicenseRequest = new DeleteLicenseRequest();
    deleteLicenseRequest.timeout(request.paramAsTime("timeout", deleteLicenseRequest.timeout()));
    deleteLicenseRequest.masterNodeTimeout(request.paramAsTime("master_timeout",
```

```
deleteLicenseRequest.masterNodeTimeout()));

        return channel -> client.es().admin().cluster().execute(DeleteLicenseAction.INSTANCE, deleteLicenseRequest,
                new RestToXContentListener<>(channel));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.rest.BytesRestResponse;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.RestResponse;
import org.elasticsearch.rest.RestStatus;
import org.elasticsearch.rest.action.RestBuilderListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;

import java.io.IOException;

import static org.elasticsearch.rest.RestRequest.Method.GET;

public class RestGetTrialStatus extends XPackRestHandler {

    RestGetTrialStatus(Settings settings, RestController controller) {
        super(settings);
        controller.registerHandler(GET, URI_BASE + "/license/trial_status", this);
    }

    @Override
    protected RestChannelConsumer doPrepareRequest(RestRequest request, XPackClient client) throws IOException
{
        return channel -> client.licensing().prepareGetStartTrial().execute(
                new RestBuilderListener<GetTrialStatusResponse>(channel) {
                    @Override
                    public RestResponse buildResponse(GetTrialStatusResponse response, XContentBuilder builder) throws
Exception {
                        builder.startObject();
                        builder.field("eligible_to_start_trial", response.isEligibleToStartTrial());
                        builder.endObject();
                        return new BytesRestResponse(RestStatus.OK, builder);
                    }
```

```
            });
    }

    @Override
    public String getName() {
        return "xpack_trial_status_action";
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.logging.LoggerMessageFormat;
import org.elasticsearch.common.unit.TimeValue;

import java.util.UUID;

abstract class ExpirationCallback {

    static final String EXPIRATION_JOB_PREFIX = ".license_expiration_job_";

    public enum Orientation {PRE, POST}

    /**
     * Callback that is triggered every <code>frequency</code> when
     * current time is between <code>max</code> and <code>min</code>
     * before license expiry.
     */
    public abstract static class Pre extends ExpirationCallback {

        /**
         * Callback schedule prior to license expiry
         *
         * @param min       latest relative time to execute before license expiry
         * @param max       earliest relative time to execute before license expiry
         * @param frequency interval between execution
         */
        Pre(TimeValue min, TimeValue max, TimeValue frequency) {
            super(Orientation.PRE, min, max, frequency);
        }
    }

    /**
     * Callback that is triggered every <code>frequency</code> when
     * current time is between <code>min</code> and <code>max</code>
```

```
 * after license expiry.
 */
public abstract static class Post extends ExpirationCallback {

    /**
     * Callback schedule after license expiry
     *
     * @param min       earliest relative time to execute after license expiry
     * @param max       latest relative time to execute after license expiry
     * @param frequency interval between execution
     */
    Post(TimeValue min, TimeValue max, TimeValue frequency) {
        super(Orientation.POST, min, max, frequency);
    }
}

private final String id;
private final Orientation orientation;
private final long min;
private final long max;
private final long frequency;

private ExpirationCallback(Orientation orientation, TimeValue min, TimeValue max, TimeValue frequency) {
    this.orientation = orientation;
    this.min = (min == null) ? 0 : min.getMillis();
    this.max = (max == null) ? Long.MAX_VALUE : max.getMillis();
    this.frequency = frequency.getMillis();
    this.id = String.join("", EXPIRATION_JOB_PREFIX, UUID.randomUUID().toString());
}

public final String getId() {
    return id;
}

public final long getFrequency() {
    return frequency;
}

/**
 * Calculates the delay for the next trigger time. When <code>now</code> is in a
 * valid time bracket with respect to <code>expirationDate</code>, the delay is 0.
 * When <code>now</code> is before the time bracket, than delay to the start of the
 * time bracket and when <code>now</code> is passed the valid time bracket, the delay
 * is <code>null</code>
 * @param expirationDate license expiry date in milliseconds
 * @param now current time in milliseconds
 * @return time delay
 */
```

```
final TimeValue delay(long expirationDate, long now) {
    final TimeValue delay;
    switch (orientation) {
        case PRE:
            if (expirationDate >= now) {
                // license not yet expired
                long preExpiryDuration = expirationDate - now;
                if (preExpiryDuration > max) {
                    // license duration is longer than maximum duration, delay it to the first match time
                    delay = TimeValue.timeValueMillis(preExpiryDuration - max);
                } else if (preExpiryDuration <= max && preExpiryDuration >= min) {
                    // no delay in valid time bracket
                    delay = TimeValue.timeValueMillis(0);
                } else {
                    // passed last match time
                    delay = null;
                }
            } else {
                // invalid after license expiry
                delay = null;
            }
            break;
        case POST:
            if (expirationDate >= now) {
                // license not yet expired, delay it to the first match time
                delay = TimeValue.timeValueMillis(expirationDate - now + min);
            } else {
                // license has expired
                long expiredDuration = now - expirationDate;
                if (expiredDuration < min) {
                    // license expiry duration is shorter than minimum duration, delay it to the first match time
                    delay = TimeValue.timeValueMillis(min - expiredDuration);
                } else if (expiredDuration >= min && expiredDuration <= max) {
                    // no delay in valid time bracket
                    delay = TimeValue.timeValueMillis(0);
                } else {
                    // passed last match time
                    delay = null;
                }
            }
            break;
        default:
            throw new IllegalStateException("orientation [" + orientation + "] unknown");
    }
    return delay;
}

/**
```

```
    * {@link org.elasticsearch.xpack.core.scheduler.SchedulerEngine.Schedule#nextScheduledTimeAfter(long,
long)}
    * with respect to license expiry date
    */
   public final long nextScheduledTimeForExpiry(long expiryDate, long startTime, long time) {
       TimeValue delay = delay(expiryDate, time);
       if (delay != null) {
           long delayInMillis = delay.getMillis();
           if (delayInMillis == 0L) {
               if (startTime == time) {
                   // initial trigger and in time bracket, schedule immediately
                   return time;
               } else {
                   // in time bracket, add frequency
                   return time + frequency;
               }
           } else {
               // not in time bracket
               return time + delayInMillis;
           }
       }
       return -1;
   }

   /**
    * Code to execute when the expiry callback is triggered in a valid
    * time bracket
    * @param license license to operate on
    */
   public abstract void on(License license);

   public final String toString() {
       return LoggerMessageFormat.format(null, "ExpirationCallback:(orientation [{}],  min [{}], max [{}], freq
[{}])",
               orientation.name(), TimeValue.timeValueMillis(min), TimeValue.timeValueMillis(max),
               TimeValue.timeValueMillis(frequency));
   }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.joda.FormatDateTimeFormatter;
import org.elasticsearch.common.joda.Joda;
import org.joda.time.MutableDateTime;
```

```
import org.joda.time.format.DateTimeFormatter;
import org.joda.time.format.ISODateTimeFormat;

public class DateUtils {

  private static final FormatDateTimeFormatter formatDateOnlyFormatter = Joda.forPattern("yyyy-MM-dd");

  private static final DateTimeFormatter dateOnlyFormatter = formatDateOnlyFormatter.parser().withZoneUTC();

  private static final DateTimeFormatter dateTimeFormatter = ISODateTimeFormat.dateTime().withZoneUTC();

  public static long endOfTheDay(String date) {
    try {
      // Try parsing using complete date/time format
      return dateTimeFormatter.parseDateTime(date).getMillis();
    } catch (IllegalArgumentException ex) {
      // Fall back to the date only format
      MutableDateTime dateTime = dateOnlyFormatter.parseMutableDateTime(date);
      dateTime.millisOfDay().set(dateTime.millisOfDay().getMaximumValue());
      return dateTime.getMillis();
    }
  }

  public static long beginningOfTheDay(String date) {
    try {
      // Try parsing using complete date/time format
      return dateTimeFormatter.parseDateTime(date).getMillis();
    } catch (IllegalArgumentException ex) {
      // Fall back to the date only format
      return dateOnlyFormatter.parseDateTime(date).getMillis();
    }

  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
import org.elasticsearch.action.support.master.TransportMasterNodeAction;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
```

```
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.transport.TransportService;

public class TransportPostStartTrialAction extends TransportMasterNodeAction<PostStartTrialRequest,
PostStartTrialResponse> {

  private final LicenseService licenseService;

  @Inject
  public TransportPostStartTrialAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                       LicenseService licenseService, ThreadPool threadPool, ActionFilters actionFilters,
                                       IndexNameExpressionResolver indexNameExpressionResolver) {
    super(settings, PostStartTrialAction.NAME, transportService, clusterService, threadPool, actionFilters,
          indexNameExpressionResolver, PostStartTrialRequest::new);
    this.licenseService = licenseService;
  }

  @Override
  protected String executor() {
    return ThreadPool.Names.SAME;
  }

  @Override
  protected PostStartTrialResponse newResponse() {
    return new PostStartTrialResponse();
  }

  @Override
  protected void masterOperation(PostStartTrialRequest request, ClusterState state,
                                 ActionListener<PostStartTrialResponse> listener) throws Exception {
    licenseService.startTrialLicense(request, listener);
  }

  @Override
  protected ClusterBlockException checkBlock(PostStartTrialRequest request, ClusterState state) {
    return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_WRITE);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import org.elasticsearch.action.ActionResponse;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;

import java.io.IOException;

public class GetLicenseResponse extends ActionResponse {

  private License license;

  GetLicenseResponse() {
  }

  GetLicenseResponse(License license) {
    this.license = license;
  }

  public License license() {
    return license;
  }

  @Override
  public void readFrom(StreamInput in) throws IOException {
    super.readFrom(in);
    if (in.readBoolean()) {
      license = License.readLicense(in);
    }
  }

  @Override
  public void writeTo(StreamOutput out) throws IOException {
    super.writeTo(out);
    if (license == null) {
      out.writeBoolean(false);
    } else {
      out.writeBoolean(true);
      license.writeTo(out);
    }
  }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import org.elasticsearch.action.ActionResponse;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;

import java.io.IOException;

class GetBasicStatusResponse extends ActionResponse {

  private boolean eligibleToStartBasic;

  GetBasicStatusResponse() {
  }

  GetBasicStatusResponse(boolean eligibleToStartBasic) {
    this.eligibleToStartBasic = eligibleToStartBasic;
  }

  boolean isEligibleToStartBasic() {
    return eligibleToStartBasic;
  }

  @Override
  public void readFrom(StreamInput in) throws IOException {
    eligibleToStartBasic = in.readBoolean();
  }

  @Override
  public void writeTo(StreamOutput out) throws IOException {
    out.writeBoolean(eligibleToStartBasic);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestBuilder;
import org.elasticsearch.client.ElasticsearchClient;

class GetTrialStatusRequestBuilder extends ActionRequestBuilder<GetTrialStatusRequest,
    GetTrialStatusResponse, GetTrialStatusRequestBuilder> {

  GetTrialStatusRequestBuilder(ElasticsearchClient client, GetTrialStatusAction action) {
    super(client, action, new GetTrialStatusRequest());
  }
```

```
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionResponse;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;

import java.io.IOException;

class GetTrialStatusResponse extends ActionResponse {

  private boolean eligibleToStartTrial;

  GetTrialStatusResponse() {
  }

  GetTrialStatusResponse(boolean eligibleToStartTrial) {
     this.eligibleToStartTrial = eligibleToStartTrial;
  }

  boolean isEligibleToStartTrial() {
     return eligibleToStartTrial;
  }

  @Override
  public void readFrom(StreamInput in) throws IOException {
     eligibleToStartTrial = in.readBoolean();
  }

  @Override
  public void writeTo(StreamOutput out) throws IOException {
     out.writeBoolean(eligibleToStartTrial);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
```

```
import org.elasticsearch.action.support.master.TransportMasterNodeReadAction;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.transport.TransportService;

public class TransportGetBasicStatusAction extends TransportMasterNodeReadAction<GetBasicStatusRequest,
GetBasicStatusResponse> {

  @Inject
  public TransportGetBasicStatusAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                       ThreadPool threadPool, ActionFilters actionFilters,
                                       IndexNameExpressionResolver indexNameExpressionResolver) {
    super(settings, GetBasicStatusAction.NAME, transportService, clusterService, threadPool, actionFilters,
          indexNameExpressionResolver, GetBasicStatusRequest::new);
  }

  @Override
  protected String executor() {
    return ThreadPool.Names.SAME;
  }

  @Override
  protected GetBasicStatusResponse newResponse() {
    return new GetBasicStatusResponse();
  }

  @Override
  protected void masterOperation(GetBasicStatusRequest request, ClusterState state,
                                 ActionListener<GetBasicStatusResponse> listener) throws Exception {
    LicensesMetaData licensesMetaData = state.metaData().custom(LicensesMetaData.TYPE);
    if (licensesMetaData == null) {
      listener.onResponse(new GetBasicStatusResponse(true));
    } else {
      License license = licensesMetaData.getLicense();
      listener.onResponse(new GetBasicStatusResponse(license == null || license.type().equals("basic") == false));
    }

  }

  @Override
  protected ClusterBlockException checkBlock(GetBasicStatusRequest request, ClusterState state) {
```

```
        return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_READ);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;

public class GetTrialStatusAction extends Action<GetTrialStatusRequest, GetTrialStatusResponse,
GetTrialStatusRequestBuilder> {

    public static final GetTrialStatusAction INSTANCE = new GetTrialStatusAction();
    public static final String NAME = "cluster:admin/xpack/license/trial_status";

    private GetTrialStatusAction() {
        super(NAME);
    }

    @Override
    public GetTrialStatusRequestBuilder newRequestBuilder(ElasticsearchClient client) {
        return new GetTrialStatusRequestBuilder(client, this);
    }

    @Override
    public GetTrialStatusResponse newResponse() {
        return new GetTrialStatusResponse();
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.support.master.MasterNodeReadOperationRequestBuilder;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.protocol.xpack.license.GetLicenseRequest;

public class GetLicenseRequestBuilder extends MasterNodeReadOperationRequestBuilder<GetLicenseRequest,
GetLicenseResponse,
        GetLicenseRequestBuilder> {
```

```
  public GetLicenseRequestBuilder(ElasticsearchClient client) {
      this(client, GetLicenseAction.INSTANCE);
  }

  /**
   * Creates new get licenses request builder
   *
   * @param client elasticsearch client
   */
  public GetLicenseRequestBuilder(ElasticsearchClient client, GetLicenseAction action) {
      super(client, action, new GetLicenseRequest());
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestBuilder;
import org.elasticsearch.client.ElasticsearchClient;

class GetBasicStatusRequestBuilder extends ActionRequestBuilder<GetBasicStatusRequest,
GetBasicStatusResponse,
      GetBasicStatusRequestBuilder> {

  GetBasicStatusRequestBuilder(ElasticsearchClient client, GetBasicStatusAction action) {
      super(client, action, new GetBasicStatusRequest());
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.AcknowledgedRequest;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;

import java.io.IOException;

public class PostStartBasicRequest extends AcknowledgedRequest<PostStartBasicRequest> {

  private boolean acknowledge = false;
```

```
    @Override
    public ActionRequestValidationException validate() {
        return null;
    }

    public PostStartBasicRequest acknowledge(boolean acknowledge) {
        this.acknowledge = acknowledge;
        return this;
    }

    public boolean isAcknowledged() {
        return acknowledge;
    }

    @Override
    public void readFrom(StreamInput in) throws IOException {
        super.readFrom(in);
        acknowledge = in.readBoolean();
    }

    @Override
    public void writeTo(StreamOutput out) throws IOException {
        super.writeTo(out);
        out.writeBoolean(acknowledge);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;

public class GetBasicStatusAction extends Action<GetBasicStatusRequest, GetBasicStatusResponse,
GetBasicStatusRequestBuilder> {

    public static final GetBasicStatusAction INSTANCE = new GetBasicStatusAction();
    public static final String NAME = "cluster:admin/xpack/license/basic_status";

    private GetBasicStatusAction() {
        super(NAME);
    }

    @Override
```

```
  public GetBasicStatusRequestBuilder newRequestBuilder(ElasticsearchClient client) {
    return new GetBasicStatusRequestBuilder(client, this);
  }

  @Override
  public GetBasicStatusResponse newResponse() {
    return new GetBasicStatusResponse();
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
import org.elasticsearch.action.support.master.TransportMasterNodeAction;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.transport.TransportService;

public class TransportPutLicenseAction extends TransportMasterNodeAction<PutLicenseRequest,
PutLicenseResponse> {

  private final LicenseService licenseService;

  @Inject
  public TransportPutLicenseAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                   LicenseService licenseService, ThreadPool threadPool, ActionFilters actionFilters,
                                   IndexNameExpressionResolver indexNameExpressionResolver) {
    super(settings, PutLicenseAction.NAME, transportService, clusterService, threadPool, actionFilters,
indexNameExpressionResolver,
          PutLicenseRequest::new);
    this.licenseService = licenseService;
  }

  @Override
```

```
  protected String executor() {
    return ThreadPool.Names.MANAGEMENT;
  }

  @Override
  protected PutLicenseResponse newResponse() {
    return new PutLicenseResponse();
  }

  @Override
  protected ClusterBlockException checkBlock(PutLicenseRequest request, ClusterState state) {
    return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_WRITE);
  }

  @Override
  protected void masterOperation(final PutLicenseRequest request, ClusterState state, final
ActionListener<PutLicenseResponse>
        listener) throws ElasticsearchException {
    licenseService.registerLicense(request, listener);
  }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.Version;
import org.elasticsearch.action.ActionResponse;
import org.elasticsearch.common.io.stream.StreamInput;
import org.elasticsearch.common.io.stream.StreamOutput;
import org.elasticsearch.rest.RestStatus;

import java.io.IOException;
import java.util.Collections;
import java.util.HashMap;
import java.util.Map;

class PostStartTrialResponse extends ActionResponse {

  // Nodes Prior to 6.3 did not have NEED_ACKNOWLEDGEMENT as part of status
  enum Pre63Status {
    UPGRADED_TO_TRIAL,
    TRIAL_ALREADY_ACTIVATED;
  }
  enum Status {
```

```
        UPGRADED_TO_TRIAL(true, null, RestStatus.OK),
        TRIAL_ALREADY_ACTIVATED(false, "Operation failed: Trial was already activated.",
RestStatus.FORBIDDEN),
        NEED_ACKNOWLEDGEMENT(false,"Operation failed: Needs acknowledgement.", RestStatus.OK);

        private final boolean isTrialStarted;

        private final String errorMessage;
        private final RestStatus restStatus;
        Status(boolean isTrialStarted, String errorMessage, RestStatus restStatus) {
            this.isTrialStarted = isTrialStarted;
            this.errorMessage = errorMessage;
            this.restStatus = restStatus;
        }

        boolean isTrialStarted() {
            return isTrialStarted;
        }

        String getErrorMessage() {
            return errorMessage;
        }

        RestStatus getRestStatus() {
            return restStatus;
        }

    }

    private Status status;
    private Map<String, String[]> acknowledgeMessages;
    private String acknowledgeMessage;

    PostStartTrialResponse() {
    }

    PostStartTrialResponse(Status status) {
        this(status, Collections.emptyMap(), null);
    }

    PostStartTrialResponse(Status status, Map<String, String[]> acknowledgeMessages, String acknowledgeMessage)
{
        this.status = status;
        this.acknowledgeMessages = acknowledgeMessages;
        this.acknowledgeMessage = acknowledgeMessage;
    }

    public Status getStatus() {
```

```
        return status;
    }

    @Override
    public void readFrom(StreamInput in) throws IOException {
        status = in.readEnum(Status.class);
        if (in.getVersion().onOrAfter(Version.V_6_3_0)) {
            acknowledgeMessage = in.readOptionalString();
            int size = in.readVInt();
            Map<String, String[]> acknowledgeMessages = new HashMap<>(size);
            for (int i = 0; i < size; i++) {
                String feature = in.readString();
                int nMessages = in.readVInt();
                String[] messages = new String[nMessages];
                for (int j = 0; j < nMessages; j++) {
                    messages[j] = in.readString();
                }
                acknowledgeMessages.put(feature, messages);
            }
            this.acknowledgeMessages = acknowledgeMessages;
        } else {
            this.acknowledgeMessages = Collections.emptyMap();
        }
    }

    @Override
    public void writeTo(StreamOutput out) throws IOException {
        Version version = Version.V_6_3_0;
        if (out.getVersion().onOrAfter(version)) {
            out.writeEnum(status);
            out.writeOptionalString(acknowledgeMessage);
            out.writeVInt(acknowledgeMessages.size());
            for (Map.Entry<String, String[]> entry : acknowledgeMessages.entrySet()) {
                out.writeString(entry.getKey());
                out.writeVInt(entry.getValue().length);
                for (String message : entry.getValue()) {
                    out.writeString(message);
                }
            }
        } else {
            if (status == Status.UPGRADED_TO_TRIAL) {
                out.writeEnum(Pre63Status.UPGRADED_TO_TRIAL);
            } else if (status == Status.TRIAL_ALREADY_ACTIVATED) {
                out.writeEnum(Pre63Status.TRIAL_ALREADY_ACTIVATED);
            } else {
                throw new IllegalArgumentException("Starting trial on node with version [" + Version.CURRENT + "]
requires " +
                        "acknowledgement parameter.");
```

```
        }
      }
    }

    Map<String, String[]> getAcknowledgementMessages() {
      return acknowledgeMessages;
    }

    String getAcknowledgementMessage() {
      return acknowledgeMessage;
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.apache.lucene.util.BytesRef;
import org.apache.lucene.util.BytesRefIterator;
import org.elasticsearch.common.bytes.BytesReference;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.common.xcontent.XContentFactory;
import org.elasticsearch.common.xcontent.XContentType;
import org.elasticsearch.core.internal.io.Streams;

import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.nio.ByteBuffer;
import java.security.InvalidKeyException;
import java.security.NoSuchAlgorithmException;
import java.security.Signature;
import java.security.SignatureException;
import java.util.Arrays;
import java.util.Base64;
import java.util.Collections;

/**
* Responsible for verifying signed licenses
*/
public class LicenseVerifier {

    /**
     * verifies the license content with the signature using the packaged
     * public key
```

```
 * @param license to verify
 * @return true if valid, false otherwise
 */
public static boolean verifyLicense(final License license, byte[] publicKeyData) {
    byte[] signedContent = null;
    byte[] publicKeyFingerprint = null;
    try {
        byte[] signatureBytes = Base64.getDecoder().decode(license.signature());
        ByteBuffer byteBuffer = ByteBuffer.wrap(signatureBytes);
        int version = byteBuffer.getInt();
        int magicLen = byteBuffer.getInt();
        byte[] magic = new byte[magicLen];
        byteBuffer.get(magic);
        int hashLen = byteBuffer.getInt();
        publicKeyFingerprint = new byte[hashLen];
        byteBuffer.get(publicKeyFingerprint);
        int signedContentLen = byteBuffer.getInt();
        signedContent = new byte[signedContentLen];
        byteBuffer.get(signedContent);
        XContentBuilder contentBuilder = XContentFactory.contentBuilder(XContentType.JSON);
        license.toXContent(contentBuilder, new
ToXContent.MapParams(Collections.singletonMap(License.LICENSE_SPEC_VIEW_MODE, "true")));
        Signature rsa = Signature.getInstance("SHA512withRSA");
        rsa.initVerify(CryptUtils.readPublicKey(publicKeyData));
        BytesRefIterator iterator = BytesReference.bytes(contentBuilder).iterator();
        BytesRef ref;
        while((ref = iterator.next()) != null) {
            rsa.update(ref.bytes, ref.offset, ref.length);
        }
        return rsa.verify(signedContent);
    } catch (IOException | NoSuchAlgorithmException | SignatureException | InvalidKeyException e) {
        throw new IllegalStateException(e);
    } finally {
        if (signedContent != null) {
            Arrays.fill(signedContent, (byte) 0);
        }
    }
}

public static boolean verifyLicense(final License license) {
    final byte[] publicKeyBytes;
    try (InputStream is = LicenseVerifier.class.getResourceAsStream("/public.key")) {
        ByteArrayOutputStream out = new ByteArrayOutputStream();
        Streams.copy(is, out);
        publicKeyBytes = out.toByteArray();
    } catch (IOException ex) {
        throw new IllegalStateException(ex);
    }
```

```
        return verifyLicense(license, publicKeyBytes);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.MasterNodeReadRequest;

public class GetTrialStatusRequest extends MasterNodeReadRequest<GetTrialStatusRequest> {

    public GetTrialStatusRequest() {
    }

    @Override
    public ActionRequestValidationException validate() {
        return null;
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.action.RestToXContentListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;

import java.io.IOException;

import static org.elasticsearch.rest.RestRequest.Method.POST;
import static org.elasticsearch.rest.RestRequest.Method.PUT;

public class RestPutLicenseAction extends XPackRestHandler {
    public RestPutLicenseAction(Settings settings, RestController controller) {
        super(settings);
        // @deprecated Remove deprecations in 6.0
        controller.registerWithDeprecatedHandler(POST, URI_BASE + "/license", this,
                                                 POST, "/_license", deprecationLogger);
```

```
        controller.registerWithDeprecatedHandler(PUT, URI_BASE + "/license", this,
                                                 PUT, "/_license", deprecationLogger);

        // Remove _licenses support entirely in 6.0
        controller.registerAsDeprecatedHandler(POST, "/_licenses", this,
                                               "[POST /_licenses] is deprecated! Use " +
                                               "[POST /_xpack/license] instead.",
                                               deprecationLogger);
        controller.registerAsDeprecatedHandler(PUT, "/_licenses", this,
                                               "[PUT /_licenses] is deprecated! Use " +
                                               "[PUT /_xpack/license] instead.",
                                               deprecationLogger);
    }

    @Override
    public String getName() {
        return "xpack_put_license_action";
    }

    @Override
    public RestChannelConsumer doPrepareRequest(final RestRequest request, final XPackClient client) throws
IOException {
        if (request.hasContent() == false) {
            throw new IllegalArgumentException("The license must be provided in the request body");
        }
        PutLicenseRequest putLicenseRequest = new PutLicenseRequest();
        putLicenseRequest.license(request.content(), request.getXContentType());
        putLicenseRequest.acknowledge(request.paramAsBoolean("acknowledge", false));
        putLicenseRequest.timeout(request.paramAsTime("timeout", putLicenseRequest.timeout()));
        putLicenseRequest.masterNodeTimeout(request.paramAsTime("master_timeout",
putLicenseRequest.masterNodeTimeout()));

        if ("basic".equals(putLicenseRequest.license().type())) {
            throw new IllegalArgumentException("Installing basic licenses is no longer allowed. Use the POST " +
                    "/_xpack/license/start_basic API to install a basic license that does not expire.");
        }

        return channel -> client.es().admin().cluster().execute(PutLicenseAction.INSTANCE, putLicenseRequest,
                new RestToXContentListener<>(channel));
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;
```

```
import org.elasticsearch.action.ActionRequestValidationException;
import org.elasticsearch.action.support.master.MasterNodeReadRequest;

public class GetBasicStatusRequest extends MasterNodeReadRequest<GetBasicStatusRequest> {

  public GetBasicStatusRequest() {
  }

  @Override
  public ActionRequestValidationException validate() {
    return null;
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.master.AcknowledgedResponse;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.protocol.xpack.license.GetLicenseRequest;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;

public class LicensingClient {

  private final ElasticsearchClient client;

  public LicensingClient(ElasticsearchClient client) {
    this.client = client;
  }

  public PutLicenseRequestBuilder preparePutLicense(License license) {
    return new PutLicenseRequestBuilder(client).setLicense(license);
  }

  public void putLicense(PutLicenseRequest request, ActionListener<PutLicenseResponse> listener) {
    client.execute(PutLicenseAction.INSTANCE, request, listener);
  }

  public GetLicenseRequestBuilder prepareGetLicense() {
    return new GetLicenseRequestBuilder(client);
  }

  public void getLicense(GetLicenseRequest request, ActionListener<GetLicenseResponse> listener) {
```

```
        client.execute(GetLicenseAction.INSTANCE, request, listener);
    }

    public DeleteLicenseRequestBuilder prepareDeleteLicense() {
        return new DeleteLicenseRequestBuilder(client);
    }

    public void deleteLicense(DeleteLicenseRequest request, ActionListener<AcknowledgedResponse> listener) {
        client.execute(DeleteLicenseAction.INSTANCE, request, listener);
    }

    public PostStartTrialRequestBuilder preparePostStartTrial() {
        return new PostStartTrialRequestBuilder(client, PostStartTrialAction.INSTANCE);
    }

    public GetTrialStatusRequestBuilder prepareGetStartTrial() {
        return new GetTrialStatusRequestBuilder(client, GetTrialStatusAction.INSTANCE);
    }

    public void postStartTrial(PostStartTrialRequest request, ActionListener<PostStartTrialResponse> listener) {
        client.execute(PostStartTrialAction.INSTANCE, request, listener);
    }

    public void postStartBasic(PostStartBasicRequest request, ActionListener<PostStartBasicResponse> listener) {
        client.execute(PostStartBasicAction.INSTANCE, request, listener);
    }

    public PostStartBasicRequestBuilder preparePostStartBasic() {
        return new PostStartBasicRequestBuilder(client, PostStartBasicAction.INSTANCE);
    }

    public GetBasicStatusRequestBuilder prepareGetStartBasic() {
        return new GetBasicStatusRequestBuilder(client, GetBasicStatusAction.INSTANCE);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ContextPreservingActionListener;
import org.elasticsearch.client.Client;
import org.elasticsearch.common.util.concurrent.ThreadContext;
```

```
import org.elasticsearch.protocol.xpack.XPackInfoRequest;
import org.elasticsearch.protocol.xpack.XPackInfoResponse;
import org.elasticsearch.protocol.xpack.license.LicenseStatus;
import org.elasticsearch.transport.RemoteClusterAware;
import org.elasticsearch.xpack.core.action.XPackInfoAction;

import java.util.EnumSet;
import java.util.Iterator;
import java.util.List;
import java.util.Locale;
import java.util.concurrent.atomic.AtomicReference;
import java.util.function.Predicate;
import java.util.stream.Collectors;

/**
* Checks remote clusters for license compatibility with a specified license predicate.
*/
public final class RemoteClusterLicenseChecker {

  /**
   * Encapsulates the license info of a remote cluster.
   */
  public static final class RemoteClusterLicenseInfo {

    private final String clusterAlias;

    /**
     * The alias of the remote cluster.
     *
     * @return the cluster alias
     */
    public String clusterAlias() {
      return clusterAlias;
    }

    private final XPackInfoResponse.LicenseInfo licenseInfo;

    /**
     * The license info of the remote cluster.
     *
     * @return the license info
     */
    public XPackInfoResponse.LicenseInfo licenseInfo() {
      return licenseInfo;
    }

    RemoteClusterLicenseInfo(final String clusterAlias, final XPackInfoResponse.LicenseInfo licenseInfo) {
      this.clusterAlias = clusterAlias;
```

```
        this.licenseInfo = licenseInfo;
    }

}

/**
 * Encapsulates a remote cluster license check. The check is either successful if the license of the remote cluster is compatible with
 * the predicate used to check license compatibility, or the check is a failure.
 */
public static final class LicenseCheck {

    private final RemoteClusterLicenseInfo remoteClusterLicenseInfo;

    /**
     * The remote cluster license info. This method should only be invoked if this instance represents a failing license check.
     *
     * @return the remote cluster license info
     */
    public RemoteClusterLicenseInfo remoteClusterLicenseInfo() {
        assert isSuccess() == false;
        return remoteClusterLicenseInfo;
    }

    private static final LicenseCheck SUCCESS = new LicenseCheck(null);

    /**
     * A successful license check.
     *
     * @return a successful license check instance
     */
    public static LicenseCheck success() {
        return SUCCESS;
    }

    /**
     * Test if this instance represents a successful license check.
     *
     * @return true if this instance represents a successful license check, otherwise false
     */
    public boolean isSuccess() {
        return this == SUCCESS;
    }

    /**
     * Creates a failing license check encapsulating the specified remote cluster license info.
     *
```

```
     * @param remoteClusterLicenseInfo the remote cluster license info
     * @return a failing license check
     */
    public static LicenseCheck failure(final RemoteClusterLicenseInfo remoteClusterLicenseInfo) {
        return new LicenseCheck(remoteClusterLicenseInfo);
    }

    private LicenseCheck(final RemoteClusterLicenseInfo remoteClusterLicenseInfo) {
        this.remoteClusterLicenseInfo = remoteClusterLicenseInfo;
    }

}

private final Client client;
private final Predicate<License.OperationMode> predicate;

/**
 * Constructs a remote cluster license checker with the specified license predicate for checking license
compatibility. The predicate
 * does not need to check for the active license state as this is handled by the remote cluster license checker.
 *
 * @param client    the client
 * @param predicate the license predicate
 */
public RemoteClusterLicenseChecker(final Client client, final Predicate<License.OperationMode> predicate) {
    this.client = client;
    this.predicate = predicate;
}

public static boolean isLicensePlatinumOrTrial(final XPackInfoResponse.LicenseInfo licenseInfo) {
    final License.OperationMode mode = License.OperationMode.resolve(licenseInfo.getMode());
    return mode == License.OperationMode.PLATINUM || mode == License.OperationMode.TRIAL;
}

/**
 * Checks the specified clusters for license compatibility. The specified callback will be invoked once if all
clusters are
 * license-compatible, otherwise the specified callback will be invoked once on the first cluster that is not license-
compatible.
 *
 * @param clusterAliases the cluster aliases to check
 * @param listener       a callback
 */
public void checkRemoteClusterLicenses(final List<String> clusterAliases, final ActionListener<LicenseCheck>
listener) {
    final Iterator<String> clusterAliasesIterator = clusterAliases.iterator();
    if (clusterAliasesIterator.hasNext() == false) {
        listener.onResponse(LicenseCheck.success());
```

```
            return;
        }

        final AtomicReference<String> clusterAlias = new AtomicReference<>();

        final ActionListener<XPackInfoResponse> infoListener = new ActionListener<XPackInfoResponse>() {

            @Override
            public void onResponse(final XPackInfoResponse xPackInfoResponse) {
                final XPackInfoResponse.LicenseInfo licenseInfo = xPackInfoResponse.getLicenseInfo();
                if ((licenseInfo.getStatus() == LicenseStatus.ACTIVE) == false
                        || predicate.test(License.OperationMode.resolve(licenseInfo.getMode())) == false) {
                    listener.onResponse(LicenseCheck.failure(new RemoteClusterLicenseInfo(clusterAlias.get(),
licenseInfo)));
                    return;
                }

                if (clusterAliasesIterator.hasNext()) {
                    clusterAlias.set(clusterAliasesIterator.next());
                    // recurse to the next cluster
                    remoteClusterLicense(clusterAlias.get(), this);
                } else {
                    listener.onResponse(LicenseCheck.success());
                }
            }

            @Override
            public void onFailure(final Exception e) {
                final String message = "could not determine the license type for cluster [" + clusterAlias.get() + "]";
                listener.onFailure(new ElasticsearchException(message, e));
            }

        };

        // check the license on the first cluster, and then we recursively check licenses on the remaining clusters
        clusterAlias.set(clusterAliasesIterator.next());
        remoteClusterLicense(clusterAlias.get(), infoListener);
    }

    private void remoteClusterLicense(final String clusterAlias, final ActionListener<XPackInfoResponse> listener) {
        final ThreadContext threadContext = client.threadPool().getThreadContext();
        final ContextPreservingActionListener<XPackInfoResponse> contextPreservingActionListener =
                new ContextPreservingActionListener<>(threadContext.newRestorableContext(false), listener);
        try (ThreadContext.StoredContext ignore = threadContext.stashContext()) {
            // we stash any context here since this is an internal execution and should not leak any existing context
information
            threadContext.markAsSystemContext();
```

```
            final XPackInfoRequest request = new XPackInfoRequest();
            request.setCategories(EnumSet.of(XPackInfoRequest.Category.LICENSE));
            try {
                client.getRemoteClusterClient(clusterAlias).execute(XPackInfoAction.INSTANCE, request,
contextPreservingActionListener);
            } catch (final Exception e) {
                contextPreservingActionListener.onFailure(e);
            }
        }
    }

    /**
     * Predicate to test if the index name represents the name of a remote index.
     *
     * @param index the index name
     * @return true if the collection of indices contains a remote index, otherwise false
     */
    public static boolean isRemoteIndex(final String index) {
        return index.indexOf(RemoteClusterAware.REMOTE_CLUSTER_INDEX_SEPARATOR) != -1;
    }

    /**
     * Predicate to test if the collection of index names contains any that represent the name of a remote index.
     *
     * @param indices the collection of index names
     * @return true if the collection of index names contains a name that represents a remote index, otherwise false
     */
    public static boolean containsRemoteIndex(final List<String> indices) {
        return indices.stream().anyMatch(RemoteClusterLicenseChecker::isRemoteIndex);
    }

    /**
     * Filters the collection of index names for names that represent a remote index. Remote index names are of the
form
     * {@code cluster_name:index_name}.
     *
     * @param indices the collection of index names
     * @return list of index names that represent remote index names
     */
    public static List<String> remoteIndices(final List<String> indices) {
        return indices.stream().filter(RemoteClusterLicenseChecker::isRemoteIndex).collect(Collectors.toList());
    }

    /**
     * Extract the list of remote cluster aliases from the list of index names. Remote index names are of the form
     * {@code cluster_alias:index_name} and the cluster_alias is extracted for each index name that represents a
remote index.
     *
```

```
    * @param indices the collection of index names
    * @return the remote cluster names
    */
   public static List<String> remoteClusterAliases(final List<String> indices) {
      return indices.stream()
            .filter(RemoteClusterLicenseChecker::isRemoteIndex)
            .map(index -> index.substring(0,
index.indexOf(RemoteClusterAware.REMOTE_CLUSTER_INDEX_SEPARATOR)))
            .distinct()
            .collect(Collectors.toList());
   }

   /**
    * Constructs an error message for license incompatibility.
    *
    * @param feature                the name of the feature that initiated the remote cluster license check.
    * @param remoteClusterLicenseInfo the remote cluster license info of the cluster that failed the license check
    * @return an error message representing license incompatibility
    */
   public static String buildErrorMessage(
         final String feature,
         final RemoteClusterLicenseInfo remoteClusterLicenseInfo,
         final Predicate<XPackInfoResponse.LicenseInfo> predicate) {
      final StringBuilder error = new StringBuilder();
      if (remoteClusterLicenseInfo.licenseInfo().getStatus() != LicenseStatus.ACTIVE) {
         error.append(String.format(Locale.ROOT, "the license on cluster [%s] is not active",
remoteClusterLicenseInfo.clusterAlias()));
      } else {
         assert predicate.test(remoteClusterLicenseInfo.licenseInfo()) == false : "license must be incompatible to build
error message";
         final String message = String.format(
               Locale.ROOT,
               "the license mode [%s] on cluster [%s] does not enable [%s]",
               License.OperationMode.resolve(remoteClusterLicenseInfo.licenseInfo().getMode()),
               remoteClusterLicenseInfo.clusterAlias(),
               feature);
         error.append(message);
      }

      return error.toString();
   }

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
package org.elasticsearch.license;

import org.elasticsearch.Version;

/**
* Marker interface for callbacks that are invoked when the license state changes.
*/
@FunctionalInterface
public interface LicenseStateListener {

   /**
    * Callback when the license state changes. See {@link XPackLicenseState#update(License.OperationMode,
boolean, Version)}.
    */
   void licenseStateChanged();

}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;

public class PutLicenseAction extends Action<PutLicenseRequest, PutLicenseResponse,
PutLicenseRequestBuilder> {

   public static final PutLicenseAction INSTANCE = new PutLicenseAction();
   public static final String NAME = "cluster:admin/xpack/license/put";

   private PutLicenseAction() {
      super(NAME);
   }

   @Override
   public PutLicenseResponse newResponse() {
      return new PutLicenseResponse();
   }

   @Override
   public PutLicenseRequestBuilder newRequestBuilder(ElasticsearchClient client) {
      return new PutLicenseRequestBuilder(client, this);
   }
```

```
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.apache.logging.log4j.Logger;
import org.apache.logging.log4j.message.ParameterizedMessage;
import org.elasticsearch.Version;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.xpack.core.XPackPlugin;

import java.time.Clock;
import java.util.Collections;
import java.util.Map;
import java.util.UUID;

public class StartTrialClusterTask extends ClusterStateUpdateTask {

  private static final String ACKNOWLEDGEMENT_HEADER = "This API initiates a free 30-day trial for all
platinum features. " +
        "By starting this trial, you agree that it is subject to the terms and conditions at" +
        " https://www.elastic.co/legal/trial_license/. To begin your free trial, call /start_trial again and specify " +
        "the \"acknowledge=true\" parameter.";

  private static final Map<String, String[]> ACK_MESSAGES = Collections.singletonMap("security",
        new String[] {"With a trial license, X-Pack security features are available, but are not enabled by default."});

  private final Logger logger;
  private final String clusterName;
  private final PostStartTrialRequest request;
  private final ActionListener<PostStartTrialResponse> listener;
  private final Clock clock;

  StartTrialClusterTask(Logger logger, String clusterName, Clock clock, PostStartTrialRequest request,
                        ActionListener<PostStartTrialResponse> listener) {
    this.logger = logger;
    this.clusterName = clusterName;
    this.request = request;
    this.listener = listener;
    this.clock = clock;
  }
```

```
  @Override
  public void clusterStateProcessed(String source, ClusterState oldState, ClusterState newState) {
    LicensesMetaData oldLicensesMetaData = oldState.metaData().custom(LicensesMetaData.TYPE);
    logger.debug("started self generated trial license: {}", oldLicensesMetaData);

    if (request.isAcknowledged() == false) {
      listener.onResponse(new
PostStartTrialResponse(PostStartTrialResponse.Status.NEED_ACKNOWLEDGEMENT,
            ACK_MESSAGES, ACKNOWLEDGEMENT_HEADER));
    } else if (oldLicensesMetaData == null || oldLicensesMetaData.isEligibleForTrial()) {
      listener.onResponse(new PostStartTrialResponse(PostStartTrialResponse.Status.UPGRADED_TO_TRIAL));
    } else {
      listener.onResponse(new
PostStartTrialResponse(PostStartTrialResponse.Status.TRIAL_ALREADY_ACTIVATED));
    }
  }

  @Override
  public ClusterState execute(ClusterState currentState) throws Exception {
    XPackPlugin.checkReadyForXPackCustomMetadata(currentState);
    LicensesMetaData currentLicensesMetaData = currentState.metaData().custom(LicensesMetaData.TYPE);

    if (request.isAcknowledged() == false) {
      return currentState;
    } else if (currentLicensesMetaData == null || currentLicensesMetaData.isEligibleForTrial()) {
      long issueDate = clock.millis();
      MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
      long expiryDate = issueDate +
LicenseService.NON_BASIC_SELF_GENERATED_LICENSE_DURATION.getMillis();

      License.Builder specBuilder = License.builder()
            .uid(UUID.randomUUID().toString())
            .issuedTo(clusterName)
            .maxNodes(LicenseService.SELF_GENERATED_LICENSE_MAX_NODES)
            .issueDate(issueDate)
            .type(request.getType())
            .expiryDate(expiryDate);
      License selfGeneratedLicense = SelfGeneratedLicense.create(specBuilder, currentState.nodes());
      LicensesMetaData newLicensesMetaData = new LicensesMetaData(selfGeneratedLicense,
Version.CURRENT);
      mdBuilder.putCustom(LicensesMetaData.TYPE, newLicensesMetaData);
      return ClusterState.builder(currentState).metaData(mdBuilder).build();
    } else {
      return currentState;
    }
  }
```

```
  @Override
  public void onFailure(String source, @Nullable Exception e) {
      logger.error(new ParameterizedMessage("unexpected failure during [{}]", source), e);
      listener.onFailure(e);
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionRequestBuilder;
import org.elasticsearch.client.ElasticsearchClient;

class PostStartTrialRequestBuilder extends ActionRequestBuilder<PostStartTrialRequest,
      PostStartTrialResponse, PostStartTrialRequestBuilder> {

  PostStartTrialRequestBuilder(ElasticsearchClient client, PostStartTrialAction action) {
      super(client, action, new PostStartTrialRequest());
  }

  public PostStartTrialRequestBuilder setAcknowledge(boolean acknowledge) {
      request.acknowledge(acknowledge);
      return this;
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
import org.elasticsearch.action.support.master.TransportMasterNodeReadAction;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.protocol.xpack.license.GetLicenseRequest;
import org.elasticsearch.threadpool.ThreadPool;
```

```
import org.elasticsearch.transport.TransportService;

public class TransportGetLicenseAction extends TransportMasterNodeReadAction<GetLicenseRequest,
GetLicenseResponse> {

  private final LicenseService licenseService;

  @Inject
  public TransportGetLicenseAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                   LicenseService licenseService, ThreadPool threadPool, ActionFilters actionFilters,
                                   IndexNameExpressionResolver indexNameExpressionResolver) {
    super(settings, GetLicenseAction.NAME, transportService, clusterService, threadPool, actionFilters,
indexNameExpressionResolver,
          GetLicenseRequest::new);
    this.licenseService = licenseService;
  }

  @Override
  protected String executor() {
    return ThreadPool.Names.MANAGEMENT;
  }

  @Override
  protected GetLicenseResponse newResponse() {
    return new GetLicenseResponse();
  }

  @Override
  protected ClusterBlockException checkBlock(GetLicenseRequest request, ClusterState state) {
    return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_READ);
  }

  @Override
  protected void masterOperation(final GetLicenseRequest request, ClusterState state,
                                 final ActionListener<GetLicenseResponse> listener) throws ElasticsearchException {
    listener.onResponse(new GetLicenseResponse(licenseService.getLicense()));
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;
```

```
public class PostStartTrialAction extends Action<PostStartTrialRequest, PostStartTrialResponse,
PostStartTrialRequestBuilder> {

  public static final PostStartTrialAction INSTANCE = new PostStartTrialAction();
  public static final String NAME = "cluster:admin/xpack/license/start_trial";

  private PostStartTrialAction() {
     super(NAME);
  }

  @Override
  public PostStartTrialRequestBuilder newRequestBuilder(ElasticsearchClient client) {
     return new PostStartTrialRequestBuilder(client, this);
  }

  @Override
  public PostStartTrialResponse newResponse() {
     return new PostStartTrialResponse();
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.Action;
import org.elasticsearch.client.ElasticsearchClient;

public class PostStartBasicAction extends Action<PostStartBasicRequest, PostStartBasicResponse,
PostStartBasicRequestBuilder> {

  public static final PostStartBasicAction INSTANCE = new PostStartBasicAction();
  public static final String NAME = "cluster:admin/xpack/license/start_basic";

  private PostStartBasicAction() {
     super(NAME);
  }

  @Override
  public PostStartBasicRequestBuilder newRequestBuilder(ElasticsearchClient client) {
     return new PostStartBasicRequestBuilder(client, this);
  }

  @Override
  public PostStartBasicResponse newResponse() {
```

```
        return new PostStartBasicResponse();
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.Version;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.cluster.AckedClusterStateUpdateTask;
import org.elasticsearch.cluster.ClusterChangedEvent;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ClusterStateListener;
import org.elasticsearch.cluster.ack.ClusterStateUpdateResponse;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.cluster.node.DiscoveryNode;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.component.AbstractLifecycleComponent;
import org.elasticsearch.common.component.Lifecycle;
import org.elasticsearch.common.joda.FormatDateTimeFormatter;
import org.elasticsearch.common.joda.Joda;
import org.elasticsearch.common.logging.LoggerMessageFormat;
import org.elasticsearch.common.settings.Setting;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.unit.TimeValue;
import org.elasticsearch.discovery.DiscoveryModule;
import org.elasticsearch.env.Environment;
import org.elasticsearch.gateway.GatewayService;
import org.elasticsearch.protocol.xpack.XPackInfoResponse;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.protocol.xpack.license.LicensesStatus;
import org.elasticsearch.protocol.xpack.license.PutLicenseResponse;
import org.elasticsearch.watcher.ResourceWatcherService;
import org.elasticsearch.xpack.core.XPackPlugin;
import org.elasticsearch.xpack.core.XPackSettings;
import org.elasticsearch.xpack.core.scheduler.SchedulerEngine;

import java.time.Clock;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.HashMap;
import java.util.HashSet;
import java.util.List;
```

```
import java.util.Locale;
import java.util.Map;
import java.util.Set;
import java.util.concurrent.atomic.AtomicReference;

/**
* Service responsible for managing {@link LicensesMetaData}.
* <p>
* On the master node, the service handles updating the cluster state when a new license is registered.
* It also listens on all nodes for cluster state updates, and updates {@link XPackLicenseState} when
* the license changes are detected in the cluster state.
*/
public class LicenseService extends AbstractLifecycleComponent implements ClusterStateListener,
SchedulerEngine.Listener {

  public static final Setting<String> SELF_GENERATED_LICENSE_TYPE = new
Setting<>("xpack.license.self_generated.type",
        (s) -> "basic", (s) -> {
      if (SelfGeneratedLicense.validSelfGeneratedType(s)) {
        return s;
      } else {
        throw new IllegalArgumentException("Illegal self generated license type [" + s + "]. Must be trial or basic.");
      }
  }, Setting.Property.NodeScope);

  // pkg private for tests
  static final TimeValue NON_BASIC_SELF_GENERATED_LICENSE_DURATION =
TimeValue.timeValueHours(30 * 24);

  static final Set<String> VALID_TRIAL_TYPES = new HashSet<>(Arrays.asList("trial", "platinum", "gold"));

  /**
   * Duration of grace period after a license has expired
   */
  static final TimeValue GRACE_PERIOD_DURATION = days(7);

  public static final long BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS =
        XPackInfoResponse.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS;

  private final ClusterService clusterService;

  /**
   * The xpack feature state to update when license changes are made.
   */
  private final XPackLicenseState licenseState;

  /**
   * Currently active license
```

```
 */
private final AtomicReference<License> currentLicense = new AtomicReference<>();
private SchedulerEngine scheduler;
private final Clock clock;

/**
 * File watcher for operation mode changes
 */
private final OperationModeFileWatcher operationModeFileWatcher;

/**
 * Callbacks to notify relative to license expiry
 */
private List<ExpirationCallback> expirationCallbacks = new ArrayList<>();

/**
 * Max number of nodes licensed by generated trial license
 */
static final int SELF_GENERATED_LICENSE_MAX_NODES = 1000;

public static final String LICENSE_JOB = "licenseJob";

private static final FormatDateTimeFormatter DATE_FORMATTER = Joda.forPattern("EEEE, MMMMM dd,
yyyy", Locale.ROOT);

private static final String ACKNOWLEDGEMENT_HEADER = "This license update requires acknowledgement.
To acknowledge the license, " +
        "please read the following messages and update the license again, this time with the \"acknowledge=true\"
parameter:";

public LicenseService(Settings settings, ClusterService clusterService, Clock clock, Environment env,
                      ResourceWatcherService resourceWatcherService, XPackLicenseState licenseState) {
    super(settings);
    this.clusterService = clusterService;
    this.clock = clock;
    this.scheduler = new SchedulerEngine(settings, clock);
    this.licenseState = licenseState;
    this.operationModeFileWatcher = new OperationModeFileWatcher(resourceWatcherService,
        XPackPlugin.resolveConfigFile(env, "license_mode"), logger,
        () -> updateLicenseState(getLicensesMetaData()));
    this.scheduler.register(this);
    populateExpirationCallbacks();
}

private void logExpirationWarning(long expirationMillis, boolean expired) {
    String expiredMsg = expired ? "expired" : "will expire";
    String general = LoggerMessageFormat.format(null, "License [{}] on [{}].\n" +
            "# If you have a new license, please update it. Otherwise, please reach out to\n" +
```

```
            "# your support contact.\n" +
            "# ", expiredMsg, DATE_FORMATTER.printer().print(expirationMillis));
    if (expired) {
        general = general.toUpperCase(Locale.ROOT);
    }
    StringBuilder builder = new StringBuilder(general);
    builder.append(System.lineSeparator());
    if (expired) {
        builder.append("# COMMERCIAL PLUGINS OPERATING WITH REDUCED FUNCTIONALITY");
    } else {
        builder.append("# Commercial plugins operate with reduced functionality on license expiration:");
    }
    XPackLicenseState.EXPIRATION_MESSAGES.forEach((feature, messages) -> {
        if (messages.length > 0) {
            builder.append(System.lineSeparator());
            builder.append("# - ");
            builder.append(feature);
            for (String message : messages) {
                builder.append(System.lineSeparator());
                builder.append("#  - ");
                builder.append(message);
            }
        }
    });
    logger.warn("{}", builder);
}

private void populateExpirationCallbacks() {
    expirationCallbacks.add(new ExpirationCallback.Pre(days(7), days(25), days(1)) {
        @Override
        public void on(License license) {
            logExpirationWarning(license.expiryDate(), false);
        }
    });
    expirationCallbacks.add(new ExpirationCallback.Pre(days(0), days(7), TimeValue.timeValueMinutes(10)) {
        @Override
        public void on(License license) {
            logExpirationWarning(license.expiryDate(), false);
        }
    });
    expirationCallbacks.add(new ExpirationCallback.Post(days(0), null, TimeValue.timeValueMinutes(10)) {
        @Override
        public void on(License license) {
            // logged when grace period begins
            logExpirationWarning(license.expiryDate(), true);
        }
    });
}
```

```
  /**
   * Registers new license in the cluster
   * Master only operation. Installs a new license on the master provided it is VALID
   */
  public void registerLicense(final PutLicenseRequest request, final ActionListener<PutLicenseResponse> listener)
{
      final License newLicense = request.license();
      final long now = clock.millis();
      if (!LicenseVerifier.verifyLicense(newLicense) || newLicense.issueDate() > now || newLicense.startDate() >
now) {
         listener.onResponse(new PutLicenseResponse(true, LicensesStatus.INVALID));
      } else if (newLicense.type().equals("basic")) {
         listener.onFailure(new IllegalArgumentException("Registering basic licenses is not allowed."));
      } else if (newLicense.expiryDate() < now) {
         listener.onResponse(new PutLicenseResponse(true, LicensesStatus.EXPIRED));
      } else {
         if (!request.acknowledged()) {
            // TODO: ack messages should be generated on the master, since another node's cluster state may be
behind...
            final License currentLicense = getLicense();
            if (currentLicense != null) {
               Map<String, String[]> acknowledgeMessages = getAckMessages(newLicense, currentLicense);
               if (acknowledgeMessages.isEmpty() == false) {
                  // needs acknowledgement
                  listener.onResponse(new PutLicenseResponse(false, LicensesStatus.VALID,
ACKNOWLEDGEMENT_HEADER,
                        acknowledgeMessages));
                  return;
               }
            }
         }

         if (XPackSettings.SECURITY_ENABLED.get(settings)) {
            // TODO we should really validate that all nodes have xpack installed and are consistently configured but
this
            // should happen on a different level and not in this code
            if (newLicense.isProductionLicense()
               && XPackSettings.TRANSPORT_SSL_ENABLED.get(settings) == false
               && isProductionMode(settings, clusterService.localNode())) {
               // security is on but TLS is not configured we gonna fail the entire request and throw an exception
               throw new IllegalStateException("Cannot install a [" + newLicense.operationMode() +
                  "] license unless TLS is configured or security is disabled");
            } else if (XPackSettings.FIPS_MODE_ENABLED.get(settings)
               && newLicense.operationMode() != License.OperationMode.PLATINUM
               && newLicense.operationMode() != License.OperationMode.TRIAL) {
               throw new IllegalStateException("Cannot install a [" + newLicense.operationMode() +
                  "] license unless FIPS mode is disabled");
```

```
            }
        }

        clusterService.submitStateUpdateTask("register license [" + newLicense.uid() + "]", new
                AckedClusterStateUpdateTask<PutLicenseResponse>(request, listener) {
                    @Override
                    protected PutLicenseResponse newResponse(boolean acknowledged) {
                        return new PutLicenseResponse(acknowledged, LicensesStatus.VALID);
                    }

                    @Override
                    public ClusterState execute(ClusterState currentState) throws Exception {
                        XPackPlugin.checkReadyForXPackCustomMetadata(currentState);
                        MetaData currentMetadata = currentState.metaData();
                        LicensesMetaData licensesMetaData = currentMetadata.custom(LicensesMetaData.TYPE);
                        Version trialVersion = null;
                        if (licensesMetaData != null) {
                            trialVersion = licensesMetaData.getMostRecentTrialVersion();
                        }
                        MetaData.Builder mdBuilder = MetaData.builder(currentMetadata);
                        mdBuilder.putCustom(LicensesMetaData.TYPE, new LicensesMetaData(newLicense,
trialVersion));
                        return ClusterState.builder(currentState).metaData(mdBuilder).build();
                    }
                });
    }
}

public static Map<String, String[]> getAckMessages(License newLicense, License currentLicense) {
    Map<String, String[]> acknowledgeMessages = new HashMap<>();
    if (!License.isAutoGeneratedLicense(currentLicense.signature()) // current license is not auto-generated
            && currentLicense.issueDate() > newLicense.issueDate()) { // and has a later issue date
        acknowledgeMessages.put("license", new String[]{
                "The new license is older than the currently installed license. " +
                        "Are you sure you want to override the current license?"});
    }
    XPackLicenseState.ACKNOWLEDGMENT_MESSAGES.forEach((feature, ackMessages) -> {
        String[] messages = ackMessages.apply(currentLicense.operationMode(), newLicense.operationMode());
        if (messages.length > 0) {
            acknowledgeMessages.put(feature, messages);
        }
    });
    return acknowledgeMessages;
}


private static TimeValue days(int days) {
    return TimeValue.timeValueHours(days * 24);
```

```
    }

    @Override
    public void triggered(SchedulerEngine.Event event) {
        final LicensesMetaData licensesMetaData = getLicensesMetaData();
        if (licensesMetaData != null) {
            final License license = licensesMetaData.getLicense();
            if (event.getJobName().equals(LICENSE_JOB)) {
                updateLicenseState(license, licensesMetaData.getMostRecentTrialVersion());
            } else if (event.getJobName().startsWith(ExpirationCallback.EXPIRATION_JOB_PREFIX)) {
                expirationCallbacks.stream()
                        .filter(expirationCallback -> expirationCallback.getId().equals(event.getJobName()))
                        .forEach(expirationCallback -> expirationCallback.on(license));
            }
        }
    }

    /**
     * Remove license from the cluster state metadata
     */
    public void removeLicense(final DeleteLicenseRequest request, final
ActionListener<ClusterStateUpdateResponse> listener) {
        clusterService.submitStateUpdateTask("delete license",
                new AckedClusterStateUpdateTask<ClusterStateUpdateResponse>(request, listener) {
                    @Override
                    protected ClusterStateUpdateResponse newResponse(boolean acknowledged) {
                        return new ClusterStateUpdateResponse(acknowledged);
                    }

                    @Override
                    public ClusterState execute(ClusterState currentState) throws Exception {
                        MetaData metaData = currentState.metaData();
                        final LicensesMetaData currentLicenses = metaData.custom(LicensesMetaData.TYPE);
                        if (currentLicenses.getLicense() != LicensesMetaData.LICENSE_TOMBSTONE) {
                            MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
                            LicensesMetaData newMetadata = new
LicensesMetaData(LicensesMetaData.LICENSE_TOMBSTONE,
                                    currentLicenses.getMostRecentTrialVersion());
                            mdBuilder.putCustom(LicensesMetaData.TYPE, newMetadata);
                            return ClusterState.builder(currentState).metaData(mdBuilder).build();
                        } else {
                            return currentState;
                        }
                    }
                });
    }

    public License getLicense() {
```

```
        final License license = getLicense(clusterService.state().metaData());
        return license == LicensesMetaData.LICENSE_TOMBSTONE ? null : license;
    }

    private LicensesMetaData getLicensesMetaData() {
        return this.clusterService.state().metaData().custom(LicensesMetaData.TYPE);
    }

    void startTrialLicense(PostStartTrialRequest request, final ActionListener<PostStartTrialResponse> listener) {
        if (VALID_TRIAL_TYPES.contains(request.getType()) == false) {
            throw new IllegalArgumentException("Cannot start trial of type [" + request.getType() + "]. Valid trial types
are "
                    + VALID_TRIAL_TYPES + ".");
        }
        StartTrialClusterTask task = new StartTrialClusterTask(logger, clusterService.getClusterName().value(), clock,
request, listener);
        clusterService.submitStateUpdateTask("started trial license", task);
    }

    void startBasicLicense(PostStartBasicRequest request, final ActionListener<PostStartBasicResponse> listener) {
        StartBasicClusterTask task = new StartBasicClusterTask(logger, clusterService.getClusterName().value(),
clock, request, listener);
        clusterService.submitStateUpdateTask("start basic license", task);
    }

    /**
     * Master-only operation to generate a one-time global self generated license.
     * The self generated license is only generated and stored if the current cluster state metadata
     * has no existing license. If the cluster currently has a basic license that has an expiration date,
     * a new basic license with no expiration date is generated.
     */
    private void registerOrUpdateSelfGeneratedLicense() {
        clusterService.submitStateUpdateTask("maybe generate license for cluster",
                new StartupSelfGeneratedLicenseTask(settings, clock, clusterService));
    }

    @Override
    protected void doStart() throws ElasticsearchException {
        clusterService.addListener(this);
        scheduler.start(Collections.emptyList());
        logger.debug("initializing license state");
        if (clusterService.lifecycleState() == Lifecycle.State.STARTED) {
            final ClusterState clusterState = clusterService.state();
            if (clusterState.blocks().hasGlobalBlock(GatewayService.STATE_NOT_RECOVERED_BLOCK) == false
&&
                    clusterState.nodes().getMasterNode() != null &&
XPackPlugin.isReadyForXPackCustomMetadata(clusterState)) {
                final LicensesMetaData currentMetaData = clusterState.metaData().custom(LicensesMetaData.TYPE);
```

```
            boolean noLicense = currentMetaData == null || currentMetaData.getLicense() == null;
            if (clusterState.getNodes().isLocalNodeElectedMaster() &&
                    (noLicense || LicenseUtils.licenseNeedsExtended(currentMetaData.getLicense()))) {
                // triggers a cluster changed event eventually notifying the current licensee
                registerOrUpdateSelfGeneratedLicense();
            }
        }
    }
}

@Override
protected void doStop() throws ElasticsearchException {
    clusterService.removeListener(this);
    scheduler.stop();
    // clear current license
    currentLicense.set(null);
}

@Override
protected void doClose() throws ElasticsearchException {
}

/**
 * When there is no global block on {@link
org.elasticsearch.gateway.GatewayService#STATE_NOT_RECOVERED_BLOCK}
 * notify licensees and issue auto-generated license if no license has been installed/issued yet.
 */
@Override
public void clusterChanged(ClusterChangedEvent event) {
    final ClusterState previousClusterState = event.previousState();
    final ClusterState currentClusterState = event.state();
    if (!currentClusterState.blocks().hasGlobalBlock(GatewayService.STATE_NOT_RECOVERED_BLOCK)) {
        if (XPackPlugin.isReadyForXPackCustomMetadata(currentClusterState) == false) {
            logger.debug("cannot add license to cluster as the following nodes might not understand the license
metadata: {}",
                () -> XPackPlugin.nodesNotReadyForXPackCustomMetadata(currentClusterState));
            return;
        }

        final LicensesMetaData prevLicensesMetaData =
previousClusterState.getMetaData().custom(LicensesMetaData.TYPE);
        final LicensesMetaData currentLicensesMetaData =
currentClusterState.getMetaData().custom(LicensesMetaData.TYPE);
        if (logger.isDebugEnabled()) {
            logger.debug("previous [{}]", prevLicensesMetaData);
            logger.debug("current [{}]", currentLicensesMetaData);
        }
        // notify all interested plugins
```

```
        if (previousClusterState.blocks().hasGlobalBlock(GatewayService.STATE_NOT_RECOVERED_BLOCK)
                || prevLicensesMetaData == null) {
            if (currentLicensesMetaData != null) {
                onUpdate(currentLicensesMetaData);
            }
        } else if (!prevLicensesMetaData.equals(currentLicensesMetaData)) {
            onUpdate(currentLicensesMetaData);
        }

        License currentLicense = null;
        boolean noLicenseInPrevMetadata = prevLicensesMetaData == null || prevLicensesMetaData.getLicense()
== null;
        if (noLicenseInPrevMetadata == false) {
            currentLicense = prevLicensesMetaData.getLicense();
        }
        boolean noLicenseInCurrentMetadata = (currentLicensesMetaData == null ||
currentLicensesMetaData.getLicense() == null);
        if (noLicenseInCurrentMetadata == false) {
            currentLicense = currentLicensesMetaData.getLicense();
        }

        boolean noLicense = noLicenseInPrevMetadata && noLicenseInCurrentMetadata;
        // auto-generate license if no licenses ever existed or if the current license is basic and
        // needs extended or if the license signature needs to be updated. this will trigger a subsequent cluster changed
event
        if (currentClusterState.getNodes().isLocalNodeElectedMaster() &&
                (noLicense || LicenseUtils.licenseNeedsExtended(currentLicense) ||
                    LicenseUtils.signatureNeedsUpdate(currentLicense, currentClusterState.nodes()))) {
            registerOrUpdateSelfGeneratedLicense();
        }
    } else if (logger.isDebugEnabled()) {
        logger.debug("skipped license notifications reason: [{}]",
GatewayService.STATE_NOT_RECOVERED_BLOCK);
    }
  }

  private void updateLicenseState(LicensesMetaData licensesMetaData) {
    if (licensesMetaData != null) {
        updateLicenseState(getLicense(licensesMetaData), licensesMetaData.getMostRecentTrialVersion());
    }
  }

  protected void updateLicenseState(final License license, Version mostRecentTrialVersion) {
    if (license == LicensesMetaData.LICENSE_TOMBSTONE) {
        // implies license has been explicitly deleted
        licenseState.update(License.OperationMode.MISSING, false, mostRecentTrialVersion);
        return;
    }
```

```
        if (license != null) {
            long time = clock.millis();
            boolean active;
            if (license.expiryDate() == BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS) {
                active = true;
            } else {
                // We subtract the grace period from the current time to avoid overflowing on an expiration
                // date that is near Long.MAX_VALUE
                active = time >= license.issueDate() && time - GRACE_PERIOD_DURATION.getMillis() <
license.expiryDate();
            }
            licenseState.update(license.operationMode(), active, mostRecentTrialVersion);

            if (active) {
                if (time < license.expiryDate()) {
                    logger.debug("license [{}] - valid", license.uid());
                } else {
                    logger.warn("license [{}] - grace", license.uid());
                }
            } else {
                logger.warn("license [{}] - expired", license.uid());
            }
        }
    }

    /**
     * Notifies registered licensees of license state change and/or new active license
     * based on the license in <code>currentLicensesMetaData</code>.
     * Additionally schedules license expiry notifications and event callbacks
     * relative to the current license's expiry
     */
    private void onUpdate(final LicensesMetaData currentLicensesMetaData) {
        final License license = getLicense(currentLicensesMetaData);
        // license can be null if the trial license is yet to be auto-generated
        // in this case, it is a no-op
        if (license != null) {
            final License previousLicense = currentLicense.get();
            if (license.equals(previousLicense) == false) {
                currentLicense.set(license);
                license.setOperationModeFileWatcher(operationModeFileWatcher);
                scheduler.add(new SchedulerEngine.Job(LICENSE_JOB, nextLicenseCheck(license)));
                for (ExpirationCallback expirationCallback : expirationCallbacks) {
                    scheduler.add(new SchedulerEngine.Job(expirationCallback.getId(),
                            (startTime, now) ->
                                    expirationCallback.nextScheduledTimeForExpiry(license.expiryDate(), startTime, now)));
                }
                if (previousLicense != null) {
                    // remove operationModeFileWatcher to gc the old license object
```

```
            previousLicense.removeOperationModeFileWatcher();
        }
        logger.info("license [{}] mode [{}] - valid", license.uid(),
                license.operationMode().name().toLowerCase(Locale.ROOT));
    }
    updateLicenseState(license, currentLicensesMetaData.getMostRecentTrialVersion());
  }
}

// pkg private for tests
static SchedulerEngine.Schedule nextLicenseCheck(License license) {
  return (startTime, time) -> {
    if (time < license.issueDate()) {
        // when we encounter a license with a future issue date
        // which can happen with autogenerated license,
        // we want to schedule a notification on the license issue date
        // so the license is notificed once it is valid
        // see https://github.com/elastic/x-plugins/issues/983
        return license.issueDate();
    } else if (time < license.expiryDate()) {
        return license.expiryDate();
    } else if (time < license.expiryDate() + GRACE_PERIOD_DURATION.getMillis()) {
        return license.expiryDate() + GRACE_PERIOD_DURATION.getMillis();
    }
    return -1; // license is expired, no need to check again
  };
}

public static License getLicense(final MetaData metaData) {
  final LicensesMetaData licensesMetaData = metaData.custom(LicensesMetaData.TYPE);
  return getLicense(licensesMetaData);
}

static License getLicense(final LicensesMetaData metaData) {
  if (metaData != null) {
    License license = metaData.getLicense();
    if (license == LicensesMetaData.LICENSE_TOMBSTONE) {
        return license;
    } else if (license != null) {
        boolean autoGeneratedLicense = License.isAutoGeneratedLicense(license.signature());
        if ((autoGeneratedLicense && SelfGeneratedLicense.verify(license))
                || (!autoGeneratedLicense && LicenseVerifier.verifyLicense(license))) {
            return license;
        }
    }
  }
  return null;
}
```

```
  private static boolean isProductionMode(Settings settings, DiscoveryNode localNode) {
      final boolean singleNodeDisco = "single-
node".equals(DiscoveryModule.DISCOVERY_TYPE_SETTING.get(settings));
      return singleNodeDisco == false && isBoundToLoopback(localNode) == false;
  }

  private static boolean isBoundToLoopback(DiscoveryNode localNode) {
      return localNode.getAddress().address().getAddress().isLoopbackAddress();
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
import org.elasticsearch.action.support.master.TransportMasterNodeReadAction;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.transport.TransportService;

public class TransportGetTrialStatusAction extends TransportMasterNodeReadAction<GetTrialStatusRequest,
GetTrialStatusResponse> {

  @Inject
  public TransportGetTrialStatusAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                       ThreadPool threadPool, ActionFilters actionFilters,
                                       IndexNameExpressionResolver indexNameExpressionResolver) {
      super(settings, GetTrialStatusAction.NAME, transportService, clusterService, threadPool, actionFilters,
            indexNameExpressionResolver, GetTrialStatusRequest::new);
  }

  @Override
  protected String executor() {
      return ThreadPool.Names.SAME;
  }
```

```
    @Override
    protected GetTrialStatusResponse newResponse() {
        return new GetTrialStatusResponse();
    }

    @Override
    protected void masterOperation(GetTrialStatusRequest request, ClusterState state,
                                   ActionListener<GetTrialStatusResponse> listener) throws Exception {
        LicensesMetaData licensesMetaData = state.metaData().custom(LicensesMetaData.TYPE);
        listener.onResponse(new GetTrialStatusResponse(licensesMetaData == null ||
licensesMetaData.isEligibleForTrial()));

    }

    @Override
    protected ClusterBlockException checkBlock(GetTrialStatusRequest request, ClusterState state) {
        return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_READ);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.apache.logging.log4j.Logger;
import org.apache.logging.log4j.message.ParameterizedMessage;
import org.elasticsearch.Version;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ClusterStateUpdateTask;
import org.elasticsearch.cluster.metadata.MetaData;
import org.elasticsearch.common.Nullable;
import org.elasticsearch.xpack.core.XPackPlugin;

import java.time.Clock;
import java.util.Collections;
import java.util.Map;
import java.util.UUID;
import java.util.concurrent.atomic.AtomicReference;

public class StartBasicClusterTask extends ClusterStateUpdateTask {

    private static final String ACKNOWLEDGEMENT_HEADER = "This license update requires acknowledgement.
To acknowledge the license, " +
            "please read the following messages and call /start_basic again, this time with the \"acknowledge=true\"
parameter:";
```

```
    private final Logger logger;
    private final String clusterName;
    private final PostStartBasicRequest request;
    private final ActionListener<PostStartBasicResponse> listener;
    private final Clock clock;
    private AtomicReference<Map<String, String[]>> ackMessages = new
AtomicReference<>(Collections.emptyMap());

    StartBasicClusterTask(Logger logger, String clusterName, Clock clock, PostStartBasicRequest request,
                          ActionListener<PostStartBasicResponse> listener) {
        this.logger = logger;
        this.clusterName = clusterName;
        this.request = request;
        this.listener = listener;
        this.clock = clock;
    }

    @Override
    public void clusterStateProcessed(String source, ClusterState oldState, ClusterState newState) {
        LicensesMetaData oldLicensesMetaData = oldState.metaData().custom(LicensesMetaData.TYPE);
        logger.debug("license prior to starting basic license: {}", oldLicensesMetaData);
        License oldLicense = LicensesMetaData.extractLicense(oldLicensesMetaData);
        Map<String, String[]> acknowledgeMessages = ackMessages.get();
        if (acknowledgeMessages.isEmpty() == false) {
            listener.onResponse(new
PostStartBasicResponse(PostStartBasicResponse.Status.NEED_ACKNOWLEDGEMENT, acknowledgeMessages,
                    ACKNOWLEDGEMENT_HEADER));
        } else if (oldLicense != null && oldLicense.type().equals("basic")) {
            listener.onResponse(new
PostStartBasicResponse(PostStartBasicResponse.Status.ALREADY_USING_BASIC));
        }  else {
            listener.onResponse(new PostStartBasicResponse(PostStartBasicResponse.Status.GENERATED_BASIC));
        }
    }

    @Override
    public ClusterState execute(ClusterState currentState) throws Exception {
        XPackPlugin.checkReadyForXPackCustomMetadata(currentState);
        LicensesMetaData licensesMetaData = currentState.metaData().custom(LicensesMetaData.TYPE);
        License currentLicense = LicensesMetaData.extractLicense(licensesMetaData);
        if (currentLicense == null || currentLicense.type().equals("basic") == false) {
            long issueDate = clock.millis();
            MetaData.Builder mdBuilder = MetaData.builder(currentState.metaData());
            License.Builder specBuilder = License.builder()
                    .uid(UUID.randomUUID().toString())
                    .issuedTo(clusterName)
                    .maxNodes(LicenseService.SELF_GENERATED_LICENSE_MAX_NODES)
```

```
                .issueDate(issueDate)
                .type("basic")
                .expiryDate(LicenseService.BASIC_SELF_GENERATED_LICENSE_EXPIRATION_MILLIS);
            License selfGeneratedLicense = SelfGeneratedLicense.create(specBuilder, currentState.nodes());
            if (request.isAcknowledged() == false && currentLicense != null) {
                Map<String, String[]> ackMessages = LicenseService.getAckMessages(selfGeneratedLicense,
currentLicense);
                if (ackMessages.isEmpty() == false) {
                    this.ackMessages.set(ackMessages);
                    return currentState;
                }
            }
            Version trialVersion = null;
            if (licensesMetaData != null) {
                trialVersion = licensesMetaData.getMostRecentTrialVersion();
            }
            LicensesMetaData newLicensesMetaData = new LicensesMetaData(selfGeneratedLicense, trialVersion);
            mdBuilder.putCustom(LicensesMetaData.TYPE, newLicensesMetaData);
            return ClusterState.builder(currentState).metaData(mdBuilder).build();
        } else {
            return currentState;
        }
    }

    @Override
    public void onFailure(String source, @Nullable Exception e) {
        logger.error(new ParameterizedMessage("unexpected failure during [{}]", source), e);
        listener.onFailure(e);
    }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.common.xcontent.ToXContent;
import org.elasticsearch.common.xcontent.XContentBuilder;
import org.elasticsearch.rest.BytesRestResponse;
import org.elasticsearch.rest.RestController;
import org.elasticsearch.rest.RestRequest;
import org.elasticsearch.rest.RestResponse;
import org.elasticsearch.rest.action.RestBuilderListener;
import org.elasticsearch.xpack.core.XPackClient;
import org.elasticsearch.xpack.core.rest.XPackRestHandler;
```

```
import java.io.IOException;

import static org.elasticsearch.rest.RestRequest.Method.POST;

public class RestPostStartBasicLicense extends XPackRestHandler {

  RestPostStartBasicLicense(Settings settings, RestController controller) {
    super(settings);
    controller.registerHandler(POST, URI_BASE + "/license/start_basic", this);
  }

  @Override
  protected RestChannelConsumer doPrepareRequest(RestRequest request, XPackClient client) throws IOException
{
    PostStartBasicRequest startBasicRequest = new PostStartBasicRequest();
    startBasicRequest.acknowledge(request.paramAsBoolean("acknowledge", false));
    startBasicRequest.timeout(request.paramAsTime("timeout", startBasicRequest.timeout()));
    startBasicRequest.masterNodeTimeout(request.paramAsTime("master_timeout",
startBasicRequest.masterNodeTimeout()));
    return channel -> client.licensing().postStartBasic(startBasicRequest, new
RestBuilderListener<PostStartBasicResponse>(channel) {
      @Override
      public RestResponse buildResponse(PostStartBasicResponse response, XContentBuilder builder) throws
Exception {
        PostStartBasicResponse.Status status = response.getStatus();
        response.toXContent(builder, ToXContent.EMPTY_PARAMS);
        return new BytesRestResponse(status.getRestStatus(), builder);
      }
    });
  }

  @Override
  public String getName() {
    return "xpack_start_basic_action";
  }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
package org.elasticsearch.license;

import org.elasticsearch.ElasticsearchException;
import org.elasticsearch.action.ActionListener;
import org.elasticsearch.action.support.ActionFilters;
import org.elasticsearch.action.support.master.AcknowledgedResponse;
import org.elasticsearch.action.support.master.TransportMasterNodeAction;
```

```
import org.elasticsearch.cluster.ClusterState;
import org.elasticsearch.cluster.ack.ClusterStateUpdateResponse;
import org.elasticsearch.cluster.block.ClusterBlockException;
import org.elasticsearch.cluster.block.ClusterBlockLevel;
import org.elasticsearch.cluster.metadata.IndexNameExpressionResolver;
import org.elasticsearch.cluster.service.ClusterService;
import org.elasticsearch.common.inject.Inject;
import org.elasticsearch.common.settings.Settings;
import org.elasticsearch.protocol.xpack.license.DeleteLicenseRequest;
import org.elasticsearch.threadpool.ThreadPool;
import org.elasticsearch.transport.TransportService;

public class TransportDeleteLicenseAction extends TransportMasterNodeAction<DeleteLicenseRequest,
AcknowledgedResponse> {

  private final LicenseService licenseService;

  @Inject
  public TransportDeleteLicenseAction(Settings settings, TransportService transportService, ClusterService
clusterService,
                                      LicenseService licenseService, ThreadPool threadPool, ActionFilters actionFilters,
                                      IndexNameExpressionResolver indexNameExpressionResolver) {
    super(settings, DeleteLicenseAction.NAME, transportService, clusterService, threadPool, actionFilters,
        indexNameExpressionResolver, DeleteLicenseRequest::new);
    this.licenseService = licenseService;
  }

  @Override
  protected String executor() {
    return ThreadPool.Names.MANAGEMENT;
  }

  @Override
  protected AcknowledgedResponse newResponse() {
    return new AcknowledgedResponse();
  }

  @Override
  protected ClusterBlockException checkBlock(DeleteLicenseRequest request, ClusterState state) {
    return state.blocks().globalBlockedException(ClusterBlockLevel.METADATA_WRITE);
  }

  @Override
  protected void masterOperation(final DeleteLicenseRequest request, ClusterState state, final
ActionListener<AcknowledgedResponse>
      listener) throws ElasticsearchException {
    licenseService.removeLicense(request, new ActionListener<ClusterStateUpdateResponse>() {
      @Override
```

```
        public void onResponse(ClusterStateUpdateResponse clusterStateUpdateResponse) {
            listener.onResponse(new AcknowledgedResponse(clusterStateUpdateResponse.isAcknowledged()));
        }

        @Override
        public void onFailure(Exception e) {
            listener.onFailure(e);
        }
    });
  }
}
```

UnboundID LDAP SDK Free Use License

THIS IS AN AGREEMENT BETWEEN YOU ("YOU") AND PING IDENTITY CORPORATION ("PING IDENTITY") REGARDING YOUR USE OF UNBOUNDID LDAP SDK FOR JAVA AND ANY ASSOCIATED DOCUMENTATION, OBJECT CODE, COMPILED LIBRARIES, SOURCE CODE AND SOURCE FILES OR OTHER MATERIALS MADE AVAILABLE BY PING IDENTITY (COLLECTIVELY REFERRED TO IN THIS AGREEMENT AS THE ("SDK").

BY INSTALLING, ACCESSING OR OTHERWISE USING THE SDK, YOU ACCEPT THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT INSTALL, ACCESS OR USE THE SDK.

USE OF THE SDK. Subject to your compliance with this Agreement, Ping Identity grants to You a non-exclusive, royalty-free license, under Ping Identity's intellectual property rights in the SDK, to use, reproduce, modify and distribute this release of the SDK; provided that no license is granted herein under any patents that may be infringed by your modifications, derivative works or by other works in which the SDK may be incorporated (collectively, your "Applications"). You may reproduce and redistribute the SDK with your Applications provided that you (i) include this license file and an unmodified copy of the unboundid-ldapsdk.jar file; and (ii) such redistribution is subject to a license whose terms do not conflict with or contradict the terms of this Agreement. You may also reproduce and redistribute the SDK without your Applications provided that you redistribute the SDK complete and unmodified (i.e., with all "read me" files, copyright notices, and other legal notices and terms that Ping Identity has included in the SDK).

SCOPE OF LICENSES. This Agreement does not grant You the right to use any Ping Identity intellectual property which is not included as part of the SDK. The SDK is licensed, not sold. This Agreement only gives You some rights to use the SDK. Ping Identity reserves all other rights. Unless applicable law gives You more rights despite this limitation, You may use the SDK only as expressly permitted in this Agreement.

SUPPORT. Ping Identity is not obligated to provide any technical or other support ("Support Services") for the SDK to You under this Agreement. However, if Ping Identity chooses to provide any Support Services to You, Your use of

such Support Services will be governed by then-current Ping Identity support policies.

TERMINATION. Ping Identity reserves the right to discontinue offering the SDK and to modify the SDK at any time in its sole discretion. Notwithstanding anything contained in this Agreement to the contrary, Ping Identity may also, in its sole discretion, terminate or suspend access to the SDK to You or any end user at any time. In addition, if you fail to comply with the terms of this Agreement, then any rights granted herein will be automatically terminated if such failure is not corrected within 30 days of the initial notification of such failure. You acknowledge that termination and/or monetary damages may not be a sufficient remedy if You breach this Agreement and that Ping Identity will be entitled, without waiving any other rights or remedies, to injunctive or equitable relief as may be deemed proper by a court of competent jurisdiction in the event of a breach. Ping Identity may also terminate this Agreement if the SDK becomes, or in Ping Identity's reasonable opinion is likely to become, the subject of a claim of intellectual property infringement or trade secret misappropriation. All rights and licenses granted herein will simultaneously and automatically terminate upon termination of this Agreement for any reason.

DISCLAIMER OF WARRANTY. THE SDK IS PROVIDED "AS IS" AND PING IDENTITY DOES NOT WARRANT THAT THE SDK WILL BE ERROR-FREE, VIRUS-FREE, WILL PERFORM IN AN UNINTERRUPTED, SECURE OR TIMELY MANNER, OR WILL INTEROPERATE WITH OTHER HARDWARE, SOFTWARE, SYSTEMS OR DATA. TO THE MAXIMUM EXTENT ALLOWED BY LAW, ALL
CONDITIONS, REPRESENTATIONS AND WARRANTIES, WHETHER EXPRESS, IMPLIED, STATUTORY OR OTHERWISE INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE (EVEN IF PING IDENTITY HAD BEEN INFORMED OF SUCH PURPOSE), OR NON-INFRINGEMENT OF THIRD PARTY RIGHTS ARE HEREBY DISCLAIMED.

LIMITATION OF LIABILITY. IN NO EVENT WILL PING IDENTITY OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, REVENUE, DATA OR DATA USE, BUSINESS INTERRUPTION, COST OF COVER, DIRECT, INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SDK OR IN ANY WAY RELATED TO THIS AGREEMENT, EVEN IF PING IDENTITY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

ADDITIONAL RIGHTS. Certain states do not allow the exclusion of implied warranties or limitation of liability for certain kinds of damages, so the exclusion of limited warranties and limitation of liability set forth above may not apply to You.

EXPORT RESTRICTIONS. The SDK is subject to United States export control laws. You acknowledge and agree that You are responsible for compliance with all domestic and international export laws and regulations that apply to the SDK.

MISCELLANEOUS. This Agreement constitutes the entire agreement with respect to the SDK. If any provision of this Agreement shall be held to be invalid, illegal or unenforceable, the validity, legality and enforceability of the remaining provisions shall in no way be affected or impaired thereby. This Agreement and performance hereunder shall be governed by and construed in accordance with the laws of the State of Texas without regard to its conflict of laws rules. Any disputes related to this Agreement shall be exclusively litigated in the state or federal courts located in Travis County, Texas.
Apache Commons Compress
Copyright 2002-2015 The Apache Software Foundation

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

The files in the package org.apache.commons.compress.archivers.sevenz
were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/),
which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (http://www.7-zip.org/sdk.html)
Apache POI
Copyright 2003-2015 The Apache Software Foundation

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

This product contains parts that were originally based on software from BEA.
Copyright (c) 2000-2003, BEA Systems, <http://www.bea.com/>.

This product contains W3C XML Schema documents. Copyright 2001-2003 (c)
World Wide Web Consortium (Massachusetts Institute of Technology, European
Research Consortium for Informatics and Mathematics, Keio University)

This product contains the Piccolo XML Parser for Java
(http://piccolo.sourceforge.net/). Copyright 2002 Yuval Oren.

This product contains the chunks_parse_cmds.tbl file from the vsdump program.
Copyright (C) 2006-2007 Valek Filippov (frob@df.ru)

This product contains parts of the eID Applet project
(http://eid-applet.googlecode.com). Copyright (c) 2009-2014
FedICT (federal ICT department of Belgium), e-Contract.be BVBA (https://www.e-contract.be),
Bart Hanssens from FedICT

```
  ==========================================================================
  ==  NOTICE file corresponding to section 4(d) of the Apache License,   ==
  ==  Version 2.0, in this case for the Apache XmlBeans distribution.    ==
  ==========================================================================
```

This product includes software developed by

The Apache Software Foundation (http://www.apache.org/).

Portions of this software were originally based on the following:
- software copyright (c) 2000-2003, BEA Systems, <http://www.bea.com/>.

Aside from contributions to the Apache XMLBeans project, this software also includes:

- one or more source files from the Apache Xerces-J and Apache Axis products, Copyright (c) 1999-2003 Apache Software Foundation

- W3C XML Schema documents Copyright 2001-2003 (c) World Wide Web Consortium (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University)

- resolver.jar from Apache Xml Commons project, Copyright (c) 2001-2003 Apache Software Foundation

- Piccolo XML Parser for Java from http://piccolo.sourceforge.net/, Copyright 2002 Yuval Oren under the terms of the Apache Software License 2.0

- JSR-173 Streaming API for XML from http://sourceforge.net/projects/xmlpullparser/, Copyright 2005 BEA under the terms of the Apache Software License 2.0

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

as modifying the License.

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

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

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

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

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

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

APACHE POI SUBCOMPONENTS:

Apache POI includes subcomponents with separate copyright notices and
license terms. Your use of these subcomponents is subject to the terms
and conditions of the following licenses:

Office Open XML schemas (ooxml-schemas-1.1.jar)

The Office Open XML schema definitions used by Apache POI are
a part of the Office Open XML ECMA Specification (ECMA-376, [1]).
As defined in section 9.4 of the ECMA bylaws [2], this specification
is available to all interested parties without restriction:

9.4 All documents when approved shall be made available to
all interested parties without restriction.

Furthermore, both Microsoft and Adobe have granted patent licenses
to this work [3,4,5].

[1] http://www.ecma-international.org/publications/standards/Ecma-376.htm
[2] http://www.ecma-international.org/memento/Ecmabylaws.htm
[3] http://www.microsoft.com/openspecifications/en/us/programs/osp/default.aspx
[4] http://www.ecma-international.org/publications/files/ECMA-ST/Ecma%20PATENT/Patent%20statements%20ok/ECMA-376%20Edition%202%20Microsoft%20Patent%20Declaration.pdf
[5] http://www.ecma-international.org/publications/files/ECMA-ST/Ecma%20PATENT/Patent%20statements%20ok/ECMA-376%20Adobe%20Patent%20Declaration.pdf

JUnit test library (junit-4.11.jar)

Common Public License - v 1.0

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

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and
documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

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

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

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

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

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

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

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

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright

license set forth in this Agreement.

3. REQUIREMENTS

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

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

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

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

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program,

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

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

5. NO WARRANTY

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

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING

WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

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

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

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses

to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

with Licensor regarding such Contributions.

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

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

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

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

END OF TERMS AND CONDITIONS

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

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

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

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

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

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(http://www.bouncycastle.org)

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

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

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

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

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

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

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

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

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

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

the Derivative Works; and

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

Copyright [yyyy] [name of copyright owner]

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

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

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

See the License for the specific language governing permissions and
limitations under the License.

APACHE TIKA SUBCOMPONENTS

Apache Tika includes a number of subcomponents with separate copyright notices
and license terms. Your use of these subcomponents is subject to the terms and
conditions of the following licenses.

MIME type information from file-4.26.tar.gz (http://www.darwinsys.com/file/)

Copyright (c) Ian F. Darwin 1986, 1987, 1989, 1990, 1991, 1992, 1994, 1995.
Software written by Ian F. Darwin and others;
maintained 1994- Christos Zoulas.

This software is not subject to any export provision of the United States
Department of Commerce, and may be exported to any country or planet.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice immediately at the beginning of the file, without modification,
this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

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

Charset detection code from ICU4J (http://site.icu-project.org/)

Copyright (c) 1995-2009 International Business Machines Corporation
and others

All rights reserved.

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

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

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

Parsing functionality provided by the NetCDF Java Library (http://www.unidata.ucar.edu/software/netcdf-java/)

Copyright 1993-2010 University Corporation for Atmospheric Research/Unidata

Portions of this software were developed by the Unidata Program at the University Corporation for Atmospheric Research.

Access and use of this software shall impose the following obligations and understandings on the user. The user is granted the right, without any fee or cost, to use, copy, modify, alter, enhance and distribute this software, and any derivative works thereof, and its supporting documentation for any purpose whatsoever, provided that this entire notice appears in all copies of the software, derivative works and supporting documentation. Further, UCAR requests that the user credit UCAR/Unidata in any publications that result from the use of this software or in any product that includes this software, although this is not an obligation. The names UCAR and/or Unidata, however, may not be used in any advertising or publicity to endorse or promote any products or commercial entity unless specific written permission is obtained from UCAR/Unidata. The user also understands that UCAR/Unidata is not obligated to provide the user with any support, consulting, training or assistance of any kind with regard to the use, operation and performance of this software nor to provide the user with any updates, revisions, new versions or "bug fixes."

THIS SOFTWARE IS PROVIDED BY UCAR/UNIDATA "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL UCAR/UNIDATA BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL
DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE ACCESS,
USE OR PERFORMANCE OF THIS SOFTWARE.

IPTC Photo Metadata descriptions are taken from the IPTC Photo Metadata Standard, July 2010, Copyright 2010 International Press Telecommunications Council.

1. The Specifications and Materials are licensed for use only on the condition that you agree to be bound by the terms of this license. Subject to this and other licensing requirements contained herein, you may, on a non-exclusive basis, use the Specifications and Materials.
2. The IPTC openly provides the Specifications and Materials for voluntary use by individuals, partnerships, companies, corporations, organizations and any other entity for use at the entity's own risk. This disclaimer, license and release is intended to apply to the IPTC, its officers, directors, agents, representatives, members, contributors, affiliates, contractors, or co-venturers acting jointly or severally.
3. The Document and translations thereof may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the copyright and license notices and references to the IPTC appearing in the Document and the terms of this Specifications License Agreement are included on all such copies and derivative works. Further, upon the receipt of written permission from the IPTC, the Document may be modified for the purpose of developing applications that use IPTC Specifications or as required to translate the Document into languages other than English.
4. Any use, duplication, distribution, or exploitation of the Document and Specifications and Materials in any manner is at your own risk.
5. NO WARRANTY, EXPRESSED OR IMPLIED, IS MADE REGARDING THE ACCURACY, ADEQUACY, COMPLETENESS, LEGALITY, RELIABILITY OR USEFULNESS OF ANY INFORMATION CONTAINED IN THE DOCUMENT OR IN ANY SPECIFICATION OR OTHER PRODUCT OR SERVICE PRODUCED OR SPONSORED BY THE IPTC. THE DOCUMENT AND THE INFORMATION CONTAINED HEREIN AND INCLUDED IN ANY SPECIFICATION OR OTHER PRODUCT OR SERVICE OF THE IPTC IS PROVIDED ON AN "AS IS" BASIS. THE IPTC DISCLAIMS ALL WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, ANY ACTUAL OR ASSERTED WARRANTY OF NON-INFRINGEMENT OF PROPRIETARY RIGHTS, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. NEITHER THE IPTC NOR ITS CONTRIBUTORS SHALL BE HELD LIABLE FOR ANY IMPROPER OR INCORRECT USE OF INFORMATION. NEITHER THE IPTC NOR ITS CONTRIBUTORS ASSUME ANY RESPONSIBILITY FOR ANYONE'S USE OF INFORMATION PROVIDED BY THE IPTC. IN NO EVENT SHALL THE IPTC OR ITS CONTRIBUTORS BE LIABLE TO ANYONE FOR DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, COMPENSATORY DAMAGES, LOST PROFITS, LOST DATA OR ANY FORM OF SPECIAL, INCIDENTAL, INDIRECT, CONSEQUENTIAL OR PUNITIVE DAMAGES OF ANY KIND WHETHER BASED ON BREACH OF CONTRACT OR WARRANTY, TORT,

PRODUCT LIABILITY OR OTHERWISE.

6. The IPTC takes no position regarding the validity or scope of any Intellectual Property or other rights that might be claimed to pertain to the implementation or use of the technology described in the Document or the extent to which any license under such rights might or might not be available. The IPTC does not represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication, assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of the Specifications and Materials, can be obtained from the Managing Director of the IPTC.

7. By using the Specifications and Materials including the Document in any manner or for any purpose, you release the IPTC from all liabilities, claims, causes of action, allegations, losses, injuries, damages, or detriments of any nature arising from or relating to the use of the Specifications, Materials or any portion thereof. You further agree not to file a lawsuit, make a claim, or take any other formal or informal legal action against the IPTC, resulting from your acquisition, use, duplication, distribution, or exploitation of the Specifications, Materials or any portion thereof. Finally, you hereby agree that the IPTC is not liable for any direct, indirect, special or consequential damages arising from or relating to your acquisition, use, duplication, distribution, or exploitation of the Specifications, Materials or any portion thereof.

8. Specifications and Materials may be downloaded or copied provided that ALL copies retain the ownership, copyright and license notices.

9. Materials may not be edited, modified, or presented in a context that creates a misleading or false impression or statement as to the positions, actions, or statements of the IPTC.

10. The name and trademarks of the IPTC may not be used in advertising, publicity, or in relation to products or services and their names without the specific, written prior permission of the IPTC. Any permitted use of the trademarks of the IPTC, whether registered or not, shall be accompanied by an appropriate mark and attribution, as agreed with the IPTC.

11. Specifications may be extended by both members and non-members to provide additional functionality (Extension Specifications) provided that there is a clear recognition of the IPTC IP and its ownership in the Extension Specifications and the related documentation and provided that the extensions are clearly identified and provided that a perpetual license is granted by the creator of the Extension Specifications for other members and non-members to use the Extension Specifications and to continue extensions of the Extension Specifications. The IPTC does not waive any of its rights in the Specifications and Materials in this context. The Extension Specifications may be considered the intellectual property of their creator. The IPTC expressly disclaims any responsibility for damage caused by an extension to the Specifications.

12. Specifications and Materials may be included in derivative work of both members and non-members provided that there is a clear recognition of the IPTC IP and its ownership in the derivative work and its related documentation. The IPTC does not waive any of its rights in the Specifications and Materials in this context. Derivative work in its entirety may be considered the intellectual property of the creator of the work .The IPTC expressly disclaims any responsibility for damage caused when its IP is used in a derivative context.

13. This Specifications License Agreement is perpetual subject to your conformance to the terms of this Agreement. The IPTC may terminate this Specifications License Agreement immediately upon your breach of this Agreement and, upon such termination you will cease all use, duplication, distribution, and/or exploitation in any manner of the Specifications and Materials.

14. This Specifications License Agreement reflects the entire agreement of the parties regarding the subject matter hereof and supersedes all prior agreements or representations regarding such matters, whether written or oral. To the extent any portion or provision of this Specifications License Agreement is found to be illegal or unenforceable, then the remaining provisions of this Specifications License Agreement will remain in full force and effect and the illegal or unenforceable provision will be construed to give it such effect as it may properly have that is consistent with the intentions of the parties.

15. This Specifications License Agreement may only be modified in writing signed by an authorized representative

of the IPTC.

16. This Specifications License Agreement is governed by the law of United Kingdom, as such law is applied to contracts made and fully performed in the United Kingdom. Any disputes arising from or relating to this Specifications License Agreement will be resolved in the courts of the United Kingdom. You consent to the jurisdiction of such courts over you and covenant not to assert before such courts any objection to proceeding in such forums.

JUnRAR (https://github.com/edmund-wagner/junrar/)

```
    JUnRAR is based on the UnRAR tool, and covered by the same license
    It was formerly available from http://java-unrar.svn.sourceforge.net/

******   *****   ******   UnRAR - free utility for RAR archives
**  **  **  **  **  **  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
******  *******  ******    License for use and distribution of
**  **  **  **  **  **  ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
**  **  **  **  **  **        FREE portable version
                      ~~~~~~~~~~~~~~~~~~~~~~

    The source code of UnRAR utility is freeware. This means:

  1. All copyrights to RAR and the utility UnRAR are exclusively
     owned by the author - Alexander Roshal.

  2. The UnRAR sources may be used in any software to handle RAR
     archives without limitations free of charge, but cannot be used
     to re-create the RAR compression algorithm, which is proprietary.
     Distribution of modified UnRAR sources in separate form or as a
     part of other software is permitted, provided that it is clearly
     stated in the documentation and source comments that the code may
     not be used to develop a RAR (WinRAR) compatible archiver.

  3. The UnRAR utility may be freely distributed. It is allowed
     to distribute UnRAR inside of other software packages.

  4. THE RAR ARCHIVER AND THE UnRAR UTILITY ARE DISTRIBUTED "AS IS".
     NO WARRANTY OF ANY KIND IS EXPRESSED OR IMPLIED.  YOU USE AT
     YOUR OWN RISK. THE AUTHOR WILL NOT BE LIABLE FOR DATA LOSS,
     DAMAGES, LOSS OF PROFITS OR ANY OTHER KIND OF LOSS WHILE USING
     OR MISUSING THIS SOFTWARE.

  5. Installing and using the UnRAR utility signifies acceptance of
     these terms and conditions of the license.

  6. If you don't agree with terms of the license you must remove
     UnRAR files from your storage devices and cease to use the
     utility.
```

Thank you for your interest in RAR and UnRAR. Alexander L. Roshal

Sqlite (bundled in org.xerial's sqlite-jdbc)

This product bundles Sqlite, which is in the Public Domain. For details see: https://www.sqlite.org/copyright.html

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

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

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

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

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

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

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

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

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

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

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

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

6. Trademarks. This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

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

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

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

END OF TERMS AND CONDITIONS

THIS PRODUCT ALSO INCLUDES THIRD PARTY SOFTWARE REDISTRIBUTED UNDER THE FOLLOWING LICENSES:

Apache Commons Logging,
The Apache Software License, Version 1.1 (commons-logging-1.1.1.jar)

The Apache Software License, Version 1.1

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

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

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

3. The end-user documentation included with the redistribution,
   if any, must include the following acknowledgment:
      "This product includes software developed by the
       Apache Software Foundation (http://www.apache.org/)."
   Alternately, this acknowledgment may appear in the software itself,
   if and wherever such third-party acknowledgments normally appear.

4. The names "Apache" and "Apache Software Foundation" must
   not be used to endorse or promote products derived from this
   software without prior written permission. For written
   permission, please contact apache@apache.org.

5. Products derived from this software may not be called "Apache",
   nor may "Apache" appear in their name, without prior written
   permission of the Apache Software Foundation.

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

Test messages from the Perl-MIME-Tools project,

The "Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make reasonable modifications.

Definitions:

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

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

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

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

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

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

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

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

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

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

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

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

d) make other distribution arrangements with the Copyright Holder.

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

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

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

c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this

Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.

7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

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

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

The End

Copyright (c) 2006-2007, www.jempbox.org
All rights reserved.

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

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

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY

DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

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

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

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

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

4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

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

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

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

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

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

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

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

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

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

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

END OF TERMS AND CONDITIONS

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

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

same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

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

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

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

EXTERNAL COMPONENTS

Apache PDFBox includes a number of components with separate copyright notices
and license terms. Your use of these components is subject to the terms and
conditions of the following licenses.

Contributions made to the original PDFBox and FontBox projects:

Copyright (c) 2002-2007, www.pdfbox.org
All rights reserved.

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

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

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

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

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR

SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Adobe Font Metrics (AFM) for PDF Core 14 Fonts

This file and the 14 PostScript(R) AFM files it accompanies may be used, copied, and distributed for any purpose and without charge, with or without modification, provided that all copyright notices are retained; that the AFM files are not distributed without this file; that all modifications to this file or any of the AFM files are prominently noted in the modified file(s); and that this paragraph is not modified. Adobe Systems has no responsibility or obligation to support the use of the AFM files.

CMaps for PDF Fonts (http://opensource.adobe.com/wiki/display/cmap/Downloads)

Copyright 1990-2009 Adobe Systems Incorporated.
All rights reserved.

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

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

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

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

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

PaDaF PDF/A preflight (http://sourceforge.net/projects/padaf)

Copyright 2010 Atos Worldline SAS

Licensed by Atos Worldline SAS under one or more
contributor license agreements. See the NOTICE file distributed with
this work for additional information regarding copyright ownership.
Atos Worldline SAS licenses this file to You under the Apache License, Version 2.0
(the "License"); you may not use this file except in compliance with
the License. You may obtain a copy of the License at

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

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

OSXAdapter

Version: 2.0

Disclaimer: IMPORTANT: This Apple software is supplied to you by
Apple Inc. ("Apple") in consideration of your agreement to the
following terms, and your use, installation, modification or
redistribution of this Apple software constitutes acceptance of these
terms. If you do not agree with these terms, please do not use,
install, modify or redistribute this Apple software.

In consideration of your agreement to abide by the following terms, and
subject to these terms, Apple grants you a personal, non-exclusive
license, under Apple's copyrights in this original Apple software (the
"Apple Software"), to use, reproduce, modify and redistribute the Apple
Software, with or without modifications, in source and/or binary forms;
provided that if you redistribute the Apple Software in its entirety and
without modifications, you must retain this notice and the following
text and disclaimers in all such redistributions of the Apple Software.
Neither the name, trademarks, service marks or logos of Apple Inc.
may be used to endorse or promote products derived from the Apple
Software without specific prior written permission from Apple. Except
as expressly stated in this notice, no other rights or licenses, express
or implied, are granted by Apple herein, including but not limited to
any patent rights that may be infringed by your derivative works or by
other works in which the Apple Software may be incorporated.

The Apple Software is provided by Apple on an "AS IS" basis. APPLE
MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION

THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, REGARDING THE APPLE SOFTWARE OR ITS USE AND OPERATION ALONE OR IN COMBINATION WITH YOUR PRODUCTS.

IN NO EVENT SHALL APPLE BE LIABLE FOR ANY SPECIAL, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) ARISING IN ANY WAY OUT OF THE USE, REPRODUCTION, MODIFICATION AND/OR DISTRIBUTION OF THE APPLE SOFTWARE, HOWEVER CAUSED AND WHETHER UNDER THEORY OF CONTRACT, TORT (INCLUDING NEGLIGENCE), STRICT LIABILITY OR OTHERWISE, EVEN IF APPLE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (C) 2003-2007 Apple, Inc., All Rights Reserved
Apache Commons Logging
Copyright 2003-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache PDFBox
Copyright 2014 The Apache Software Foundation

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

Based on source code originally developed in the PDFBox and
FontBox projects.

Copyright (c) 2002-2007, www.pdfbox.org

Based on source code originally developed in the PaDaF project.
Copyright (c) 2010 Atos Worldline SAS

Includes the Adobe Glyph List
Copyright 1997, 1998, 2002, 2007, 2010 Adobe Systems Incorporated.

Includes the Zapf Dingbats Glyph List
Copyright 2002, 2010 Adobe Systems Incorporated.

Includes OSXAdapter
Copyright (C) 2003-2007 Apple, Inc., All Rights Reserved
Apache Tika
Copyright 2015 The Apache Software Foundation

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

Copyright 1993-2010 University Corporation for Atmospheric Research/Unidata

This software contains code derived from UCAR/Unidata's NetCDF library.

Tika-server component uses CDDL-licensed dependencies: jersey (http://jersey.java.net/) and Grizzly (http://grizzly.java.net/)

Tika-parsers component uses CDDL/LGPL dual-licensed dependency: jhighlight (https://github.com/codelibs/jhighlight)

OpenCSV: Copyright 2005 Bytecode Pty Ltd. Licensed under the Apache License, Version 2.0

IPTC Photo Metadata descriptions Copyright 2010 International Press Telecommunications Council.
Apache Commons CLI
Copyright 2001-2009 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
Copyright (c) 2004-2014 QOS.ch
All rights reserved.

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

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

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache Commons IO
Copyright 2002-2014 The Apache Software Foundation

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

MOZILLA PUBLIC LICENSE
Version 1.1

---

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this

License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a

copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters
(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming

interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.
Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.
You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.
You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ

from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code

governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days

of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer

software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in

compliance with the License. You may obtain a copy of the License at
http://www.mozilla.org/MPL/

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

The Original Code is _____.

The Initial Developer of the Original Code is _____.
Portions created by _____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms
of the _____ license (the "[___] License"), in which case the
provisions of [_____] License are applicable instead of those
above. If you wish to allow use of your version of this file only
under the terms of the [____] License and not to allow others to use
your version of this file under the MPL, indicate your decision by
deleting the provisions above and replace them with the notice and
other provisions required by the [___] License. If you do not delete
the provisions above, a recipient may use your version of this file
under either the MPL or the [___] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of
the notices in the Source Code files of the Original Code. You should
use the text of this Exhibit A rather than the text found in the
Original Code Source Code for Your Modifications.]

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(http://www.bouncycastle.org)

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

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

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Commons Collections

Copyright 2001-2015 The Apache Software Foundation

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

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

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

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

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

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

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

as of the date such litigation is filed.

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

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

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

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

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

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

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

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

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

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

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

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

END OF TERMS AND CONDITIONS

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

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

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

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

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

==========
== NOTICE file for use with the Apache License, Version 2.0, ==
==========

Apache JAMES Mime4j
Copyright 2004-2010 The Apache Software Foundation

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

This product test suite includes data (mimetools-testmsgs folder) developed
by Eryq and ZeeGee Software Inc as part of the "MIME-tools" Perl5 toolkit
and licensed under the Artistic License

Licensing of XZ for Java
==========

All the files in this package have been written by Lasse Collin
and/or Igor Pavlov. All these files have been put into the
public domain. You can do whatever you want with these files.

This software is provided "as is", without any warranty.

Apache Commons Lang
Copyright 2001-2014 The Apache Software Foundation

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

This product includes software from the Spring Framework,

under the Apache License 2.0 (see: StringUtils.containsWhitespace())
Apache HttpComponents Client
Copyright 1999-2015 The Apache Software Foundation

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

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

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

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

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

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

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

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

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

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

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

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

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

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

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

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

6. Trademarks. This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

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

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

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

END OF TERMS AND CONDITIONS

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

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

identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

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

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

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

Apache Commons Codec
Copyright 2002-2015 The Apache Software Foundation

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

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==========

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.
Copyright (c) 2007-2009, JSR305 expert group
All rights reserved.

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

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

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimer in the documentation
  and/or other materials provided with the distribution.
* Neither the name of the JSR305 expert group nor the names of its
  contributors may be used to endorse or promote products derived from
  this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Apache HttpComponents Core
Copyright 2005-2014 The Apache Software Foundation

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

This project contains annotations derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls. See http://www.jcip.net

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

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

source, and configuration files.

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

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

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

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

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

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

You may add Your own copyright statement to Your modifications and

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

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

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

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

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

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

defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

==========

This project contains annotations in the package org.apache.http.annotation
which are derived from JCIP-ANNOTATIONS
Copyright (c) 2005 Brian Goetz and Tim Peierls.
See http://www.jcip.net and the Creative Commons Attribution License
(http://creativecommons.org/licenses/by/2.5)
Full text: http://creativecommons.org/licenses/by/2.5/legalcode

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

"Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
"Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
"Licensor" means the individual or entity that offers the Work under the terms of this License.
"Original Author" means the individual or entity who created the Work.
"Work" means the copyrightable work of authorship offered under the terms of this License.
"You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable

laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
to create and reproduce Derivative Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.

For the avoidance of doubt, where the work is a musical composition:
Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.
Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).
Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions.The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.
If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative

Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTIBILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. Termination

This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

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

No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

Apache Commons Configuration
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
This product includes software developed by The Apache Software
Foundation (http://www.apache.org/).
Copyright (c) <YEAR>, <OWNER>
All rights reserved.

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

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

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

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

Apache Commons Codec
Copyright 2002-2014 The Apache Software Foundation

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

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==========

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.
Copyright (c) 2004-2014 QOS.ch
All rights reserved.

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

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

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
Apache Commons Lang
Copyright 2001-2015 The Apache Software Foundation

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

This product includes software from the Spring Framework,
under the Apache License 2.0 (see: StringUtils.containsWhitespace())
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0 1.

Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following: A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications; B. Any new file that contains any part of the Original Software or previous Modification; or C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

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

2. License Grants.

2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may

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

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY. COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER

THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS. The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS. This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such

jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS. As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.
Apache HTrace (incubating) is Apache 2.0 Licensed. See below for licensing
of dependencies that are NOT Apache Licensed.

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

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

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

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

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

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

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

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

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

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

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

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

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

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

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

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

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

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

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

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

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

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

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

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Units, a string formatting go library, is Copyright (c) 2014 Alec Thomas
and MIT licensed: https://github.com/alecthomas/units/blob/master/COPYING

D3, a javascript library for manipulating data, used by htrace-hbase
is Copyright 2010-2014, Michael Bostock and BSD licensed:
https://github.com/mbostock/d3/blob/master/LICENSE

Bootstrap, an html, css, and javascript framework, is
Copyright (c) 2011-2015 Twitter, Inc and MIT licensed:
https://github.com/twbs/bootstrap/blob/master/LICENSE

underscore, a javascript library of functional programming helpers, is
(c) 2009-2014 Jeremy Ashkenas, DocumentCloud and Investigative Reporters
& Editors and an MIT license:
https://github.com/jashkenas/underscore/blob/master/LICENSE

jquery, a javascript library, is Copyright jQuery Foundation and other
contributors, https://jquery.org/. The software consists of
voluntary contributions made by many individuals. For exact
contribution history, see the revision history
available at https://github.com/jquery/jquery
It is MIT licensed:
https://github.com/jquery/jquery/blob/master/LICENSE.txt

backbone, is a javascript library, that is Copyright (c) 2010-2014
Jeremy Ashkenas, DocumentCloud. It is MIT licensed:
https://github.com/jashkenas/backbone/blob/master/LICENSE

moment.js is a front end time conversion project.
It is (c) 2011-2014 Tim Wood, Iskren Chernev, Moment.js contributors
and shared under the MIT license:
https://github.com/moment/moment/blob/develop/LICENSE

CMP is an implementation of the MessagePack serialization format in
C. It is licensed under the MIT license:
https://github.com/camgunz/cmp/blob/master/LICENSE

go-codec is an implementation of several serialization and deserialization
codecs in Go. It is licensed under the MIT license:
https://github.com/ugorji/go/blob/master/LICENSE
Apache Commons Collections
Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).
Apache HTrace
Copyright 2015 The Apache Software Foundation

This product includes software developed at The Apache Software
Foundation (http://www.apache.org/).

In addition, this product includes software dependencies. See
the accompanying LICENSE.txt for a listing of dependencies
that are NOT Apache licensed (with pointers to their licensing)

Apache HTrace includes an Apache Thrift connector to Zipkin. Zipkin
is a distributed tracing system that is Apache 2.0 Licensed.
Copyright 2012 Twitter, Inc.

Our Owl logo we took from http://www.clker.com/clipart-13653.html.
It is public domain/free.
AWS SDK for Java
Copyright 2010-2014 Amazon.com, Inc. or its affiliates. All Rights Reserved.

This product includes software developed by
Amazon Technologies, Inc (http://www.amazon.com/).

****
THIRD PARTY COMPONENTS
****
This software includes third party software subject to the following copyrights:
- XML parsing and utility functions from JetS3t - Copyright 2006-2009 James Murty.
- JSON parsing and utility functions from JSON.org - Copyright 2002 JSON.org.
- PKCS#1 PEM encoded private key parsing and utility functions from oauth.googlecode.com - Copyright 1998-2010 AOL Inc.

The licenses for these third party components are included in LICENSE.txt
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or

modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients. rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient.s rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form

under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF

DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN

INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item,. as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such

responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have.

You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other

recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or

distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the

original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.
Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

1. You must give any other recipients of the Work or Derivative Works a copy of this License; and
2. You must cause any modified files to carry prominent notices stating that You changed the files; and
3. You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
4. If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR

PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Note: Other license terms may apply to certain, identified software files contained within or distributed with the accompanying software if such terms are included in the directory containing the accompanying software. Such other license terms will then apply in lieu of the terms of the software license above.

JSON processing code subject to the JSON License from JSON.org:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims,

the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You

distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise

make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)
The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show

them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this

License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License

which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later

version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked

independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2006 Envoi Solutions LLC

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Copyright 2016, Google Inc.
All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright 2014, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above

copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright 2016, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright 2008 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
/*
* Copyright (c) 2007-present, Stephen Colebourne & Michael Nascimento Santos
*
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
* * Redistributions of source code must retain the above copyright notice,
*   this list of conditions and the following disclaimer.
*
* * Redistributions in binary form must reproduce the above copyright notice,
*   this list of conditions and the following disclaimer in the documentation
*   and/or other materials provided with the distribution.
*
* * Neither the name of JSR-310 nor the names of its contributors
*   may be used to endorse or promote products derived from this software
*   without specific prior written permission.

*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
* A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
This software is Copyright (c) 2013 by MaxMind, Inc.

This is free software, licensed under the Apache License, Version 2.0.
This software is Copyright (c) 2014 by MaxMind, Inc.

This is free software, licensed under the Apache License, Version 2.0.
Apache Lucene
Copyright 2001-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Includes software from other Apache Software Foundation projects,
including, but not limited to:
- Apache Ant
- Apache Jakarta Regexp
- Apache Commons
- Apache Xerces

ICU4J, (under analysis/icu) is licensed under an MIT styles license
and Copyright (c) 1995-2008 International Business Machines Corporation and others

Some data files (under analysis/icu/src/data) are derived from Unicode data such
as the Unicode Character Database. See http://unicode.org/copyright.html for more
details.

Brics Automaton (under core/src/java/org/apache/lucene/util/automaton) is
BSD-licensed, created by Anders Mller. See http://www.brics.dk/automaton/

The levenshtein automata tables (under core/src/java/org/apache/lucene/util/automaton) were
automatically generated with the moman/finenight FSA library, created by
Jean-Philippe Barrette-LaPierre. This library is available under an MIT license,
see http://sites.google.com/site/rrettesite/moman and
http://bitbucket.org/jpbarrette/moman/overview/

The class org.apache.lucene.util.WeakIdentityMap was derived from
the Apache CXF project and is Apache License 2.0.

The Google Code Prettify is Apache License 2.0.
See http://code.google.com/p/google-code-prettify/

JUnit (junit-4.10) is licensed under the Common Public License v. 1.0
See http://junit.sourceforge.net/cpl-v10.html

This product includes code (JaspellTernarySearchTrie) from Java Spelling Checkin
g Package (jaspell): http://jaspell.sourceforge.net/
License: The BSD License (http://www.opensource.org/licenses/bsd-license.php)

The snowball stemmers in
 analysis/common/src/java/net/sf/snowball
were developed by Martin Porter and Richard Boulton.
The snowball stopword lists in
 analysis/common/src/resources/org/apache/lucene/analysis/snowball
were developed by Martin Porter and Richard Boulton.
The full snowball package is available from
 http://snowball.tartarus.org/

The KStem stemmer in
 analysis/common/src/org/apache/lucene/analysis/en
was developed by Bob Krovetz and Sergio Guzman-Lara (CIIR-UMass Amherst)
under the BSD-license.

The Arabic,Persian,Romanian,Bulgarian, Hindi and Bengali analyzers (common) come with a default
stopword list that is BSD-licensed created by Jacques Savoy. These files reside in:
analysis/common/src/resources/org/apache/lucene/analysis/ar/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/fa/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/ro/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/bg/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/hi/stopwords.txt,
analysis/common/src/resources/org/apache/lucene/analysis/bn/stopwords.txt
See http://members.unine.ch/jacques.savoy/clef/index.html.

The German,Spanish,Finnish,French,Hungarian,Italian,Portuguese,Russian and Swedish light stemmers
(common) are based on BSD-licensed reference implementations created by Jacques Savoy and
Ljiljana Dolamic. These files reside in:
analysis/common/src/java/org/apache/lucene/analysis/de/GermanLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/de/GermanMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/es/SpanishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fi/FinnishLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/fr/FrenchMinimalStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/hu/HungarianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/it/ItalianLightStemmer.java

analysis/common/src/java/org/apache/lucene/analysis/pt/PortugueseLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/ru/RussianLightStemmer.java
analysis/common/src/java/org/apache/lucene/analysis/sv/SwedishLightStemmer.java

The Stempel analyzer (stempel) includes BSD-licensed software developed
by the Egothor project http://egothor.sf.net/, created by Leo Galambos, Martin Kvapil,
and Edmond Nolan.

The Polish analyzer (stempel) comes with a default
stopword list that is BSD-licensed created by the Carrot2 project. The file resides
in stempel/src/resources/org/apache/lucene/analysis/pl/stopwords.txt.
See http://project.carrot2.org/license.html.

The SmartChineseAnalyzer source code (smartcn) was
provided by Xiaoping Gao and copyright 2009 by www.imdict.net.

WordBreakTestUnicode_*.java (under modules/analysis/common/src/test/)
is derived from Unicode data such as the Unicode Character Database.
See http://unicode.org/copyright.html for more details.

The Morfologik analyzer (morfologik) includes BSD-licensed software
developed by Dawid Weiss and Marcin Mikowski (http://morfologik.blogspot.com/).

Morfologik uses data from Polish ispell/myspell dictionary
(http://www.sjp.pl/slownik/en/) licenced on the terms of (inter alia)
LGPL and Creative Commons ShareAlike.

Morfologic includes data from BSD-licensed dictionary of Polish (SGJP)
(http://sgjp.pl/morfeusz/)

Servlet-api.jar and javax.servlet-*.jar are under the CDDL license, the original
source code for this can be found at http://www.eclipse.org/jetty/downloads.php

==========
Kuromoji Japanese Morphological Analyzer - Apache Lucene Integration
==========

This software includes a binary and/or source version of data from

mecab-ipadic-2.7.0-20070801

which can be obtained from

http://atilika.com/releases/mecab-ipadic/mecab-ipadic-2.7.0-20070801.tar.gz

or

http://jaist.dl.sourceforge.net/project/mecab/mecab-ipadic/2.7.0-20070801/mecab-ipadic-2.7.0-20070801.tar.gz

==========
mecab-ipadic-2.7.0-20070801 Notice
==========

Nara Institute of Science and Technology (NAIST),
the copyright holders, disclaims all warranties with regard to this
software, including all implied warranties of merchantability and
fitness, in no event shall NAIST be liable for
any special, indirect or consequential damages or any damages
whatsoever resulting from loss of use, data or profits, whether in an
action of contract, negligence or other tortuous action, arising out
of or in connection with the use or performance of this software.

A large portion of the dictionary entries
originate from ICOT Free Software. The following conditions for ICOT
Free Software applies to the current dictionary as well.

Each User may also freely distribute the Program, whether in its
original form or modified, to any third party or parties, PROVIDED
that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
on, or be attached to, the Program, which is distributed substantially
in the same form as set out herein and that such intended
distribution, if actually made, will neither violate or otherwise
contravene any of the laws and regulations of the countries having
jurisdiction over the User or the intended distribution itself.

NO WARRANTY

The program was produced on an experimental basis in the course of the
research and development conducted during the project and is provided
to users as so produced on an experimental basis. Accordingly, the
program is provided without any warranty whatsoever, whether express,
implied, statutory or otherwise. The term "warranty" used herein
includes, but is not limited to, any warranty of the quality,
performance, merchantability and fitness for a particular purpose of
the program and the nonexistence of any infringement or violation of
any right of any third party.

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all

damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
"use" as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.

In the case where the program, whether in its original form or
modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.

==========
Nori Korean Morphological Analyzer - Apache Lucene Integration
==========

This software includes a binary and/or source version of data from

mecab-ko-dic-2.0.3-20170922

which can be obtained from

https://bitbucket.org/eunjeon/mecab-ko-dic/downloads/mecab-ko-dic-2.0.3-20170922.tar.gz
Copyright (c) 2000-2015 The Legion of the Bouncy Castle Inc. (http://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software
and associated documentation files (the "Software"), to deal in the Software without restriction,
including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial
portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A
PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF

CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
DEALINGS IN THE SOFTWARE.
Format: https://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Copyright: Elasticsearch B.V. <info@elastic.co>
License: ${license.name}
${license.text}
Apache Portable Runtime Utility Library
Copyright (c) 2009 The Apache Software Foundation.

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were developed at the National Center
for Supercomputing Applications (NCSA) at the University of
Illinois at Urbana-Champaign.

This software contains code derived from the RSA Data Security
Inc. MD5 Message-Digest Algorithm, including various
modifications by Spyglass Inc., Carnegie Mellon University, and
Bell Communications Research, Inc (Bellcore).
name,version,revision,url,license,copyright
Apache Portable Runtime Iconv Library,1.2.1,,https://apr.apache.org,Apache-2.0,Copyright (c) 2000 Konstantin Chuguev
name,version,revision,url,license,copyright
RapidXml,1.13,,http://rapidxml.sourceforge.net,MIT,"Copyright (c) 2006, 2007 Marcin Kalicinski"
Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
   claim that you wrote the original software. If you use this software
   in a product, an acknowledgment in the product documentation would be
   appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
   misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

Use of this software is granted under one of the following two licenses,
to be chosen freely by the user.

1. Boost Software License - Version 1.0 - August 17th, 2003
==========

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2. The MIT License
==========

Copyright (c) 2006, 2007 Marcin Kalicinski

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MICROSOFT SOFTWARE LICENSE TERMS

MICROSOFT VISUAL STUDIO EXPRESS 2013 FOR WINDOWS DESKTOP

These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft

* updates,
* supplements,
* Internet-based services, and
* support services

for this software, unless other terms accompany those items. If so, those terms apply.

BY USING THE SOFTWARE, YOU ACCEPT THESE TERMS. IF YOU DO NOT ACCEPT THEM, DO NOT USE THE SOFTWARE.

AS DESCRIBED BELOW, USING SOME FEATURES ALSO OPERATES AS YOUR CONSENT TO THE TRANSMISSION OF CERTAIN STANDARD COMPUTER INFORMATION FOR INTERNET-BASED SERVICES.

If you comply with these license terms, you have the rights below.

1. INSTALLATION AND USE RIGHTS.

a. Installation and Use. One user may install and use any number of copies of the software on your devices to design, develop and test your programs.

b. Trial Edition. The initial installation of the software is a trial edition. The length of the trial period is thirty (30) days. You may convert your trial rights at any time to the full-use rights described below by registering with Microsoft. After the expiration of the 30-day trial period, without registering, the trial edition will stop running. You may not be able to access data used with the trial edition when it stops running.

c. Included Microsoft Programs. The software contains other Microsoft programs. The license terms with those programs apply to your use of them.

d. Third Party Notices. Additional copyright notices and license terms applicable to portions of the software are set forth in the ThirdPartyNotices file accompanying the Microsoft software. In addition to any terms and conditions of any third party license identified in the ThirdPartyNotices file, the disclaimer of warranty and limitation on and exclusion of remedies and damages provisions of this agreement shall apply to all software in this distribution.

e. Additional Licensing Requirements and/or Distribution Rights for Features Made Available with the Software:

* Windows Phone Store. If you intend to make your program available in the Windows Phone Store, you will need to comply with the terms and conditions of the Windows Phone Store Application Provider Agreement.

* Bing Maps. The software may include features that retrieve content such as maps, images and other data through the Bing Maps (or successor branded) application programming interface (the "Bing Maps API") to create reports displaying data on top of maps, aerial and hybrid imagery. If these features are included, you may use these features to create and view dynamic or static documents only in conjunction with and through methods and means of access integrated in the software. You may not otherwise copy, store, archive, or create a database of the entity information including business names, addresses and geocodes available through the Bing Maps API You may not use the Bing Maps API to provide sensor based guidance/routing, nor use any Road Traffic Data or Bird's Eye Imagery (or associated metadata) even if available through the Bing Maps API for any purpose. Your use of the Bing Maps API and associated content is also subject to the additional terms and conditions at http://go.microsoft.com/fwlink/?LinkId=21969.

* Additional Mapping APIs. The software may include application programming interfaces that provide maps and other related mapping features and services that are not provided by Bing (the "Additional Mapping APIs"). These Additional Mapping APIs are subject to additional terms and conditions and may require payment of fees to Microsoft and/or third party providers based on the use or volume of use of such Additional Mapping APIs. These terms and conditions will be provided when you obtain any necessary license keys to use such Additional Mapping APIs or when you review or receive documentation related to the use of such Additional Mapping APIs.

* Push Notifications. The Microsoft Push Notification Service may not be used to send notifications that are mission critical or otherwise could affect matters of life or death, including without limitation critical notifications related to a medical device or condition. MICROSOFT EXPRESSLY DISCLAIMS ANY WARRANTIES THAT THE USE OF THE MICROSOFT PUSH NOTIFICATION SERVICE OR DELIVERY OF MICROSOFT PUSH NOTIFICATION SERVICE NOTIFICATIONS WILL BE UNINTERRUPTED, ERROR FREE, OR OTHERWISE GUARANTEED TO OCCUR ON A REAL-TIME BASIS.

* Windows.Phone.Speech namespace API. Using speech recognition functionality via the Windows.Phone.Speech namespace APIs in a program requires the support of a speech recognition service. The service may require network connectivity at the time of recognition (e.g., when using a predefined grammar). In addition, the service may also collect speech-related data in order to provide and improve the service. The speech-related data may include, for example, information related to grammar size and string phrases in a grammar.

Also, in order for a user to use speech recognition on the phone they must first accept certain terms of use. The terms of use notify the user that data related to their use of the speech recognition service will be collected and used to provide and improve the service. If a user does not accept the terms of use and speech recognition is attempted by the application, the operation will not work and an error will be returned to the application.

* Location Framework. The software may contain a location framework component that enables support of location services in programs. In addition to the other limitations in this agreement, you must comply with all applicable local laws and regulations when using the location framework component or the rest of the software.

* Device ID Access. The software may contain a component that enables programs to access the device ID of the device that is running the program. In addition to the other limitations in this agreement, you must comply with all applicable local laws and regulations when using the device ID access component or the rest of the software.

* PlayReady Support. The software includes the Windows Phone Emulator, which contains Microsoft's PlayReady content access technology. Content owners use Microsoft PlayReady content access technology to help protect their intellectual property, including copyrighted content. This software uses PlayReady technology to access PlayReady-protected content and/or Microsoft Digital Rights Management (DRM) protected content. Microsoft may decide to revoke the software's ability to consume PlayReady-protected content for reasons including but not limited to (i) if a breach or potential breach of PlayReady technology occurs, (ii) proactive robustness enhancement, and (iii) if content owners require the revocation because the software fails to properly enforce restrictions on content usage. Revocation should not affect unprotected content or content protected by other content access technologies. Content owners may require you to upgrade PlayReady to access their content. If you decline an upgrade, you will not be able to access content that requires the upgrade and may not be able to install other operating system updates or upgrades.

* Microsoft Advertising SDKs. If you pass precise location data or other user related data (e.g., user identifier, profile data, behaviorally tracked user data) to the Microsoft Advertising SDK for Windows and Windows Phone or the Microsoft Advertising SDK, then your program must (a) notify end users that it will be collecting and using user related information and providing this information to Microsoft for Microsoft's advertising use, and (b) explicitly obtain affirmative user consent (e.g. the user must click an "Accept" or continue "Install" button) for this upon download of the software and/or application. In addition, you agree to: (a) comply with certification and other requirements for Windows and Windows Phone and Windows Store; (b) comply with Microsoft's privacy and other

policies in your collection and use of any user data; (c) not
collect or use any user identifier created or provided to you by
Microsoft for any purpose other than passing such identifier to a
Microsoft advertising service as part of your use of the service;
and (d) provide in your privacy policy and/or terms of use a link
that provides users with the ability to opt out of Microsoft
interest based advertising at the following location
https://choice.live.com/AdvertisementChoice/.
2. ADDITIONAL LICENSING REQUIREMENTS AND/OR USE RIGHTS.
a. Distributable Code. The software contains code that you are permitted
to distribute in programs you develop if you comply with the terms
below. (For this Section the term "distribution" also means deployment
of your programs for third parties to access over the Internet.)
i. Right to Use and Distribute. The code and text files listed below
are "Distributable Code."
* REDIST.TXT Files. You may copy and distribute the object code
form of code listed on the REDIST list located at
go.microsoft.com/fwlink/?LinkId=313603.
* Sample Code You may modify, copy and distribute the source and
object code form of code marked as "sample."
* Third Party Distribution. You may permit distributors of your
programs to copy and distribute the Distributable Code as part of
those programs.
ii. Distribution Requirements. For any Distributable Code you
distribute, you must
* add significant primary functionality to it in your
programs;
* for any Distributable Code having a filename extension of .lib,
distribute only the results of running such Distributable Code
through a linker with your program;
* distribute Distributable Code included in a setup program only
as part of that setup program without modification;
* require distributors and external end users to agree to
terms that protect it at least as much as this agreement;
* display your valid copyright notice on your programs; and
* indemnify, defend, and hold harmless Microsoft from any
claims, including attorneys' fees, related to the distribution or
use of your programs.
iii. Distribution Restrictions. You may not
* alter any copyright, trademark or patent notice in the
Distributable Code;
* use Microsoft's trademarks in your programs' names or in a
way that suggests your programs come from or are endorsed by
Microsoft;
* distribute Distributable Code to run on a platform other
than Microsoft operating systems, run-time technologies, or
application platforms;
* include Distributable Code in malicious, deceptive or

unlawful programs; or
* modify or distribute the source code of any Distributable
Code so that any part of it becomes subject to an Excluded
License. An Excluded License is one that requires, as a condition
of use, modification or distribution, that
* the code be disclosed or distributed in source code form;
or
* others have the right to modify it.
3. DATA.
a. Consent for Internet-Based Services. The software features described
below and in the Visual Studio 2013 privacy statement connect to
Microsoft or service provider computer systems over the Internet. In
some cases, you will not receive a separate notice when they connect.
You may switch off these features or not use them. For more information
about these features, see http://go.microsoft.com/fwlink/?LinkId=286720
. BY USING THESE FEATURES, YOU CONSENT TO THE TRANSMISSION OF THIS
INFORMATION. Microsoft does not use the information to identify or
contact you.
* Extensions and Updates; and New Project Dialog. Extensions and
Updates; and New Project Dialog can retrieve other software through
the Internet from Visual Studio Gallery, Microsoft Download Center
and MSDN Samples websites. To provide this other software,
Extensions and Updates and; and New Project Dialog send to
Microsoft the name and version of the software you are using and
language code of the device where you installed the software. In
addition, Extensions and Updates also contains an automatic update
feature that is on by default.
? For more information about this feature, including instructions
for to turning it off, see
http://go.microsoft.com/fwlink/?LinkId=286720 . You may turn
off this automatic update feature while the software is
running ("opt out"). Unless you expressly opt out of this
feature, the feature will (a) connect to Microsoft or service
provider computer systems over the Internet, (b) use Internet
protocols to send to the appropriate systems standard computer
information, such as your computer's Internet protocol
address, operating system, the name and version of the
software you are using and language code of the device where
you installed the software, and (c) prompt you to download
and/or install, current updates to the software or other third
party software. In some cases, you will not receive a separate
notice before this feature takes effect. By installing the
software, you consent to the transmission of standard computer
information.
* Notification Tool Window. The software includes a notification
tool window. When connected to the Internet, technologies that use
the notification tool window may provide information related to
extensions and other software or services that you have installed

or opted into and display those updates to you. For more information please see the privacy statement http://go.microsoft.com/fwlink/?LinkId=286720 .

* Digital Certificates. The software uses digital certificates. These digital certificates confirm the identity of Internet users sending X.509 standard encrypted information. They also can be used to digitally sign files and macros to verify the integrity and origin of the file contents. The software retrieves certificates and updates certificate revocation lists using the Internet, when available.

* Really Simple Syndication ("RSS") Feed. This software start page contains updated content that is supplied by means of an RSS feed online from Microsoft.

* Package Manager Features. For more information about these features, see the privacy statement available at go.microsoft.com/fwlink/?LinkId=205205. With respect to these package manager features, we or the third party service provider may use the computer information obtained through your use of these features, to improve our or their software and services. We or they may also share it with others, such as hardware and software vendors. They may use the information to improve how their products run with Microsoft software.

o Open Data Protocol (OData) Service. The Microsoft NuGet-Based Package Manager and the package manager feature of Microsoft ASP.NET Web Pages will access a list of packages that is supplied by means of an OData service online from Microsoft or a third-party service provider, located at go.microsoft.com/fwlink/?LinkID=206669. You may change the feed URL that the package manager feature initially points to at any time at your discretion.

Misuse of Internet-based Services. You may not use these services in any way that could harm them or impair anyone else's use of them. You may not use the services to try to gain unauthorized access to any service, data, account or network by any means.

b. Computer Information. The following features use Internet protocols, which send to the appropriate systems computer information, such as your Internet protocol address, the type of operating system, browser and name and version of the software you are using, and the language code of the device where you installed the software. Microsoft uses this information to make the Internet-based services available to you.

* Visual Studio Experience Improvement Program. This software uses the Visual Studio Experience Improvement Program ("VSEIP") which is on by default. VSEIP helps Microsoft collect information about problems that you have while using the software. Microsoft also uses VSEIP Information to improve its software and services. Microsoft may also share VSEIP data with others, such as agents acting on Microsoft's behalf and hardware and software vendors to improve how their products run with Microsoft software. To learn more about VSEIP, including how

you can opt out, see go.microsoft.com/fwlink/?LinkId=286720 .
* Data Collected Automatically. Microsoft automatically collects information identifying your installed Microsoft product, the operating system of the device, the CPU architecture of the operating system and data regarding the success or failure of the installation of the software, data identifying the cause of a crash in the product and information about the product license which is in use. Microsoft does not use this information to identify or contact you. For more information about privacy, see http://go.microsoft.com/fwlink/?LinkId=286720 .
c. Microsoft Accounts in Visual Studio. If you are running the software on Windows 8, Windows 7 with sign-in assistant, or any other version of Windows that supports providing a Microsoft account directly to the software and you are signed into a Microsoft account in those versions of Windows, you may automatically be signed into the software and Visual Studio services accessed by the software using the same Microsoft account. This allows you to access services within the software and roam the software's settings without being asked to reenter your Microsoft account credentials each time you start the software. For more information about signing into the software and the services available therein with a Microsoft Account, see the privacy statement http://go.microsoft.com/fwlink/?LinkId=286720 .
d. Use of Information. Microsoft may use the computer and services information to improve its software and services. Microsoft may also share it with others, such as hardware and software vendors. They may use the information to improve how their products run with Microsoft software.
4. Windows Components. The software may include Microsoft .NET Framework software, certain .dll's related to Microsoft Build technologies and packaging technologies, Windows controls, resource resolution and rendering technologies, Windows Identity Foundation, Windows Web Services (WWS) for Windows 2008 and Windows Library for JavaScript components. These software components are part of Windows and are licensed to you under separate Windows license terms or as a supplement to a Windows license. The product lifecycle support and license terms for the applicable Windows product apply to your use of these Windows components.
5. Microsoft .NET Framework Benchmark Testing. The software includes one or more components of the .NET Framework (".NET Components"). You may conduct internal benchmark testing of those components. You may disclose the results of any benchmark test of those components, provided that you comply with the conditions set forth at http://go.microsoft.com/fwlink/?LinkID=66406. Notwithstanding any other agreement you may have with Microsoft, if you disclose such benchmark test results, Microsoft shall have the right to disclose the results of benchmark tests it conducts of your products that compete with the applicable .NET Component, provided it complies with the same conditions set forth at http://go.microsoft.com/fwlink/?LinkID=66406.
6. SQL SERVER SOFTWARE AND WINDOWS SOFTWARE DEVELOPMENT KIT (WINDOWS SDK). The software is also accompanied by Microsoft SQL Server Software

and Windows SDK which are licensed to you under their own terms. The license terms for the SQL Server Software and Windows SDK are located in the "Licenses" folder of the software installation directory. If you do not agree to the components' license terms, you may not use them.

7. PACKAGE MANAGER AND THIRD PARTY SOFTWARE INSTALLATION FEATURES. The software includes the following features (each a "Feature"), each of which enables you to obtain software applications or packages through the Internet from other sources: Extension Manager, New Project Dialog, Web Platform Installer, and Microsoft NuGet-Based Package Manager. Those software applications and packages are offered and distributed in some cases by third parties and in some cases by Microsoft, but each such application or package is under its own license terms. Microsoft is not developing, distributing or licensing any of the third-party applications or packages to you, but instead, as a convenience, enables you to use the Features to access or obtain those applications or packages directly from the third-party application or package providers. By using the Features, you acknowledge and agree that:

* you are obtaining the applications or packages from such third parties and under separate license terms applicable to each application or package (including, with respect to the package-manager Features, any terms applicable to software dependencies that may be included in the package);

MICROSOFT MAKES NO REPRESENTATIONS, WARRANTIES OR GUARANTEES AS TO THE FEED OR GALLERY URL, ANY FEEDS OR GALLERIES FROM SUCH URL, THE INFORMATION CONTAINED THEREIN, OR ANY SOFTWARE APPLICATIONS OR PACKAGES REFERENCED IN OR ACCESSED BY YOU THROUGH SUCH FEEDS OR GALLERIES. MICROSOFT GRANTS YOU NO LICENSE RIGHTS FOR THIRD-PARTY SOFTWARE APPLICATIONS OR PACKAGES THAT ARE OBTAINED USING THE FEATURES.

8. SCOPE OF LICENSE. The software is licensed, not sold. This agreement only gives you some rights to use the software. Microsoft reserves all other rights. Unless applicable law gives you more rights despite this limitation, you may use the software only as expressly permitted in this agreement. In doing so, you must comply with any technical limitations in the software that only allow you to use it in certain ways. You may not

* disclose the results of any benchmark tests of the software to any third party without Microsoft's prior written approval, however, this does not apply to the Microsoft .NET Framework (see Section 5);
* work around any technical limitations in the software;
* reverse engineer, decompile or disassemble the software, or otherwise attempt to derive the source code for the software except, and solely to the extent: (i) permitted by applicable law, despite this limitation; or (ii) required to debug changes to any libraries licensed under the GNU Lesser General Public License which are included with and linked to by the software;
* make more copies of the software than specified in this agreement or allowed by applicable law, despite this limitation;
* publish the software for others to copy;
* rent, lease or lend the software;
* transfer the software or this agreement to any third party; or

* use the software for commercial software hosting services.

.

9. BACKUP COPY. You may make one backup copy of the software. You may use it only to reinstall the software.

10. DOCUMENTATION. Any person that has valid access to your computer or internal network may copy and use the documentation for your internal, reference purposes.

11. EXPORT RESTRICTIONS. The software is subject to United States export laws and regulations. You must comply with all domestic and international export laws and regulations that apply to the software. These laws include restrictions on destinations, end users and end use. For additional information, see www.microsoft.com/exporting .

12. SUPPORT SERVICES. Because this software is "as is," we may not provide support services for it.

13. ENTIRE AGREEMENT. This agreement, and the terms for supplements, updates, Internet-based services and support services that you use, are the entire agreement for the software and support services.

14. APPLICABLE LAW.

a. United States. If you acquired the software in the United States, Washington state law governs the interpretation of this agreement and applies to claims for breach of it, regardless of conflict of laws principles. The laws of the state where you live govern all other claims, including claims under state consumer protection laws, unfair competition laws, and in tort.

b. Outside the United States. If you acquired the software in any other country, the laws of that country apply.

15. LEGAL EFFECT. This agreement describes certain legal rights. You may have other rights under the laws of your country. You may also have rights with respect to the party from whom you acquired the software. This agreement does not change your rights under the laws of your country if the laws of your country do not permit it to do so.

16. DISCLAIMER OF WARRANTY. THE SOFTWARE IS LICENSED "AS-IS." YOU BEAR THE RISK OF USING IT. MICROSOFT GIVES NO EXPRESS WARRANTIES, GUARANTEES OR CONDITIONS. YOU MAY HAVE ADDITIONAL CONSUMER RIGHTS OR STATUTORY GUARANTEES UNDER YOUR LOCAL LAWS WHICH THIS AGREEMENT CANNOT CHANGE. TO THE EXTENT PERMITTED UNDER YOUR LOCAL LAWS, MICROSOFT EXCLUDES THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. FOR AUSTRALIA - YOU HAVE STATUTORY GUARANTEES UNDER THE AUSTRALIAN CONSUMER LAW AND NOTHING IN THESE TERMS IS INTENDED TO AFFECT THOSE RIGHTS.

17. LIMITATION ON AND EXCLUSION OF REMEDIES AND DAMAGES. YOU CAN RECOVER FROM MICROSOFT AND ITS SUPPLIERS ONLY DIRECT DAMAGES UP TO U.S. $5.00. YOU CANNOT RECOVER ANY OTHER DAMAGES, INCLUDING CONSEQUENTIAL, LOST PROFITS, SPECIAL, INDIRECT OR INCIDENTAL DAMAGES.

This limitation applies to

* anything related to the software, services, content (including code) on third party Internet sites, or third party programs; and
* claims for breach of contract, breach of warranty, guarantee or condition, strict liability, negligence, or other tort to the extent

permitted by applicable law.
It also applies even if Microsoft knew or should have known about the possibility of the damages. The above limitation or exclusion may not apply to you because your country may not allow the exclusion or limitation of incidental, consequential or other damages.

EULAID: VS2013_RTM_SS_EXPDesktop.1_ENU
Except where otherwise noted in the source code (e.g. the files hash.c, list.c and the trio files, which are covered by a similar licence but with different Copyright notices) all the files are:

Copyright (C) 1998-2003 Daniel Veillard. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE DANIEL VEILLARD BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of Daniel Veillard shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from him.
name,version,revision,url,license,copyright
SCOWL,7.1,,http://wordlist.aspell.net,Custom;https://raw.githubusercontent.com/en-wl/wordlist/scowl-7.1/scowl/Copyright,Copyright 2000-2011 by Kevin Atkinson
name,version,revision,url,license,copyright
Libxml2,2.9.4,,http://xmlsoft.org/,MIT,Copyright (C) 1998-2012 Daniel Veillard
Tencent is pleased to support the open source community by making RapidJSON available.

Copyright (C) 2015 THL A29 Limited, a Tencent company, and Milo Yip.
All rights reserved.

Terms of the MIT License:

Permission is hereby granted, free of charge, to any person obtaining a

copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, sublicense,
and/or sell copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE,
ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.
name,version,revision,url,license,copyright
Visual Studio 2013 Express Runtime Libraries,12.0,,https://www.visualstudio.com,Custom;https://s3-eu-west-1.amazonaws.com/prelert-artifacts/dependencies/License_Visual_Studio_2013_Express_English_b8a9dbc1-1fd4-4103-a83b-a2896f193ea0.htm,Copyright (C) Microsoft Corporation
The msinttypes r29
Copyright (c) 2006-2013 Alexander Chemeris
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

* Redistributions of source code must retain the above copyright notice,
 this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice,
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.
* Neither the name of copyright holder nor the names of its contributors may
 be used to endorse or promote products derived from this software without
 specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
THE POSSIBILITY OF SUCH DAMAGE.

The Moby lexicon project is complete and has
been place into the public domain. Use, sell,
rework, excerpt and use in any way on any platform.

Placing this material on internal or public servers is
also encouraged. The compiler is not aware of any
export restrictions so freely distribute world-wide.

You can verify the public domain status by contacting

Grady Ward
3449 Martha Ct.
Arcata, CA 95521-4884

daedal@myrealbox.com >>
Apache log4cxx
Copyright 2004-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).
The collective work is Copyright 2000-2011 by Kevin Atkinson as well
as any of the copyrights mentioned below:

Copyright 2000-2011 by Kevin Atkinson

Permission to use, copy, modify, distribute and sell these word
lists, the associated scripts, the output created from the scripts,
and its documentation for any purpose is hereby granted without fee,
provided that the above copyright notice appears in all copies and
that both that copyright notice and this permission notice appear in
supporting documentation. Kevin Atkinson makes no representations
about the suitability of this array for any purpose. It is provided
"as is" without express or implied warranty.

Alan Beale <biljir@pobox.com> also deserves special credit as he has,
in addition to providing the 12Dicts package and being a major
contributor to the ENABLE word list, given me an incredible amount of
feedback and created a number of special lists (those found in the
Supplement) in order to help improve the overall quality of SCOWL.

The 10 level includes the 1000 most common English words (according to
the Moby (TM) Words II [MWords] package), a subset of the 1000 most
common words on the Internet (again, according to Moby Words II), and
frequently class 16 from Brian Kelk's "UK English Wordlist
with Frequency Classification".

The MWords package was explicitly placed in the public domain:

The Moby lexicon project is complete and has
been place into the public domain. Use, sell,
rework, excerpt and use in any way on any platform.

Placing this material on internal or public servers is
also encouraged. The compiler is not aware of any
export restrictions so freely distribute world-wide.

You can verify the public domain status by contacting

Grady Ward
3449 Martha Ct.
Arcata, CA 95521-4884

grady@netcom.com
grady@northcoast.com

The "UK English Wordlist With Frequency Classification" is also in the Public Domain:

Date: Sat, 08 Jul 2000 20:27:21 +0100
From: Brian Kelk <Brian.Kelk@cl.cam.ac.uk>

> I was wondering what the copyright status of your "UK English
> Wordlist With Frequency Classification" word list as it seems to
> be lacking any copyright notice.

There were many many sources in total, but any text marked "copyright" was avoided. Locally-written documentation was one source. An earlier version of the list resided in a filespace called PUBLIC on the University mainframe, because it was considered public domain.

Date: Tue, 11 Jul 2000 19:31:34 +0100

> So are you saying your word list is also in the public domain?

That is the intention.

The 20 level includes frequency classes 7-15 from Brian's word list.

The 35 level includes frequency classes 2-6 and words appearing in at least 11 of 12 dictionaries as indicated in the 12Dicts package. All words from the 12Dicts package have had likely inflections added via my inflection database.

The 12Dicts package and Supplement is in the Public Domain.

The WordNet database, which was used in the creation of the Inflections database, is under the following copyright:

This software and database is being provided to you, the LICENSEE, by Princeton University under the following license. By obtaining, using and/or copying this software and database, you agree that you have read, understood, and will comply with these terms and conditions.:

Permission to use, copy, modify and distribute this software and database and its documentation for any purpose and without fee or royalty is hereby granted, provided that you agree to comply with the following copyright notice and statements, including the disclaimer, and that the same appear on ALL copies of the software, database and documentation, including modifications that you make for internal use or for distribution.

WordNet 1.6 Copyright 1997 by Princeton University. All rights reserved.

THIS SOFTWARE AND DATABASE IS PROVIDED "AS IS" AND PRINCETON UNIVERSITY MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PRINCETON UNIVERSITY MAKES NO REPRESENTATIONS OR WARRANTIES OF MERCHANT-ABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE LICENSED SOFTWARE, DATABASE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

The name of Princeton University or Princeton may not be used in advertising or publicity pertaining to distribution of the software and/or database. Title to copyright in this software, database and any associated documentation shall at all times remain with Princeton University and LICENSEE agrees to preserve same.

The 40 level includes words from Alan's 3esl list found in version 4.0 of his 12dicts package. Like his other stuff the 3esl list is also in the public domain.

The 50 level includes Brian's frequency class 1, words words appearing in at least 5 of 12 of the dictionaries as indicated in the 12Dicts package, and uppercase words in at least 4 of the previous 12 dictionaries. A decent number of proper names is also included: The top 1000 male, female, and Last names from the 1990 Census report; a list of names sent to me by Alan Beale; and a few names that I added myself. Finally a small list of abbreviations not commonly found in other word lists is included.

The name files form the Census report is a government document which I

don't think can be copyrighted.

The file special-jargon.50 uses common.lst and word.lst from the "Unofficial Jargon File Word Lists" which is derived from "The Jargon File". All of which is in the Public Domain. This file also contain a few extra UNIX terms which are found in the file "unix-terms" in the special/ directory.

The 55 level includes words from Alan's 2of4brif list found in version 4.0 of his 12dicts package. Like his other stuff the 2of4brif is also in the public domain.

The 60 level includes all words appearing in at least 2 of the 12 dictionaries as indicated by the 12Dicts package.

The 70 level includes Brian's frequency class 0 and the 74,550 common dictionary words from the MWords package. The common dictionary words, like those from the 12Dicts package, have had all likely inflections added. The 70 level also included the 5desk list from version 4.0 of the 12Dics package which is the public domain.

The 80 level includes the ENABLE word list, all the lists in the ENABLE supplement package (except for ABLE), the "UK Advanced Cryptics Dictionary" (UKACD), the list of signature words in from YAWL package, and the 10,196 places list from the MWords package.

The ENABLE package, mainted by M\Cooper <thegrendel@theriver.com>, is in the Public Domain:

The ENABLE master word list, WORD.LST, is herewith formally released into the Public Domain. Anyone is free to use it or distribute it in any manner they see fit. No fee or registration is required for its use nor are "contributions" solicited (if you feel you absolutely must contribute something for your own peace of mind, the authors of the ENABLE list ask that you make a donation on their behalf to your favorite charity). This word list is our gift to the Scrabble community, as an alternate to "official" word lists. Game designers may feel free to incorporate the WORD.LST into their games. Please mention the source and credit us as originators of the list. Note that if you, as a game designer, use the WORD.LST in your product, you may still copyright and protect your product, but you may *not* legally copyright or in any way restrict redistribution of the WORD.LST portion of your product. This *may* under law restrict your rights to restrict your users' rights, but that is only fair.

UKACD, by J Ross Beresford <ross@bryson.demon.co.uk>, is under the following copyright:

Copyright (c) J Ross Beresford 1993-1999. All Rights Reserved.

The following restriction is placed on the use of this publication:
if The UK Advanced Cryptics Dictionary is used in a software package
or redistributed in any form, the copyright notice must be
prominently displayed and the text of this document must be included
verbatim.

There are no other restrictions: I would like to see the list
distributed as widely as possible.

The 95 level includes the 354,984 single words, 256,772 compound words, 4,946 female names and the 3,897 male names, and 21,986 names from the MWords package, ABLE.LST from the ENABLE Supplement, and some additional words found in my part-of-speech database that were not found anywhere else.

Accent information was taken from UKACD.

My VARCON package was used to create the American, British, and Canadian word list.

Since the original word lists used used in the VARCON package came from the Ispell distribution they are under the Ispell copyright:

Copyright 1993, Geoff Kuenning, Granada Hills, CA
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. All modifications to the source code must be clearly marked as
   such. Binary redistributions based on modified source code
   must be clearly marked as modified versions in the documentation
   and/or other materials provided with the distribution.
(clause 4 removed with permission from Geoff Kuenning)
5. The name of Geoff Kuenning may not be used to endorse or promote
   products derived from this software without specific prior
   written permission.

THIS SOFTWARE IS PROVIDED BY GEOFF KUENNING AND CONTRIBUTORS ``AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT

LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL GEOFF KUENNING OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

name,version,revision,url,license,copyright

RapidJSON,1.1.0,f05edc9296507a9864d99931e203631c2ffd8d4a,http://rapidjson.org,MIT,"Copyright (C) 2015 THL A29 Limited, a Tencent company, and Milo Yip"

name,version,revision,url,license,copyright

Moby lexicon project,1,,http://icon.shef.ac.uk/Moby/,Public-Domain;http://icon.shef.ac.uk/Moby/,

Apache Portable Runtime

Copyright (c) 2009 The Apache Software Foundation.

This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign.

This software contains code derived from the RSA Data Security
Inc. MD5 Message-Digest Algorithm.

This software contains code derived from UNIX V7, Copyright(C)
Caldera International Inc.
name,version,revision,url,license,copyright
Apache Portable Runtime Utility Library,1.5.4,,https://apr.apache.org,Apache-2.0,Copyright (c) 2000-2014 The Apache Software Foundation
name,version,revision,url,license,copyright
Eigen,3.2.5,bdd17ee3b1b3a166cd5ec36dcad4fc1f3faf774a,http://eigen.tuxfamily.org,MPL-2.0,
name,version,revision,url,license,copyright
strptime,1.35,,http://cvsweb.netbsd.org/bsdweb.cgi/src/lib/libc/time/strptime.c,BSD-2-Clause-NetBSD,"Copyright (c) 1997, 1998, 2005, 2008 The NetBSD Foundation, Inc."
Of the two options provided, Elastic chooses to license RapidXml under the
MIT License.
name,version,revision,url,license,copyright
log4cxx,0.10.0,,https://logging.apache.org/log4cxx,Apache-2.0,Copyright 2004-2007 The Apache Software Foundation
name,version,revision,url,license,copyright
GCC Runtime Library,6.2.0,6ac74a62ba7258299cf85fbef9bf45333ddf10c0,https://gcc.gnu.org,GPL-3.0 WITH GCC-exception-3.1,"Copyright (C) 2016 Free Software Foundation, Inc."
name,version,revision,url,license,copyright
zlib,1.2.8,,http://zlib.net,Zlib,Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler
Copyright (c) 1997, 1998, 2005, 2008 The NetBSD Foundation, Inc.
All rights reserved.

This code was contributed to The NetBSD Foundation by Klaus Klein.
Heavily optimised by David Laight

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS
``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE

POSSIBILITY OF SUCH DAMAGE.
name,version,revision,url,license,copyright
The msinttypes,r29,,https://code.google.com/archive/p/msinttypes/,BSD-3-Clause,Copyright (c) 2006-2013 Alexander Chemeris
GCC RUNTIME LIBRARY EXCEPTION

Version 3.1, 31 March 2009

Copyright (C) 2009 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a notice placed by the copyright holder of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header files and runtime libraries with the compiled program. The purpose of this Exception is to allow compilation of non-GPL (including proprietary) programs to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for execution after a Compilation Process, or makes use of an interface provided by the Runtime Library, but is not otherwise based on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without modifications, governed by version 3 (or a specified later version) of the GNU General Public License (GPL) with the option of using any subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and use would permit combination with GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor architecture, in executable form or suitable for input to an assembler, loader, linker and/or execution phase. Notwithstanding that, Target Code does not include data in any format that is used as a compiler intermediate representation, or used for producing a compiler intermediate representation.

The "Compilation Process" transforms code entirely represented in non-intermediate languages designed for human-written code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use of source code generators and preprocessors need not be considered part of the Compilation Process, since the Compilation Process can be understood as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-compatible software, or if it is done without using any work based on GCC. For example, using non-GPL-compatible Software to optimize any GCC intermediate representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime Library with Independent Modules, even if such propagation would otherwise violate the terms of GPLv3, provided that all Target Code was generated by Eligible Compilation Processes. You may then convey such a combination under terms of your choice, consistent with the licensing of the Independent Modules.

2. No Weakening of GCC Copyleft.

The availability of this Exception does not imply any general presumption that third-party software is unaffected by the copyleft requirements of the license of GCC.
Mozilla Public License Version 2.0
==========

1. Definitions

1.1. "Contributor"
means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code

Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU

Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source

Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by

You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an   *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any           *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of    *
*  goodwill, work stoppage, computer failure or malfunction, or any    *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party's negligence to the       *
*  extent applicable law prohibits such limitation. Some               *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
*  limitation may not apply to You.                                    *
*                                                                      *
************************************************************************
```

8. Litigation

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that

jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as
defined by the Mozilla Public License, v. 2.0.
name,version,revision,url,license,copyright
Boost C++ Libraries,1.65.1,,http://www.boost.org,BSL-1.0,
name,version,revision,url,license,copyright
Apache Portable Runtime Library,1.5.2,,https://apr.apache.org,Apache-2.0,Copyright (c) 2000-2015 The Apache Software Foundation
This product includes software developed by
The Apache Software Foundation (http://www.apache.org/).

1.8 epel-release 7-11

1.9 erlang 20.1.7

1.9.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.10 Flask 1.0.2

1.10.1 Available under license :

Copyright (c) 2010 by Armin Ronacher.

Some rights reserved.

This logo or a modified version may be used by anyone to refer to the Flask project, but does not indicate endorsement by the project.

Redistribution and use in source (the SVG file) and binary forms (rendered PNG files etc.) of the image, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice and this list of conditions.

* The names of the contributors to the Flask software (see AUTHORS) may not be used to endorse or promote products derived from this software without specific prior written permission.

Note: we would appreciate that you make the image a link to http://flask.pocoo.org/ if you use it on a web page.
Copyright 2010 by the Pallets team.

Some rights reserved.

Redistribution and use in source and binary forms of the software as

well as documentation, with or without modification, are permitted
provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
 this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
 notice, this list of conditions and the following disclaimer in the
 documentation and/or other materials provided with the distribution.

* Neither the name of the copyright holder nor the names of its
 contributors may be used to endorse or promote products derived from
 this software without specific prior written permission.

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND
FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE AND DOCUMENTATION, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
License
=======

Flask is licensed under a three clause BSD License. It basically means:
do whatever you want with it as long as the copyright in Flask sticks
around, the conditions are not modified and the disclaimer is present.
Furthermore you must not use the names of the authors to promote derivatives
of the software without written consent.

The full license text can be found below (:ref:`flask-license`). For the
documentation and artwork different licenses apply.

.. _authors:

Authors

.. include:: ../AUTHORS

General License Definitions

The following section contains the full license texts for Flask and the documentation.

- "AUTHORS" hereby refers to all the authors listed in the :ref:`authors` section.

- The ":ref:`flask-license`" applies to all the source code shipped as part of Flask (Flask itself as well as the examples and the unittests) as well as documentation.

- The ":ref:`artwork-license`" applies to the project's Horn-Logo.

.. _flask-license:

Flask License

.. include:: ../LICENSE

.. _artwork-license:

Flask Artwork License

.. include:: ../artwork/LICENSE

1.11 gcc 4.8.5 :28.el7

1.11.1 Available under license :

MIT License (MIT)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
This is an attempt to acknowledge early contributions to the garbage
collector. Later contributions should instead be mentioned in
README.changes.

HISTORY -

Early versions of this collector were developed as a part of research
projects supported in part by the National Science Foundation
and the Defense Advance Research Projects Agency.

The garbage collector originated as part of the run-time system for
the Russell programming language implementation. The first version of the
garbage collector was written primarily by Al Demers. It was then refined
and mostly rewritten, primarily by Hans-J. Boehm, at Cornell U.,
the University of Washington, Rice University (where it was first used for
C and assembly code), Xerox PARC, SGI, and HP Labs. However, significant
contributions have also been made by many others.

Some other contributors:

More recent contributors are mentioned in the modification history in
README.changes. My apologies for any omissions.

The SPARC specific code was originally contributed by Mark Weiser.
The Encore Multimax modifications were supplied by
Kevin Kenny (kenny@m.cs.uiuc.edu). The adaptation to the IBM PC/RT is largely
due to Vernon Lee, on machines made available to Rice by IBM.
Much of the HP specific code and a number of good suggestions for improving the
generic code are due to Walter Underwood.
Robert Brazile (brazile@diamond.bbn.com) originally supplied the ULTRIX code.
Al Dosser (dosser@src.dec.com) and Regis Cridlig (Regis.Cridlig@cl.cam.ac.uk)
subsequently provided updates and information on variation between ULTRIX
systems. Parag Patel (parag@netcom.com) supplied the A/UX code.
Jesper Peterson(jep@mtiame.mtia.oz.au), Michel Schinz, and
Martin Tauchmann (martintauchmann@bigfoot.com) supplied the Amiga port.
Thomas Funke (thf@zelator.in-berlin.de(?)) and
Brian D.Carlstrom (bdc@clark.lcs.mit.edu) supplied the NeXT ports.
Douglas Steel (doug@wg.icl.co.uk) provided ICL DRS6000 code.
Bill Janssen (janssen@parc.xerox.com) supplied the SunOS dynamic loader
specific code. Manuel Serrano (serrano@cornas.inria.fr) supplied linux and
Sony News specific code. Al Dosser provided Alpha/OSF/1 code. He and
Dave Detlefs(detlefs@src.dec.com) also provided several generic bug fixes.
Alistair G. Crooks(agc@uts.amdahl.com) supplied the NetBSD and 386BSD ports.
Jeffrey Hsu (hsu@soda.berkeley.edu) provided the FreeBSD port.
Brent Benson (brent@jade.ssd.csd.harris.com) ported the collector to
a Motorola 88K processor running CX/UX (Harris NightHawk).

Ari Huttunen (Ari.Huttunen@hut.fi) generalized the OS/2 port to
nonIBM development environments (a nontrivial task).
Patrick Beard (beard@cs.ucdavis.edu) provided the initial MacOS port.
David Chase, then at Olivetti Research, suggested several improvements.
Scott Schwartz (schwartz@groucho.cse.psu.edu) supplied some of the
code to save and print call stacks for leak detection on a SPARC.
Jesse Hull and John Ellis supplied the C++ interface code.
Zhong Shao performed much of the experimentation that led to the
current typed allocation facility. (His dynamic type inference code hasn't
made it into the released version of the collector, yet.)
==========
compiler_rt License
==========

The compiler_rt library is dual licensed under both the University of Illinois
"BSD-Like" license and the MIT license. As a user of this code you may choose
to use it under either license. As a contributor, you agree to allow your code
to be used under both.

Full text of the relevant licenses is included below.

==========

University of Illinois/NCSA
Open Source License

Copyright (c) 2009-2012 by the contributors listed in CREDITS.TXT

All rights reserved.

Developed by:

LLVM Team

University of Illinois at Urbana-Champaign

http://llvm.org

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal with
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

* Redistributions of source code must retain the above copyright notice,
  this list of conditions and the following disclaimers.

* Redistributions in binary form must reproduce the above copyright notice,
  this list of conditions and the following disclaimers in the
  documentation and/or other materials provided with the distribution.

* Neither the names of the LLVM Team, University of Illinois at
  Urbana-Champaign, nor the names of its contributors may be used to
  endorse or promote products derived from this Software without specific
  prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.

==========

Copyright (c) 2009-2012 by the contributors listed in CREDITS.TXT

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

==========
Copyrights and Licenses for Third Party Software Distributed with LLVM:
==========
The LLVM software contains code written by third parties. Such software will
have its own individual LICENSE.TXT file in the directory in which it appears.
This file will describe the copyrights, license, and restrictions which apply
to that code.

The disclaimer of warranty in the University of Illinois Open Source License

applies to all code in the LLVM Distribution, and nothing in any of the other licenses gives permission to use the names of the LLVM Team or the University of Illinois to endorse or promote products derived from this Software.

The following pieces of software have additional or alternate copyrights, licenses, and/or restrictions:

Program	Directory
mach_override	lib/interception/mach_override

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public

License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of
this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from
a header file that is part of the Library. You may convey such object
code under terms of your choice, provided that, if the incorporated
material is not limited to numerical parameters, data structure
layouts and accessors, or small macros, inline functions and templates
(ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the
Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license
document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that,
taken together, effectively do not restrict modification of the
portions of the Library contained in the Combined Work and reverse
engineering for debugging such modifications, if you also do each of
the following:

a) Give prominent notice with each copy of the Combined Work that
the Library is used in it and that the Library and its use are
covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license
document.

c) For a Combined Work that displays copyright notices during
execution, include the copyright notice for the Library among
these notices, as well as a reference directing the user to the
copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this
License, and the Corresponding Application Code in a form
suitable for, and under terms that permit, the user to
recombine or relink the Application with a modified version of
the Linked Version to produce a modified Combined Work, in the
manner specified by section 6 of the GNU GPL for conveying

Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and

conditions either of that published version or of any later version
published by the Free Software Foundation. If the Library as you
received it does not specify a version number of the GNU Lesser
General Public License, you may choose any version of the GNU Lesser
General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide
whether future versions of the GNU Lesser General Public License shall
apply, that proxy's public statement of acceptance of any version is
permanent authorization for you to choose that version for the
Library.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.
These restrictions translate to certain responsibilities for you if you
distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights that
you have. You must make sure that they, too, receive or can get the
source code. And you must show them these terms so they know their
rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or

distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in

certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE

POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and

"recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component

(kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts,

regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a

fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal

Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and

finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of

rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or

arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF

SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it

```
   under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License.  Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

 You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

 The GNU General Public License does not permit incorporating your program
into proprietary programs.  If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library.  If this is what you want to do, use the GNU Lesser General
Public License instead of this License.  But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.
/* Definitions for BSD assembler syntax for Intel 386
  (actually AT&T syntax for insns and operands,
  adapted to BSD conventions for symbol names and debugging.)
  Copyright (C) 1988-2013 Free Software Foundation, Inc.

This file is part of GCC.

GCC is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 3, or (at your option)
any later version.

GCC is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with GCC; see the file COPYING3.  If not see
<http://www.gnu.org/licenses/>.  */

/* Use the Sequent Symmetry assembler syntax.  */

/* Define the syntax of pseudo-ops, labels and comments.  */

/* Prefix for internally generated assembler labels.  If we aren't using
  underscores, we are using prefix `.'s to identify labels that should
  be ignored, as in `i386/gas.h' --karl@cs.umb.edu  */

#define LPREFIX "L"
```

```
/* Assembler pseudos to introduce constants of various size.  */

#define ASM_BYTE "\t.byte\t"
#define ASM_SHORT "\t.word\t"
#define ASM_LONG "\t.long\t"
#define ASM_QUAD "\t.quad\t"  /* Should not be used for 32bit compilation.  */

/* This was suggested, but it shouldn't be right for DBX output. -- RMS
  #define ASM_OUTPUT_SOURCE_FILENAME(FILE, NAME) */


/* Define the syntax of labels and symbol definitions/declarations.  */

/* This is how to output an assembler line
  that says to advance the location counter by SIZE bytes.  */

#define ASM_OUTPUT_SKIP(FILE,SIZE)  \
 fprintf (FILE, "\t.space "HOST_WIDE_INT_PRINT_UNSIGNED"\n", (SIZE))

/* Define the syntax of labels and symbol definitions/declarations.  */

/* This says how to output an assembler line
  to define a global common symbol.  */

#define ASM_OUTPUT_COMMON(FILE, NAME, SIZE, ROUNDED)  \
( fputs (".comm ", (FILE)),   \
 assemble_name ((FILE), (NAME)),  \
 fprintf ((FILE), ",%u\n", (int)(ROUNDED)))

/* This says how to output an assembler line
  to define a local common symbol.  */

#define ASM_OUTPUT_LOCAL(FILE, NAME, SIZE, ROUNDED)  \
( fputs (".lcomm ", (FILE)),   \
 assemble_name ((FILE), (NAME)),  \
 fprintf ((FILE), ",%u\n", (int)(ROUNDED)))

#ifdef HAVE_GAS_LCOMM_WITH_ALIGNMENT
#define ASM_OUTPUT_ALIGNED_LOCAL(FILE, NAME, SIZE, ALIGNMENT)  \
( fputs (".lcomm ", (FILE)),   \
 assemble_name ((FILE), (NAME)),  \
 fprintf ((FILE), ",%u,%u\n", (int)(SIZE), (int)(ALIGNMENT) / BITS_PER_UNIT))
#endif

/* This is how to output an assembler line
  that says to advance the location counter
  to a multiple of 2**LOG bytes.  */
```

```
#define ASM_OUTPUT_ALIGN(FILE,LOG) \
 if ((LOG)!=0) fprintf ((FILE), "\t.align %d\n", (LOG))

/* This is how to store into the string BUF
  the symbol_ref name of an internal numbered label where
  PREFIX is the class of label and NUM is the number within the class.
  This is suitable for output with `assemble_name'.  */

#define ASM_GENERATE_INTERNAL_LABEL(BUF,PREFIX,NUMBER) \
   sprintf ((BUF), "*%s%ld", (PREFIX), (long)(NUMBER))

/* The prefix to add to user-visible assembler symbols.  */

#define USER_LABEL_PREFIX "_"

/* Sequent has some changes in the format of DBX symbols.  */
#define DBX_NO_XREFS 1

/* Don't split DBX symbols into continuations.  */
#define DBX_CONTIN_LENGTH 0
@ignore
@c Set file name and title for man page.
@setfilename gpl
@settitle GNU General Public License
@c man begin SEEALSO
gfdl(7), fsf-funding(7).
@c man end
@c man begin COPYRIGHT
Copyright @copyright{} 2007 Free Software Foundation, Inc.

Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.
@c man end
@end ignore
@node Copying
@c man begin DESCRIPTION
@unnumbered GNU General Public License
@center Version 3, 29 June 2007

@c This file is intended to be included in another file.

@display
Copyright @copyright{} 2007 Free Software Foundation, Inc. @url{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.
@end display
```

@heading Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for

individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

@heading TERMS AND CONDITIONS

@enumerate 0
@item Definitions.

``This License'' refers to version 3 of the GNU General Public License.

``Copyright'' also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

``The Program'' refers to any copyrightable work licensed under this License. Each licensee is addressed as ``you''. ``Licensees'' and ``recipients'' may be individuals or organizations.

To ``modify'' a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a ``modified version'' of the earlier work or a work ``based on'' the earlier work.

A ``covered work'' means either the unmodified Program or a work based on the Program.

To ``propagate'' a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To ``convey'' a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user

through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays ``Appropriate Legal Notices'' to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

@item Source Code.

The ``source code'' for a work means the preferred form of the work for making modifications to it. ``Object code'' means any non-source form of a work.

A ``Standard Interface'' means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The ``System Libraries'' of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A ``Major Component'', in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The ``Corresponding Source'' for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can

regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

@item Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

@item Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

@item Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

@item Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

@enumerate a
@item
The work must carry prominent notices stating that you modified it, and giving a relevant date.

@item
The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to ``keep intact all notices''.

@item
You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

@item
If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.
@end enumerate

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program,

in or on a volume of a storage or distribution medium, is called an
``aggregate'' if the compilation and its resulting copyright are not
used to limit the access or legal rights of the compilation's users
beyond what the individual works permit. Inclusion of a covered work
in an aggregate does not cause this License to apply to the other
parts of the aggregate.

@item Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of
sections 4 and 5, provided that you also convey the machine-readable
Corresponding Source under the terms of this License, in one of these
ways:

@enumerate a
@item
Convey the object code in, or embodied in, a physical product
(including a physical distribution medium), accompanied by the
Corresponding Source fixed on a durable physical medium customarily
used for software interchange.

@item
Convey the object code in, or embodied in, a physical product
(including a physical distribution medium), accompanied by a written
offer, valid for at least three years and valid for as long as you
offer spare parts or customer support for that product model, to give
anyone who possesses the object code either (1) a copy of the
Corresponding Source for all the software in the product that is
covered by this License, on a durable physical medium customarily used
for software interchange, for a price no more than your reasonable
cost of physically performing this conveying of source, or (2) access
to copy the Corresponding Source from a network server at no charge.

@item
Convey individual copies of the object code with a copy of the written
offer to provide the Corresponding Source. This alternative is
allowed only occasionally and noncommercially, and only if you
received the object code with such an offer, in accord with subsection
6b.

@item
Convey the object code by offering access from a designated place
(gratis or for a charge), and offer equivalent access to the
Corresponding Source in the same way through the same place at no
further charge. You need not require recipients to copy the
Corresponding Source along with the object code. If the place to copy
the object code is a network server, the Corresponding Source may be
on a different server (operated by you or a third party) that supports

equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

@item
Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

@end enumerate

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A ``User Product'' is either (1) a ``consumer product'', which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, ``normally used'' refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

``Installation Information'' for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install

modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

@item Additional Terms.

``Additional permissions'' are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

@enumerate a
@item
Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

@item
Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices

displayed by works containing it; or

@item
Prohibiting misrepresentation of the origin of that material, or
requiring that modified versions of such material be marked in
reasonable ways as different from the original version; or

@item
Limiting the use for publicity purposes of names of licensors or
authors of the material; or

@item
Declining to grant rights under trademark law for use of some trade
names, trademarks, or service marks; or

@item
Requiring indemnification of licensors and authors of that material by
anyone who conveys the material (or modified versions of it) with
contractual assumptions of liability to the recipient, for any
liability that these contractual assumptions directly impose on those
licensors and authors.
@end enumerate

All other non-permissive additional terms are considered ``further
restrictions'' within the meaning of section 10. If the Program as you
received it, or any part of it, contains a notice stating that it is
governed by this License along with a term that is a further
restriction, you may remove that term. If a license document contains
a further restriction but permits relicensing or conveying under this
License, you may add to a covered work material governed by the terms
of that license document, provided that the further restriction does
not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you
must place, in the relevant source files, a statement of the
additional terms that apply to those files, or a notice indicating
where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the
form of a separately written license, or stated as exceptions; the
above requirements apply either way.

@item Termination.

You may not propagate or modify a covered work except as expressly
provided under this License. Any attempt otherwise to propagate or
modify it is void, and will automatically terminate your rights under
this License (including any patent licenses granted under the third

paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

@item Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

@item Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An ``entity transaction'' is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if

the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

@item Patents.

A ``contributor'' is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's ``contributor version''.

A contributor's ``essential patent claims'' are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, ``control'' includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a ``patent license'' is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To ``grant'' such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. ``Knowingly relying'' means you have actual knowledge that, but for the patent license, your conveying the

covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is ``discriminatory'' if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

@item No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

@item Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed

under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

@item Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License ``or any later version'' applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

@item Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM ``AS IS'' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

@item Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING

WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

@item Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

@end enumerate

@heading END OF TERMS AND CONDITIONS

@heading How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the ``copyright'' line and a pointer to where the full notice is found.

@smallexample
@var{one line to give the program's name and a brief idea of what it does.}
Copyright (C) @var{year} @var{name of author}

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see @url{http://www.gnu.org/licenses/}.
@end smallexample

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

@smallexample
@var{program} Copyright (C) @var{year} @var{name of author}
This program comes with ABSOLUTELY NO WARRANTY; for details type @samp{show w}.
This is free software, and you are welcome to redistribute it
under certain conditions; type @samp{show c} for details.
@end smallexample

The hypothetical commands @samp{show w} and @samp{show c} should show
the appropriate parts of the General Public License. Of course, your
program's commands might be different; for a GUI interface, you would
use an ``about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a ``copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
@url{http://www.gnu.org/licenses/}.

The GNU General Public License does not permit incorporating your
program into proprietary programs. If your program is a subroutine
library, you may consider it more useful to permit linking proprietary
applications with the library. If this is what you want to do, use
the GNU Lesser General Public License instead of this License. But
first, please read @url{http://www.gnu.org/philosophy/why-not-lgpl.html}.
@c man end
@ignore
@c Set file name and title for man page.
@setfilename gfdl
@settitle GNU Free Documentation License
@c man begin SEEALSO
gpl(7), fsf-funding(7).
@c man end
@c man begin COPYRIGHT
Copyright @copyright{} 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.
@uref{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
@c This file is intended to be included within another document,
@c hence no sectioning command or @node.

```
@c man end
@end ignore
@c Special handling for inclusion in the install manual.
@ifset gfdlhtml
@ifnothtml
@comment node-name,     next,          previous, up
@node    GNU Free Documentation License, Concept Index, Old, Top
@end ifnothtml
@html
<h1 align="center">Installing GCC: GNU Free Documentation License</h1>
@end html
@ifnothtml
@unnumbered GNU Free Documentation License
@end ifnothtml
@end ifset
@c man begin DESCRIPTION
@ifclear gfdlhtml
@node GNU Free Documentation License
@unnumbered GNU Free Documentation License
@end ifclear

@cindex FDL, GNU Free Documentation License
@center Version 1.3, 3 November 2008

@display
Copyright @copyright{} 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc.
@uref{http://fsf.org/}

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.
@end display

@enumerate 0
@item
PREAMBLE

The purpose of this License is to make a manual, textbook, or other
functional and useful document @dfn{free} in the sense of freedom: to
assure everyone the effective freedom to copy and redistribute it,
with or without modifying it, either commercially or noncommercially.
Secondarily, this License preserves for the author and publisher a way
to get credit for their work, while not being considered responsible
for modifications made by others.

This License is a kind of ``copyleft'', which means that derivative
works of the document must themselves be free in the same sense.  It
complements the GNU General Public License, which is a copyleft
license designed for free software.
```

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

@item
APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The ``Document'', below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as ``you''. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A ``Modified Version'' of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A ``Secondary Section'' is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The ``Invariant Sections'' are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The ``Cover Texts'' are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may

be at most 5 words, and a Back-Cover Text may be at most 25 words.

A ``Transparent'' copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not ``Transparent'' is called ``Opaque''.

Examples of suitable formats for Transparent copies include plain @sc{ascii} without markup, Texinfo input format, La@TeX{} input format, @acronym{SGML} or @acronym{XML} using a publicly available @acronym{DTD}, and standard-conforming simple @acronym{HTML}, PostScript or @acronym{PDF} designed for human modification. Examples of transparent image formats include @acronym{PNG}, @acronym{XCF} and @acronym{JPG}. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, @acronym{SGML} or @acronym{XML} for which the @acronym{DTD} and/or processing tools are not generally available, and the machine-generated @acronym{HTML}, PostScript or @acronym{PDF} produced by some word processors for output purposes only.

The ``Title Page'' means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, ``Title Page'' means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The ``publisher'' means any person or entity that distributes copies of the Document to the public.

A section ``Entitled XYZ'' means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as ``Acknowledgements'', ``Dedications'', ``Endorsements'', or ``History''.) To ``Preserve the Title'' of such a section when you modify the Document means that it remains a section ``Entitled XYZ'' according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty

Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

@item
VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

@item
COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material.

If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

@item
MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

@enumerate A
@item
Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.

@item
List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.

@item
State on the Title page the name of the publisher of the Modified Version, as the publisher.

@item
Preserve all the copyright notices of the Document.

@item
Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.

@item

Include, immediately after the copyright notices, a license notice
giving the public permission to use the Modified Version under the
terms of this License, in the form shown in the Addendum below.

@item
Preserve in that license notice the full lists of Invariant Sections
and required Cover Texts given in the Document's license notice.

@item
Include an unaltered copy of this License.

@item
Preserve the section Entitled ``History'', Preserve its Title, and add
to it an item stating at least the title, year, new authors, and
publisher of the Modified Version as given on the Title Page. If
there is no section Entitled ``History'' in the Document, create one
stating the title, year, authors, and publisher of the Document as
given on its Title Page, then add an item describing the Modified
Version as stated in the previous sentence.

@item
Preserve the network location, if any, given in the Document for
public access to a Transparent copy of the Document, and likewise
the network locations given in the Document for previous versions
it was based on. These may be placed in the ``History'' section.
You may omit a network location for a work that was published at
least four years before the Document itself, or if the original
publisher of the version it refers to gives permission.

@item
For any section Entitled ``Acknowledgements'' or ``Dedications'', Preserve
the Title of the section, and preserve in the section all the
substance and tone of each of the contributor acknowledgements and/or
dedications given therein.

@item
Preserve all the Invariant Sections of the Document,
unaltered in their text and in their titles. Section numbers
or the equivalent are not considered part of the section titles.

@item
Delete any section Entitled ``Endorsements''. Such a section
may not be included in the Modified Version.

@item
Do not retitle any existing section to be Entitled ``Endorsements'' or
to conflict in title with any Invariant Section.

@item
Preserve any Warranty Disclaimers.
@end enumerate

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled ``Endorsements'', provided it contains nothing but endorsements of your Modified Version by various parties---for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

@item
COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of

Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled ``History" in the various original documents, forming one section Entitled ``History"; likewise combine any sections Entitled ``Acknowledgements", and any sections Entitled ``Dedications". You must delete all sections Entitled ``Endorsements."

@item
COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

@item
AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an ``aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

@item
TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special

permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled ``Acknowledgements'', ``Dedications'', or ``History'', the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

@item
TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

@item
FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new

versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See @uref{http://www.gnu.org/copyleft/}.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License ``or any later version'' applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

@item
RELICENSING

``Massive Multiauthor Collaboration Site'' (or ``MMC Site'') means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A ``Massive Multiauthor Collaboration'' (or ``MMC'') contained in the site means any set of copyrightable works thus published on the MMC site.

``CC-BY-SA'' means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

``Incorporate'' means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is ``eligible for relicensing'' if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

@end enumerate

@page
@unnumberedsec ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of
the License in the document and put the following copyright and
license notices just after the title page:

@smallexample
@group
 Copyright (C) @var{year} @var{your name}.
 Permission is granted to copy, distribute and/or modify this document
 under the terms of the GNU Free Documentation License, Version 1.3
 or any later version published by the Free Software Foundation;
 with no Invariant Sections, no Front-Cover Texts, and no Back-Cover
 Texts. A copy of the license is included in the section entitled ``GNU
 Free Documentation License''.
@end group
@end smallexample

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts,
replace the ``with...Texts.'' line with this:

@smallexample
@group
   with the Invariant Sections being @var{list their titles}, with
   the Front-Cover Texts being @var{list}, and with the Back-Cover Texts
   being @var{list}.
@end group
@end smallexample

If you have Invariant Sections without Cover Texts, or some other
combination of the three, merge those two alternatives to suit the
situation.

If your document contains nontrivial examples of program code, we
recommend releasing these examples in parallel under your choice of
free software license, such as the GNU General Public License,
to permit their use in free software.

@c Local Variables:
@c ispell-local-pdict: "ispell-dict"
@c End:

@c man end
   GNU LESSER GENERAL PUBLIC LICENSE
       Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in

non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does

and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based

on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and

therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the

user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any

particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"><html
xmlns="http://www.w3.org/1999/xhtml"><head><meta http-equiv="Content-Type" content="text/html;
charset=UTF-8" /><title>License</title><meta name="generator" content="DocBook XSL-NS Stylesheets
V1.78.1" /><meta name="keywords" content="ISO C++, library" /><meta name="keywords" content="ISO C++,
runtime, library" /><link rel="home" href="../index.html" title="The GNU C++ Library" /><link rel="up"
href="status.html" title="Chapter1.Status" /><link rel="prev" href="status.html" title="Chapter1.Status" /><link
rel="next" href="bugs.html" title="Bugs" /></head><body> <table width="100%"
summary="Navigation header"><tr><th colspan="3" align="center">License</th></tr><tr><td width="20%"
align="left">Prev</td><th width="60%"
align="center">Chapter1.Status</th><td width="20%" align="right"><a accesskey="n"
href="bugs.html">Next</td></tr></table><hr /> <div
class="titlepage"> <h2 class="title" style="clear: both"><a
id="manual.intro.status.license">License</h2>
There are two licenses affecting GNU libstdc++: one for the code,
and one for the documentation.

There is a license section in the FAQ regarding common <a class="link"
href="../faq.html#faq.license">questions. If you have more
questions, ask the FSF or the gcc mailing
list.
<h3 class="title"><a
id="manual.intro.status.license.gpl">The Code: GPL</h3>
The source code is distributed under the <a class="link" href="appendix_gpl.html" title="AppendixD. GNU
General Public License version 3">GNU General Public License version 3,
with the addition under section 7 of an exception described in
the GCC Runtime Library Exception, version 3.1
as follows (or see the file COPYING.RUNTIME):

GCCRUNTIMELIBRARYEXCEPTION

Version3.1,31March2009

Copyright(C)2009FreeSoftwareFoundation,Inc.

Everyoneispermittedtocopyanddistributeverbatimcopiesofthis
licensedocument,butchangingitisnotallowed.

ThisGCCRuntimeLibraryException("Exception")isanadditional
permissionundersection7oftheGNUGeneralPublicLicense,version
3("GPLv3").Itappliestoagivenfile(the"RuntimeLibrary")that
bearsanoticeplacedbythecopyrightholderofthefilestatingthat
thefileisgovernedbyGPLv3alongwiththisException.

WhenyouuseGCCtocompileaprogram,GCCmaycombineportionsof
certainGCCheaderfilesandruntimelibrarieswiththecompiled
program.ThepurposeofthisExceptionistoallowcompilationof
non-GPL(includingproprietary)programstouse,inthisway,the
headerfilesandruntimelibrariescoveredbythisException.

0.Definitions.

Afileisan"IndependentModule"ifiteitherrequirestheRuntime
LibraryforexecutionafteraCompilationProcess,ormakesuseofan
interfaceprovidedbytheRuntimeLibrary,butisnototherwisebased
ontheRuntimeLibrary.

"GCC"meansaversionoftheGNUCompilerCollection,withorwithout
modifications,governedbyversion3(oraspecifiedlaterversion)of
theGNUGeneralPublicLicense(GPL)withtheoptionofusingany
subsequentversionspublishedbytheFSF.

"GPL-compatibleSoftware"issoftwarewhoseconditionsofpropagation,
modificationandusewouldpermitcombinationwithGCCinaccordwith
thelicenseofGCC.

"TargetCode"referstooutputfromanycompilerforarealorvirtual
targetprocessorarchitecture,inexecutableformorsuitablefor
inputtoanassembler,loader,linkerand/orexecution
phase.Notwithstandingthat,TargetCodedoesnotincludedatainany
formatthatisusedasacompilerintermediaterepresentation,orused
forproducingacompilerintermediaterepresentation.

The"CompilationProcess"transformscodeentirelyrepresentedin
non-intermediatelanguagesdesignedforhuman-writtencode,and/orin
JavaVirtualMachinebytecode,intoTargetCode.Thus,forexample,

useofsourcecodegeneratorsandpreprocessorsneednotbeconsidered
partoftheCompilationProcess,sincetheCompilationProcesscanbe
understoodasstartingwiththeoutputofthegeneratorsor
preprocessors.

ACompilationProcessis"Eligible"ifitisdoneusingGCC,aloneor
withotherGPL-compatiblesoftware,orifitisdonewithoutusingany
workbasedonGCC.Forexample,usingnon-GPL-compatibleSoftwareto
optimizeanyGCCintermediaterepresentationswouldnotqualifyasan
EligibleCompilationProcess.

1.GrantofAdditionalPermission.

YouhavepermissiontopropagateaworkofTargetCodeformedby
combiningtheRuntimeLibrarywithIndependentModules,evenifsuch
propagationwouldotherwiseviolatethetermsofGPLv3,providedthat
allTargetCodewasgeneratedbyEligibleCompilationProcesses.You
maythenconveysuchacombinationundertermsofyourchoice,
consistentwiththelicensingoftheIndependentModules.

2.NoWeakeningofGCCCopyleft.

TheavailabilityofthisExceptiondoesnotimplyanygeneral
presumptionthatthird-partysoftwareisunaffectedbythecopyleft
requirementsofthelicenseofGCC.

Hopefully that text is self-explanatory. If it isn't, you need to speak
to your lawyer, or the Free Software Foundation.
<h3 class="title"><a
id="manual.intro.status.license.fdl">The Documentation: GPL, FDL</h3>
The documentation shipped with the library and made available over
the web, excluding the pages generated from source comments, are
copyrighted by the Free Software Foundation, and placed under the
 GNU Free
Documentation
License version 1.3. There are no Front-Cover Texts, no
Back-Cover Texts, and no Invariant Sections.

For documentation generated by doxygen or other automated tools
via processing source code comments and markup, the original source
code license applies to the generated files. Thus, the doxygen
documents are licensed <a class="link" href="appendix_gpl.html" title="AppendixD. GNU General Public
License version 3">GPL.

If you plan on making copies of the documentation, please let us know.
We can probably offer suggestions.
<hr /><table width="100%" summary="Navigation footer"><tr><td
width="40%" align="left">Prev</td><td width="20%"

align="center">Up</td><td width="40%" align="right">Next</td></tr><tr><td width="40%" align="left" valign="top">Chapter1.Status</td><td width="20%" align="center">Home</td><td width="40%" align="right" valign="top">Bugs</td></tr></table> </body></html>
<appendix xmlns="http://docbook.org/ns/docbook" version="5.0"
  xml:id="appendix.gpl-3.0"><info><title>
  <acronym>GNU</acronym> General Public License version 3
 </title></info>
 <?dbhtml filename="appendix_gpl.html"?>

 <para>
  Version 3, 29 June 2007
 </para>
 <para>
  Copyright 2007 Free Software Foundation, Inc.
  <link xmlns:xlink="http://www.w3.org/1999/xlink" xlink:href="http://www.fsf.org/">http://www.fsf.org/</link>
 </para>
 <para>
  Everyone is permitted to copy and distribute verbatim copies of this license
  document, but changing it is not allowed.
 </para>
 <bridgehead xml:id="gpl-3-preamble" renderas="sect1">
  Preamble
 </bridgehead>
 <para>
  The <acronym>GNU</acronym> General Public License is a free, copyleft
  license for software and other kinds of works.
 </para>
 <para>
  The licenses for most software and other practical works are designed to
  take away your freedom to share and change the works. By contrast, the
  <acronym>GNU</acronym> General Public License is intended to guarantee your
  freedom to share and change all versions of a programto make sure it
  remains free software for all its users. We, the Free Software Foundation,
  use the <acronym>GNU</acronym> General Public License for most of our
  software; it applies also to any other work released this way by its
  authors. You can apply it to your programs, too.
 </para>
 <para>
  When we speak of free software, we are referring to freedom, not price. Our
  General Public Licenses are designed to make sure that you have the freedom
  to distribute copies of free software (and charge for them if you wish),
  that you receive source code or can get it if you want it, that you can
  change the software or use pieces of it in new free programs, and that you
  know you can do these things.
 </para>
 <para>

To protect your rights, we need to prevent others from denying you these
rights or asking you to surrender the rights. Therefore, you have certain
responsibilities if you distribute copies of the software, or if you modify
it: responsibilities to respect the freedom of others.
</para>
<para>
For example, if you distribute copies of such a program, whether gratis or
for a fee, you must pass on to the recipients the same freedoms that you
received. You must make sure that they, too, receive or can get the source
code. And you must show them these terms so they know their rights.
</para>
<para>
Developers that use the <acronym>GNU</acronym> <acronym>GPL</acronym>
protect your rights with two steps: (1) assert copyright on the software,
and (2) offer you this License giving you legal permission to copy,
distribute and/or modify it.
</para>
<para>
For the developers and authors protection, the
<acronym>GPL</acronym> clearly explains that there is no warranty for this
free software. For both users and authors sake, the
<acronym>GPL</acronym> requires that modified versions be marked as changed,
so that their problems will not be attributed erroneously to authors of
previous versions.
</para>
<para>
Some devices are designed to deny users access to install or run modified
versions of the software inside them, although the manufacturer can do so.
This is fundamentally incompatible with the aim of protecting users
freedom to change the software. The systematic pattern of such abuse occurs
in the area of products for individuals to use, which is precisely where it
is most unacceptable. Therefore, we have designed this version of the
<acronym>GPL</acronym> to prohibit the practice for those products. If such
problems arise substantially in other domains, we stand ready to extend this
provision to those domains in future versions of the <acronym>GPL</acronym>,
as needed to protect the freedom of users.
</para>
<para>
Finally, every program is threatened constantly by software patents. States
should not allow patents to restrict development and use of software on
general-purpose computers, but in those that do, we wish to avoid the
special danger that patents applied to a free program could make it
effectively proprietary. To prevent this, the <acronym>GPL</acronym>
assures that patents cannot be used to render the program non-free.
</para>
<para>
The precise terms and conditions for copying, distribution and modification
follow.

</para>
<bridgehead>
TERMS AND CONDITIONS
</bridgehead>
<bridgehead xml:id="gpl-3-definitions" renderas="sect1">
0. Definitions.
</bridgehead>
<para>
This License refers to version 3 of the <acronym>GNU</acronym>
General Public License.
</para>
<para>
Copyright also means copyright-like laws that apply to other
kinds of works, such as semiconductor masks.
</para>
<para>
The Program refers to any copyrightable work licensed under
this License. Each licensee is addressed as you.
Licensees and recipients may be individuals or
organizations.
</para>
<para>
To modify a work means to copy from or adapt all or part of
the work in a fashion requiring copyright permission, other than the making
of an exact copy. The resulting work is called a modified
version of the earlier work or a work based on the
earlier work.
</para>
<para>
A covered work means either the unmodified Program or a work
based on the Program.
</para>
<para>
To propagate a work means to do anything with it that, without
permission, would make you directly or secondarily liable for infringement
under applicable copyright law, except executing it on a computer or
modifying a private copy. Propagation includes copying, distribution (with
or without modification), making available to the public, and in some
countries other activities as well.
</para>
<para>
To convey a work means any kind of propagation that enables
other parties to make or receive copies. Mere interaction with a user
through a computer network, with no transfer of a copy, is not conveying.
</para>
<para>
An interactive user interface displays Appropriate Legal
Notices to the extent that it includes a convenient and prominently

visible feature that (1) displays an appropriate copyright notice, and (2)
tells the user that there is no warranty for the work (except to the extent
that warranties are provided), that licensees may convey the work under this
License, and how to view a copy of this License. If the interface presents
a list of user commands or options, such as a menu, a prominent item in the
list meets this criterion.
</para>
<bridgehead xml:id="SourceCode" renderas="sect1">
1. Source Code.
</bridgehead>
<para>
The source code for a work means the preferred form of the
work for making modifications to it. Object code means any
non-source form of a work.
</para>
<para>
A Standard Interface means an interface that either is an
official standard defined by a recognized standards body, or, in the case of
interfaces specified for a particular programming language, one that is
widely used among developers working in that language.
</para>
<para>
The System Libraries of an executable work include anything,
other than the work as a whole, that (a) is included in the normal form of
packaging a Major Component, but which is not part of that Major Component,
and (b) serves only to enable use of the work with that Major Component, or
to implement a Standard Interface for which an implementation is available
to the public in source code form. A Major Component, in this
context, means a major essential component (kernel, window system, and so
on) of the specific operating system (if any) on which the executable work
runs, or a compiler used to produce the work, or an object code interpreter
used to run it.
</para>
<para>
The Corresponding Source for a work in object code form means
all the source code needed to generate, install, and (for an executable
work) run the object code and to modify the work, including scripts to
control those activities. However, it does not include the works
System Libraries, or general-purpose tools or generally available free
programs which are used unmodified in performing those activities but which
are not part of the work. For example, Corresponding Source includes
interface definition files associated with source files for the work, and
the source code for shared libraries and dynamically linked subprograms that
the work is specifically designed to require, such as by intimate data
communication or control flow between those subprograms and other parts of
the work.
</para>
<para>

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.
</para>
<para>
The Corresponding Source for a work in source code form is that same work.
</para>
<bridgehead xml:id="BasicPermissions" renderas="sect1">
2. Basic Permissions.
</bridgehead>
<para>
All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.
</para>
<para>
You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.
</para>
<para>
Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.
</para>
<bridgehead xml:id="Protecting" renderas="sect1">
3. Protecting Users Legal Rights From Anti-Circumvention Law.
</bridgehead>
<para>
No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.
</para>
<para>
When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered

work, and you disclaim any intention to limit operation or modification of
the work as a means of enforcing, against the works users, your or
third parties legal rights to forbid circumvention of technological
measures.
</para>
<bridgehead xml:id="ConveyingVerbatim" renderas="sect1">
4. Conveying Verbatim Copies.
</bridgehead>
<para>
You may convey verbatim copies of the Programs source code as you
receive it, in any medium, provided that you conspicuously and appropriately
publish on each copy an appropriate copyright notice; keep intact all
notices stating that this License and any non-permissive terms added in
accord with section 7 apply to the code; keep intact all notices of the
absence of any warranty; and give all recipients a copy of this License
along with the Program.
</para>
<para>
You may charge any price or no price for each copy that you convey, and you
may offer support or warranty protection for a fee.
</para>
<bridgehead xml:id="ConveyingModified" renderas="sect1">
5. Conveying Modified Source Versions.
</bridgehead>
<para>
You may convey a work based on the Program, or the modifications to produce
it from the Program, in the form of source code under the terms of section
4, provided that you also meet all of these conditions:
</para>
<orderedlist numeration="loweralpha" inheritnum="ignore" continuation="restarts">
<listitem>
<para>
The work must carry prominent notices stating that you modified it, and
giving a relevant date.
</para>
</listitem>
<listitem>
<para>
The work must carry prominent notices stating that it is released under
this License and any conditions added under section 7. This requirement
modifies the requirement in section 4 to keep intact all
notices.
</para>
</listitem>
<listitem>
<para>
You must license the entire work, as a whole, under this License to
anyone who comes into possession of a copy. This License will therefore

```
      apply, along with any applicable section 7 additional terms, to the
      whole of the work, and all its parts, regardless of how they are
      packaged.  This License gives no permission to license the work in any
      other way, but it does not invalidate such permission if you have
      separately received it.
    </para>
  </listitem>
  <listitem>
    <para>
      If the work has interactive user interfaces, each must display
      Appropriate Legal Notices; however, if the Program has interactive
      interfaces that do not display Appropriate Legal Notices, your work need
      not make them do so.
    </para>
  </listitem>
</orderedlist>
<para>
  A compilation of a covered work with other separate and independent works,
  which are not by their nature extensions of the covered work, and which are
  not combined with it such as to form a larger program, in or on a volume of
  a storage or distribution medium, is called an aggregate if
  the compilation and its resulting copyright are not used to limit the access
  or legal rights of the compilations users beyond what the individual works
  permit.  Inclusion of a covered work in an aggregate does not cause
  this License to apply to the other parts of the aggregate.
</para>
<bridgehead xml:id="ConveyingNonSource" renderas="sect1">
  6. Conveying Non-Source Forms.
</bridgehead>
<para>
  You may convey a covered work in object code form under the terms of
  sections 4 and 5, provided that you also convey the machine-readable
  Corresponding Source under the terms of this License, in one of these ways:
</para>
<orderedlist numeration="loweralpha" inheritnum="ignore" continuation="restarts">
  <listitem>
    <para>
      Convey the object code in, or embodied in, a physical product (including
      a physical distribution medium), accompanied by the Corresponding Source
      fixed on a durable physical medium customarily used for software
      interchange.
    </para>
  </listitem>
  <listitem>
    <para>
      Convey the object code in, or embodied in, a physical product (including
      a physical distribution medium), accompanied by a written offer, valid
      for at least three years and valid for as long as you offer spare parts
```

or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

</para>

</listitem>

<listitem>

<para>

Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

</para>

</listitem>

<listitem>

<para>

Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

</para>

</listitem>

<listitem>

<para>

Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

</para>

</listitem>

</orderedlist>

<para>

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

</para>

<para>

A User Product is either (1) a consumer product,
which means any tangible personal property which is normally used for
personal, family, or household purposes, or (2) anything designed or sold
for incorporation into a dwelling. In determining whether a product is a
consumer product, doubtful cases shall be resolved in favor of coverage.
For a particular product received by a particular user, normally
used refers to a typical or common use of that class of product,
regardless of the status of the particular user or of the way in which the
particular user actually uses, or expects or is expected to use, the
product. A product is a consumer product regardless of whether the product
has substantial commercial, industrial or non-consumer uses, unless such
uses represent the only significant mode of use of the product.
</para>
<para>
Installation Information for a User Product means any methods,
procedures, authorization keys, or other information required to install and
execute modified versions of a covered work in that User Product from a
modified version of its Corresponding Source. The information must suffice
to ensure that the continued functioning of the modified object code is in
no case prevented or interfered with solely because modification has been
made.
</para>
<para>
If you convey an object code work under this section in, or with, or
specifically for use in, a User Product, and the conveying occurs as part of
a transaction in which the right of possession and use of the User Product
is transferred to the recipient in perpetuity or for a fixed term
(regardless of how the transaction is characterized), the Corresponding
Source conveyed under this section must be accompanied by the Installation
Information. But this requirement does not apply if neither you nor any
third party retains the ability to install modified object code on the User
Product (for example, the work has been installed in
<acronym>ROM</acronym>).
</para>
<para>
The requirement to provide Installation Information does not include a
requirement to continue to provide support service, warranty, or updates for
a work that has been modified or installed by the recipient, or for the User
Product in which it has been modified or installed. Access to a network may
be denied when the modification itself materially and adversely affects the
operation of the network or violates the rules and protocols for
communication across the network.
</para>
<para>
Corresponding Source conveyed, and Installation Information provided, in
accord with this section must be in a format that is publicly documented
(and with an implementation available to the public in source code form),
and must require no special password or key for unpacking, reading or

```
 copying.
</para>
<bridgehead xml:id="AdditionalTerms" renderas="sect1">
  7. Additional Terms.
</bridgehead>
<para>
  Additional permissions are terms that supplement the terms of
  this License by making exceptions from one or more of its conditions.
  Additional permissions that are applicable to the entire Program shall be
  treated as though they were included in this License, to the extent that
  they are valid under applicable law.  If additional permissions apply only
  to part of the Program, that part may be used separately under those
  permissions, but the entire Program remains governed by this License
  without regard to the additional permissions.
</para>
<para>
  When you convey a copy of a covered work, you may at your option remove any
  additional permissions from that copy, or from any part of it.  (Additional
  permissions may be written to require their own removal in certain cases
  when you modify the work.)  You may place additional permissions on
  material, added by you to a covered work, for which you have or can give
  appropriate copyright permission.
</para>
<para>
  Notwithstanding any other provision of this License, for material you add
  to a covered work, you may (if authorized by the copyright holders of that
  material) supplement the terms of this License with terms:
</para>
<orderedlist numeration="loweralpha" inheritnum="ignore" continuation="restarts">
  <listitem>
    <para>
      Disclaiming warranty or limiting liability differently from the terms
      of sections 15 and 16 of this License; or
    </para>
  </listitem>
  <listitem>
    <para>
      Requiring preservation of specified reasonable legal notices or author
      attributions in that material or in the Appropriate Legal Notices
      displayed by works containing it; or
    </para>
  </listitem>
  <listitem>
    <para>
      Prohibiting misrepresentation of the origin of that material, or
      requiring that modified versions of such material be marked in
      reasonable ways as different from the original version; or
    </para>
```

```
  </listitem>
  <listitem>
    <para>
      Limiting the use for publicity purposes of names of licensors or
      authors of the material; or
    </para>
  </listitem>
  <listitem>
    <para>
      Declining to grant rights under trademark law for use of some trade
      names, trademarks, or service marks; or
    </para>
  </listitem>
  <listitem>
    <para>
      Requiring indemnification of licensors and authors of that material by
      anyone who conveys the material (or modified versions of it) with
      contractual assumptions of liability to the recipient, for any
      liability that these contractual assumptions directly impose on those
      licensors and authors.
    </para>
  </listitem>
</orderedlist>
<para>
  All other non-permissive additional terms are considered further
  restrictions within the meaning of section 10.  If the Program as
  you received it, or any part of it, contains a notice stating that it is
  governed by this License along with a term that is a further restriction,
  you may remove that term.  If a license document contains a further
  restriction but permits relicensing or conveying under this License, you
  may add to a covered work material governed by the terms of that license
  document, provided that the further restriction does not survive such
  relicensing or conveying.
</para>
<para>
  If you add terms to a covered work in accord with this section, you must
  place, in the relevant source files, a statement of the additional terms
  that apply to those files, or a notice indicating where to find the
  applicable terms.
</para>
<para>
  Additional terms, permissive or non-permissive, may be stated in the form
  of a separately written license, or stated as exceptions; the above
  requirements apply either way.
</para>
<bridgehead xml:id="gpl-3-termination" renderas="sect1">
  8. Termination.
</bridgehead>
```

<para>
You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).
</para>
<para>
However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.
</para>
<para>
Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.
</para>
<para>
Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.
</para>
<bridgehead xml:id="AcceptanceNotRequired" renderas="sect1">
9. Acceptance Not Required for Having Copies.
</bridgehead>
<para>
You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.
</para>
<bridgehead xml:id="AutomaticDownstream" renderas="sect1">
10. Automatic Licensing of Downstream Recipients.
</bridgehead>
<para>
Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing

compliance by third parties with this License.
</para>
<para>
An entity transaction is a transaction transferring control
of an organization, or substantially all assets of one, or subdividing an
organization, or merging organizations. If propagation of a covered work
results from an entity transaction, each party to that transaction who
receives a copy of the work also receives whatever licenses to the work the
partys predecessor in interest had or could give under the previous
paragraph, plus a right to possession of the Corresponding Source of the
work from the predecessor in interest, if the predecessor has it or can get
it with reasonable efforts.
</para>
<para>
You may not impose any further restrictions on the exercise of the rights
granted or affirmed under this License. For example, you may not impose a
license fee, royalty, or other charge for exercise of rights granted under
this License, and you may not initiate litigation (including a cross-claim
or counterclaim in a lawsuit) alleging that any patent claim is infringed
by making, using, selling, offering for sale, or importing the Program or
any portion of it.
</para>
<bridgehead xml:id="Patents" renderas="sect1">
11. Patents.
</bridgehead>
<para>
A contributor is a copyright holder who authorizes use under
this License of the Program or a work on which the Program is based. The
work thus licensed is called the contributors contributor
version.
</para>
<para>
A contributors essential patent claims are all patent
claims owned or controlled by the contributor, whether already acquired or
hereafter acquired, that would be infringed by some manner, permitted by
this License, of making, using, or selling its contributor version, but do
not include claims that would be infringed only as a consequence of further
modification of the contributor version. For purposes of this definition,
control includes the right to grant patent sublicenses in a
manner consistent with the requirements of this License.
</para>
<para>
Each contributor grants you a non-exclusive, worldwide, royalty-free patent
license under the contributors essential patent claims, to make, use,
sell, offer for sale, import and otherwise run, modify and propagate the
contents of its contributor version.
</para>
<para>

In the following three paragraphs, a patent license is any
express agreement or commitment, however denominated, not to enforce a
patent (such as an express permission to practice a patent or covenant not
to sue for patent infringement). To grant such a patent
license to a party means to make such an agreement or commitment not to
enforce a patent against the party.
</para>
<para>
If you convey a covered work, knowingly relying on a patent license, and the
Corresponding Source of the work is not available for anyone to copy, free
of charge and under the terms of this License, through a publicly available
network server or other readily accessible means, then you must either (1)
cause the Corresponding Source to be so available, or (2) arrange to deprive
yourself of the benefit of the patent license for this particular work, or
(3) arrange, in a manner consistent with the requirements of this License,
to extend the patent license to downstream recipients. Knowingly
relying means you have actual knowledge that, but for the patent
license, your conveying the covered work in a country, or your
recipients use of the covered work in a country, would infringe one
or more identifiable patents in that country that you have reason to believe
are valid.
</para>
<para>
If, pursuant to or in connection with a single transaction or arrangement,
you convey, or propagate by procuring conveyance of, a covered work, and
grant a patent license to some of the parties receiving the covered work
authorizing them to use, propagate, modify or convey a specific copy of the
covered work, then the patent license you grant is automatically extended to
all recipients of the covered work and works based on it.
</para>
<para>
A patent license is discriminatory if it does not include
within the scope of its coverage, prohibits the exercise of, or is
conditioned on the non-exercise of one or more of the rights that are
specifically granted under this License. You may not convey a covered work
if you are a party to an arrangement with a third party that is in the
business of distributing software, under which you make payment to the third
party based on the extent of your activity of conveying the work, and under
which the third party grants, to any of the parties who would receive the
covered work from you, a discriminatory patent license (a) in connection
with copies of the covered work conveyed by you (or copies made from those
copies), or (b) primarily for and in connection with specific products or
compilations that contain the covered work, unless you entered into that
arrangement, or that patent license was granted, prior to 28 March 2007.
</para>
<para>
Nothing in this License shall be construed as excluding or limiting any
implied license or other defenses to infringement that may otherwise be

available to you under applicable patent law.
</para>
<bridgehead xml:id="NoSurrender" renderas="sect1">
12. No Surrender of Others Freedom.
</bridgehead>
<para>
If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.
</para>
<bridgehead xml:id="UsedWithAGPL" renderas="sect1">
13. Use with the <acronym>GNU</acronym> Affero General Public License.
</bridgehead>
<para>
Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the <acronym>GNU</acronym> Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the <acronym>GNU</acronym> Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.
</para>
<bridgehead xml:id="RevisedVersions" renderas="sect1">
14. Revised Versions of this License.
</bridgehead>
<para>
The Free Software Foundation may publish revised and/or new versions of the <acronym>GNU</acronym> General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.
</para>
<para>
Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the <acronym>GNU</acronym> General Public License or any later version applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the <acronym>GNU</acronym> General Public License, you may choose any version ever published by the Free Software Foundation.
</para>

<para>
If the Program specifies that a proxy can decide which future versions of
the <acronym>GNU</acronym> General Public License can be used, that
proxys public statement of acceptance of a version permanently
authorizes you to choose that version for the Program.
</para>
<para>
Later license versions may give you additional or different permissions.
However, no additional obligations are imposed on any author or copyright
holder as a result of your choosing to follow a later version.
</para>
<bridgehead xml:id="WarrantyDisclaimer" renderas="sect1">
15. Disclaimer of Warranty.
</bridgehead>
<para>
THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE
LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE PROGRAM AS IS WITHOUT WARRANTY OF
ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.
THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH
YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL
NECESSARY SERVICING, REPAIR OR CORRECTION.
</para>
<bridgehead xml:id="LiabilityLimitation" renderas="sect1">
16. Limitation of Liability.
</bridgehead>
<para>
IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL
ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE
PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY
GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE
OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA
OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD
PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS),
EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF
SUCH DAMAGES.
</para>
<bridgehead xml:id="InterpretationSecs1516" renderas="sect1">
17. Interpretation of Sections 15 and 16.
</bridgehead>
<para>
If the disclaimer of warranty and limitation of liability provided above
cannot be given local legal effect according to their terms, reviewing
courts shall apply local law that most closely approximates an absolute
waiver of all civil liability in connection with the Program, unless a
warranty or assumption of liability accompanies a copy of the Program in
return for a fee.

</para>
<bridgehead>
END OF TERMS AND CONDITIONS
</bridgehead>
<bridgehead xml:id="HowToApply" renderas="sect1">
How to Apply These Terms to Your New Programs
</bridgehead>
<para>
If you develop a new program, and you want it to be of the greatest possible
use to the public, the best way to achieve this is to make it free software
which everyone can redistribute and change under these terms.
</para>
<para>
To do so, attach the following notices to the program. It is safest to
attach them to the start of each source file to most effectively state the
exclusion of warranty; and each file should have at least the
copyright line and a pointer to where the full notice is
found.
</para>
<screen>
<replaceable>one line to give the programs name and a brief idea of what it does.</replaceable>
Copyright (C) <replaceable>year</replaceable> <replaceable>name of author</replaceable>

This program is free software: you can redistribute it and/or modify
it under the terms of the <acronym>GNU</acronym> General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
<acronym>GNU</acronym> General Public License for more details.

You should have received a copy of the <acronym>GNU</acronym> General Public License
along with this program. If not, see <link xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="http://www.gnu.org/licenses/">http://www.gnu.org/licenses/</link>.
</screen>
<para>
Also add information on how to contact you by electronic and paper mail.
</para>
<para>
If the program does terminal interaction, make it output a short notice like
this when it starts in an interactive mode:
</para>
<screen>
<replaceable>program</replaceable> Copyright (C) <replaceable>year</replaceable> <replaceable>name of
author</replaceable>
This program comes with ABSOLUTELY NO WARRANTY; for details type <literal>show w</literal>.

This is free software, and you are welcome to redistribute it
under certain conditions; type <literal>show c</literal> for details.
</screen>
<para>
The hypothetical commands <literal>show w</literal> and
<literal>show c</literal> should show the appropriate parts of
the General Public License. Of course, your programs commands might be
different; for a GUI interface, you would use an about box.
</para>
<para>
You should also get your employer (if you work as a programmer) or school,
if any, to sign a copyright disclaimer for the program, if
necessary. For more information on this, and how to apply and follow the
<acronym>GNU</acronym> <acronym>GPL</acronym>, see
<link xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="http://www.gnu.org/licenses/">http://www.gnu.org/licenses/</link>.
</para>
<para>
The <acronym>GNU</acronym> General Public License does not permit
incorporating your program into proprietary programs. If your program is a
subroutine library, you may consider it more useful to permit linking
proprietary applications with the library. If this is what you want to do,
use the <acronym>GNU</acronym> Lesser General Public License instead of this
License. But first, please read <link xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="http://www.gnu.org/philosophy/why-not-lgpl.html">http://www.gnu.org/philosophy/why-not-
lgpl.html</link>.
</para>
</appendix>
<appendix xmlns="http://docbook.org/ns/docbook" version="5.0"
xml:id="appendix.gfdl-1.3">
<info><title>GNU Free Documentation License</title></info>
<?dbhtml filename="appendix_gfdl.html"?>

<simpara>Version 1.3, 3 November 2008</simpara>
<simpara>
Copyright 2000, 2001, 2002, 2007, 2008
<link xmlns:xlink="http://www.w3.org/1999/xlink" xlink:href="http://www.fsf.org/">Free Software Foundation,
Inc.</link>
</simpara>
<simpara>
Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.
</simpara>
<bridgehead xml:id="fdl-1-section0" renderas="sect2">
0. PREAMBLE
</bridgehead>
<simpara>
The purpose of this License is to make a manual, textbook, or other

functional and useful document free in the sense of freedom:
to assure everyone the effective freedom to copy and redistribute it, with
or without modifying it, either commercially or
noncommercially. Secondarily, this License preserves for the author and
publisher a way to get credit for their work, while not being considered
responsible for modifications made by others.
</simpara>
<simpara>
This License is a kind of copyleft, which means that
derivative works of the document must themselves be free in the same
sense. It complements the GNU General Public License, which is a copyleft
license designed for free software.
</simpara>
<simpara>
We have designed this License in order to use it for manuals for free
software, because free software needs free documentation: a free program
should come with manuals providing the same freedoms that the software
does. But this License is not limited to software manuals; it can be used
for any textual work, regardless of subject matter or whether it is
published as a printed book. We recommend this License principally for
works whose purpose is instruction or reference.
</simpara>
<bridgehead xml:id="fdl-1-section1" renderas="sect2">
1. APPLICABILITY AND DEFINITIONS
</bridgehead>
<simpara>
This License applies to any manual or other work, in any medium, that
contains a notice placed by the copyright holder saying it can be
distributed under the terms of this License. Such a notice grants a
world-wide, royalty-free license, unlimited in duration, to use that work
under the conditions stated herein. The Document, below,
refers to any such manual or work. Any member of the public is a licensee,
and is addressed as you. You accept the license if you copy,
modify or distribute the work in a way requiring permission under
copyright law.
</simpara>
<simpara>
A Modified Version of the Document means any work containing
the Document or a portion of it, either copied verbatim, or with
modifications and/or translated into another language.
</simpara>
<simpara>
A Secondary Section is a named appendix or a front-matter
section of the Document that deals exclusively with the relationship of
the publishers or authors of the Document to the Documents overall
subject (or to related matters) and contains nothing that could fall
directly within that overall subject. (Thus, if the Document is in part a
textbook of mathematics, a Secondary Section may not explain any

mathematics.) The relationship could be a matter of historical connection
with the subject or with related matters, or of legal, commercial,
philosophical, ethical or political position regarding them.
</simpara>
<simpara>
The Invariant Sections are certain Secondary Sections whose
titles are designated, as being those of Invariant Sections, in the notice
that says that the Document is released under this License. If a section
does not fit the above definition of Secondary then it is not allowed to
be designated as Invariant. The Document may contain zero Invariant
Sections. If the Document does not identify any Invariant Sections then
there are none.
</simpara>
<simpara>
The Cover Texts are certain short passages of text that are
listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says
that the Document is released under this License. A Front-Cover Text may
be at most 5 words, and a Back-Cover Text may be at most 25 words.
</simpara>
<simpara>
A Transparent copy of the Document means a machine-readable
copy, represented in a format whose specification is available to the
general public, that is suitable for revising the document
straightforwardly with generic text editors or (for images composed of
pixels) generic paint programs or (for drawings) some widely available
drawing editor, and that is suitable for input to text formatters or for
automatic translation to a variety of formats suitable for input to text
formatters. A copy made in an otherwise Transparent file format whose
markup, or absence of markup, has been arranged to thwart or discourage
subsequent modification by readers is not Transparent. An image format is
not Transparent if used for any substantial amount of text. A copy that is
not Transparent is called Opaque.
</simpara>
<simpara>
Examples of suitable formats for Transparent copies include plain ASCII
without markup, Texinfo input format, LaTeX input format, SGML or XML
using a publicly available DTD, and standard-conforming simple HTML,
PostScript or PDF designed for human modification. Examples of transparent
image formats include PNG, XCF and JPG. Opaque formats include proprietary
formats that can be read and edited only by proprietary word processors,
SGML or XML for which the DTD and/or processing tools are not generally
available, and the machine-generated HTML, PostScript or PDF produced by
some word processors for output purposes only.
</simpara>
<simpara>
The Title Page means, for a printed book, the title page
itself, plus such following pages as are needed to hold, legibly, the
material this License requires to appear in the title page. For works in

formats which do not have any title page as such, Title Page
means the text near the most prominent appearance of the works
title, preceding the beginning of the body of the text.
</simpara>
<simpara>
The publisher means any person or entity that distributes
copies of the Document to the public.
</simpara>
<simpara>
A section Entitled XYZ means a named subunit of the Document
whose title either is precisely XYZ or contains XYZ in parentheses
following text that translates XYZ in another language. (Here XYZ stands
for a specific section name mentioned below, such as
Acknowledgements, Dedications,
Endorsements, or History.) To Preserve
the Title of such a section when you modify the Document means that
it remains a section Entitled XYZ according to this
definition.
</simpara>
<simpara>
The Document may include Warranty Disclaimers next to the notice which
states that this License applies to the Document. These Warranty
Disclaimers are considered to be included by reference in this License,
but only as regards disclaiming warranties: any other implication that
these Warranty Disclaimers may have is void and has no effect on the
meaning of this License.
</simpara>
<bridgehead xml:id="fdl-1-section2" renderas="sect2">
2. VERBATIM COPYING
</bridgehead>
<simpara>
You may copy and distribute the Document in any medium, either
commercially or noncommercially, provided that this License, the copyright
notices, and the license notice saying this License applies to the
Document are reproduced in all copies, and that you add no other
conditions whatsoever to those of this License. You may not use technical
measures to obstruct or control the reading or further copying of the
copies you make or distribute. However, you may accept compensation in
exchange for copies. If you distribute a large enough number of copies you
must also follow the conditions in section 3.
</simpara>
<simpara>
You may also lend copies, under the same conditions stated above, and you
may publicly display copies.
</simpara>
<bridgehead xml:id="fdl-1-section3" renderas="sect2">
3. COPYING IN QUANTITY
</bridgehead>

<simpara>
If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Documents license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.
</simpara>
<simpara>
If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.
</simpara>
<simpara>
If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.
</simpara>
<simpara>
It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.
</simpara>
<bridgehead xml:id="fdl-1-section4" renderas="sect2">
4. MODIFICATIONS
</bridgehead>
<simpara>
You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:
</simpara>
<orderedlist numeration="upperalpha">

<listitem>
<simpara>
Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
</simpara>
</listitem>
<listitem>
<simpara>
List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
</simpara>
</listitem>
<listitem>
<simpara>
State on the Title page the name of the publisher of the Modified Version, as the publisher.
</simpara>
</listitem>
<listitem>
<simpara>
Preserve all the copyright notices of the Document.
</simpara>
</listitem>
<listitem>
<simpara>
Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
</simpara>
</listitem>
<listitem>
<simpara>
Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
</simpara>
</listitem>
<listitem>
<simpara>
Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Documents license notice.
</simpara>

</listitem>
<listitem>
<simpara>
Include an unaltered copy of this License.
</simpara>
</listitem>
<listitem>
<simpara>
Preserve the section Entitled History, Preserve its
Title, and add to it an item stating at least the title, year, new
authors, and publisher of the Modified Version as given on the Title
Page. If there is no section Entitled History in the
Document, create one stating the title, year, authors, and publisher
of the Document as given on its Title Page, then add an item
describing the Modified Version as stated in the previous sentence.
</simpara>
</listitem>
<listitem>
<simpara>
Preserve the network location, if any, given in the Document for
public access to a Transparent copy of the Document, and likewise the
network locations given in the Document for previous versions it was
based on. These may be placed in the History
section. You may omit a network location for a work that was published
at least four years before the Document itself, or if the original
publisher of the version it refers to gives permission.
</simpara>
</listitem>
<listitem>
<simpara>
For any section Entitled Acknowledgements or
Dedications, Preserve the Title of the section, and
preserve in the section all the substance and tone of each of the
contributor acknowledgements and/or dedications given therein.
</simpara>
</listitem>
<listitem>
<simpara>
Preserve all the Invariant Sections of the Document, unaltered in
their text and in their titles. Section numbers or the equivalent are
not considered part of the section titles.
</simpara>
</listitem>
<listitem>
<simpara>
Delete any section Entitled Endorsements. Such a section
may not be included in the Modified Version.
</simpara>

```
  </listitem>
  <listitem>
    <simpara>
      Do not retitle any existing section to be Entitled
      Endorsements or to conflict in title with any Invariant
      Section.
    </simpara>
  </listitem>
  <listitem>
    <simpara>
      Preserve any Warranty Disclaimers.
    </simpara>
  </listitem>
</orderedlist>
<simpara>
  If the Modified Version includes new front-matter sections or appendices
  that qualify as Secondary Sections and contain no material copied from the
  Document, you may at your option designate some or all of these sections
  as invariant. To do this, add their titles to the list of Invariant
  Sections in the Modified Versions license notice. These titles
  must be distinct from any other section titles.
</simpara>
<simpara>
  You may add a section Entitled Endorsements, provided it
  contains nothing but endorsements of your Modified Version by various
  parties  for example, statements of peer review or that the text
  has been approved by an organization as the authoritative definition of a
  standard.
</simpara>
<simpara>
  You may add a passage of up to five words as a Front-Cover Text, and a
  passage of up to 25 words as a Back-Cover Text, to the end of the list of
  Cover Texts in the Modified Version. Only one passage of Front-Cover Text
  and one of Back-Cover Text may be added by (or through arrangements made
  by) any one entity. If the Document already includes a cover text for the
  same cover, previously added by you or by arrangement made by the same
  entity you are acting on behalf of, you may not add another; but you may
  replace the old one, on explicit permission from the previous publisher
  that added the old one.
</simpara>
<simpara>
  The author(s) and publisher(s) of the Document do not by this License give
  permission to use their names for publicity for or to assert or imply
  endorsement of any Modified Version.
</simpara>
<bridgehead xml:id="fdl-1-section5" renderas="sect2">
  5. COMBINING DOCUMENTS
</bridgehead>
```

```
<simpara>
  You may combine the Document with other documents released under this
  License, under the terms defined in section 4 above for modified versions,
  provided that you include in the combination all of the Invariant Sections
  of all of the original documents, unmodified, and list them all as
  Invariant Sections of your combined work in its license notice, and that
  you preserve all their Warranty Disclaimers.
</simpara>
<simpara>
  The combined work need only contain one copy of this License, and multiple
  identical Invariant Sections may be replaced with a single copy. If there
  are multiple Invariant Sections with the same name but different contents,
  make the title of each such section unique by adding at the end of it, in
  parentheses, the name of the original author or publisher of that section
  if known, or else a unique number. Make the same adjustment to the section
  titles in the list of Invariant Sections in the license notice of the
  combined work.
</simpara>
<simpara>
  In the combination, you must combine any sections Entitled
  History in the various original documents, forming one
  section Entitled History; likewise combine any sections
  Entitled Acknowledgements, and any sections Entitled
  Dedications. You must delete all sections Entitled
  Endorsements.
</simpara>
<bridgehead xml:id="fdl-1-section6" renderas="sect2">
  6. COLLECTIONS OF DOCUMENTS
</bridgehead>
<simpara>
  You may make a collection consisting of the Document and other documents
  released under this License, and replace the individual copies of this
  License in the various documents with a single copy that is included in
  the collection, provided that you follow the rules of this License for
  verbatim copying of each of the documents in all other respects.
</simpara>
<simpara>
  You may extract a single document from such a collection, and distribute
  it individually under this License, provided you insert a copy of this
  License into the extracted document, and follow this License in all other
  respects regarding verbatim copying of that document.
</simpara>
<bridgehead xml:id="fdl-1-section7" renderas="sect2">
  7. AGGREGATION WITH INDEPENDENT WORKS
</bridgehead>
<simpara>
  A compilation of the Document or its derivatives with other separate and
  independent documents or works, in or on a volume of a storage or
```

distribution medium, is called an aggregate if the copyright
resulting from the compilation is not used to limit the legal rights of
the compilations users beyond what the individual works
permit. When the Document is included in an aggregate, this License does
not apply to the other works in the aggregate which are not themselves
derivative works of the Document.
</simpara>
<simpara>
If the Cover Text requirement of section 3 is applicable to these copies
of the Document, then if the Document is less than one half of the entire
aggregate, the Documents Cover Texts may be placed on covers that
bracket the Document within the aggregate, or the electronic equivalent of
covers if the Document is in electronic form. Otherwise they must appear
on printed covers that bracket the whole aggregate.
</simpara>
<bridgehead xml:id="fdl-1-section8" renderas="sect2">
8. TRANSLATION
</bridgehead>
<simpara>
Translation is considered a kind of modification, so you may distribute
translations of the Document under the terms of section 4. Replacing
Invariant Sections with translations requires special permission from
their copyright holders, but you may include translations of some or all
Invariant Sections in addition to the original versions of these Invariant
Sections. You may include a translation of this License, and all the
license notices in the Document, and any Warranty Disclaimers, provided
that you also include the original English version of this License and the
original versions of those notices and disclaimers. In case of a
disagreement between the translation and the original version of this
License or a notice or disclaimer, the original version will prevail.
</simpara>
<simpara>
If a section in the Document is Entitled Acknowledgements,
Dedications, or History, the requirement
(section 4) to Preserve its Title (section 1) will typically require
changing the actual title.
</simpara>
<bridgehead xml:id="fdl-1-section9" renderas="sect2">
9. TERMINATION
</bridgehead>
<simpara>
You may not copy, modify, sublicense, or distribute the Document except as
expressly provided under this License. Any attempt otherwise to copy,
modify, sublicense, or distribute it is void, and will automatically
terminate your rights under this License.
</simpara>
<simpara>
However, if you cease all violation of this License, then your license

from a particular copyright holder is reinstated (a) provisionally, unless
and until the copyright holder explicitly and finally terminates your
license, and (b) permanently, if the copyright holder fails to notify you
of the violation by some reasonable means prior to 60 days after the
cessation.
</simpara>
<simpara>
Moreover, your license from a particular copyright holder is reinstated
permanently if the copyright holder notifies you of the violation by some
reasonable means, this is the first time you have received notice of
violation of this License (for any work) from that copyright holder, and
you cure the violation prior to 30 days after your receipt of the notice.
</simpara>
<simpara>
Termination of your rights under this section does not terminate the
licenses of parties who have received copies or rights from you under this
License. If your rights have been terminated and not permanently
reinstated, receipt of a copy of some or all of the same material does not
give you any rights to use it.
</simpara>
<bridgehead xml:id="fdl-1-section10" renderas="sect2">
10. FUTURE REVISIONS OF THIS LICENSE
</bridgehead>
<simpara>
The Free Software Foundation may publish new, revised versions of the GNU
Free Documentation License from time to time. Such new versions will be
similar in spirit to the present version, but may differ in detail to
address new problems or concerns. See
<link xmlns:xlink="http://www.w3.org/1999/xlink" xlink:href="http://www.gnu.org/copyleft/">Copyleft</link>.
</simpara>
<simpara>
Each version of the License is given a distinguishing version number. If
the Document specifies that a particular numbered version of this License
or any later version applies to it, you have the option of
following the terms and conditions either of that specified version or of
any later version that has been published (not as a draft) by the Free
Software Foundation. If the Document does not specify a version number of
this License, you may choose any version ever published (not as a draft)
by the Free Software Foundation. If the Document specifies that a proxy
can decide which future versions of this License can be used, that
proxys public statement of acceptance of a version permanently
authorizes you to choose that version for the Document.
</simpara>
<bridgehead xml:id="fdl-1-section11" renderas="sect2">
11. RELICENSING
</bridgehead>
<simpara>
Massive Multiauthor Collaboration Site (or MMC

Site) means any World Wide Web server that publishes copyrightable
works and also provides prominent facilities for anybody to edit those
works. A public wiki that anybody can edit is an example of such a
server. A Massive Multiauthor Collaboration (or
MMC) contained in the site means any set of copyrightable
works thus published on the MMC site.
</simpara>
<simpara>
CC-BY-SA means the Creative Commons Attribution-Share Alike
3.0 license published by Creative Commons Corporation, a not-for-profit
corporation with a principal place of business in San Francisco,
California, as well as future copyleft versions of that license published
by that same organization.
</simpara>
<simpara>
Incorporate means to publish or republish a Document, in
whole or in part, as part of another Document.
</simpara>
<simpara>
An MMC is eligible for relicensing if it is licensed under
this License, and if all works that were first published under this
License somewhere other than this MMC, and subsequently incorporated in
whole or in part into the MMC, (1) had no cover texts or invariant
sections, and (2) were thus incorporated prior to November 1, 2008.
</simpara>
<simpara>
The operator of an MMC Site may republish an MMC contained in the site
under CC-BY-SA on the same site at any time before August 1, 2009,
provided the MMC is eligible for relicensing.
</simpara>
<bridgehead xml:id="fdl-1-addendum" renderas="sect2">
ADDENDUM: How to use this License for your documents
</bridgehead>
<simpara>
To use this License in a document you have written, include a copy of the
License in the document and put the following copyright and license
notices just after the title page:
</simpara>
<screen>Copyright YEAR YOUR NAME

Permission is granted to copy, distribute and/or modify this document under the
terms of the GNU Free Documentation License, Version 1.3 or any later version
published by the Free Software Foundation; with no Invariant Sections, no
Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in
the section entitled GNU Free Documentation License.</screen>
<simpara>
If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts,
replace the with Texts. line with this:

</simpara>
<screen>with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.</screen>
<simpara>
If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.
</simpara>
<simpara>
If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.
</simpara>
</appendix>

GCC RUNTIME LIBRARY EXCEPTION

Version 3.1, 31 March 2009

Copyright (C) 2009 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a notice placed by the copyright holder of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header files and runtime libraries with the compiled program. The purpose of this Exception is to allow compilation of non-GPL (including proprietary) programs to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for execution after a Compilation Process, or makes use of an interface provided by the Runtime Library, but is not otherwise based on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without modifications, governed by version 3 (or a specified later version) of the GNU General Public License (GPL) with the option of using any subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and use would permit combination with GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor architecture, in executable form or suitable for input to an assembler, loader, linker and/or execution phase. Notwithstanding that, Target Code does not include data in any format that is used as a compiler intermediate representation, or used for producing a compiler intermediate representation.

The "Compilation Process" transforms code entirely represented in non-intermediate languages designed for human-written code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use of source code generators and preprocessors need not be considered part of the Compilation Process, since the Compilation Process can be understood as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-compatible software, or if it is done without using any work based on GCC. For example, using non-GPL-compatible Software to optimize any GCC intermediate representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime Library with Independent Modules, even if such propagation would otherwise violate the terms of GPLv3, provided that all Target Code was generated by Eligible Compilation Processes. You may then convey such a combination under terms of your choice, consistent with the licensing of the Independent Modules.

2. No Weakening of GCC Copyleft.

The availability of this Exception does not imply any general presumption that third-party software is unaffected by the copyleft requirements of the license of GCC.
@node Library Copying
@appendixsec GNU LESSER GENERAL PUBLIC LICENSE

@cindex LGPL, Lesser General Public License
@center Version 2.1, February 1999

@display
Copyright @copyright{} 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street - Fifth Floor, Boston, MA 02110-1301, USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence the
version number 2.1.]
@end display

@appendixsubsec Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software---to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software---typically libraries---of the Free
Software Foundation and other authors who decide to use it. You can use
it too, but we suggest you first think carefully about whether this
license or the ordinary General Public License is the better strategy to
use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of it
in new free programs; and that you are informed that you can do these
things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the @dfn{Lesser} General Public License because it does @emph{Less} to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in

non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a ``work based on the library'' and a ``work that uses the library''. The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

@iftex
@appendixsubsec TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
@end iftex
@ifinfo
@center GNU LESSER GENERAL PUBLIC LICENSE
@center TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
@end ifinfo

@enumerate 0
@item
This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called ``this License''). Each licensee is addressed as ``you''.

A ``library'' means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The ``Library'', below, refers to any such software library or work which has been distributed under these terms. A ``work based on the Library'' means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term ``modification''.)

``Source code'' for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

@item
You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

@item
You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

@enumerate a
@item
The modified work must itself be a software library.

@item
You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

@item
You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

@item
If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has
a purpose that is entirely well-defined independent of the
application. Therefore, Subsection 2d requires that any
application-supplied function or table used by this function must
be optional: if the application does not supply it, the square
root function must still compute square roots.)
@end enumerate

These requirements apply to the modified work as a whole. If
identifiable sections of that work are not derived from the Library,
and can be reasonably considered independent and separate works in
themselves, then this License, and its terms, do not apply to those
sections when you distribute them as separate works. But when you
distribute the same sections as part of a whole which is a work based
on the Library, the distribution of the whole must be on the terms of
this License, whose permissions for other licensees extend to the
entire whole, and thus to each and every part regardless of who wrote
it.

Thus, it is not the intent of this section to claim rights or contest
your rights to work written entirely by you; rather, the intent is to
exercise the right to control the distribution of derivative or
collective works based on the Library.

In addition, mere aggregation of another work not based on the Library
with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

@item
You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.

Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

@item

You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

@item
A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a ``work that uses the Library''. Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a ``work that uses the Library'' with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a ``work that uses the library''. The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a ``work that uses the Library'' uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

@item
As an exception to the Sections above, you may also combine or link a ``work that uses the Library'' with the Library to produce a

work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

@enumerate a
@item
Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable ``work that uses the Library'', as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

@item
Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

@item
Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

@item
If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

@item

Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.
@end enumerate

For an executable, the required form of the ``work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies the
executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.

@item
You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

@enumerate a
@item
Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

@item
Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.
@end enumerate

@item
You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any
attempt otherwise to copy, modify, sublicense, link with, or
distribute the Library is void, and will automatically terminate your
rights under this License. However, parties who have received copies,
or rights, from you under this License will not have their licenses
terminated so long as such parties remain in full compliance.

@item
You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

@item
Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

@item
If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to
be a consequence of the rest of this License.

@item
If the distribution and/or use of the Library is restricted in
certain countries either by patents or by copyrighted interfaces, the
original copyright holder who places the Library under this License may add
an explicit geographical distribution limitation excluding those countries,
so that distribution is permitted only in or among countries not thus
excluded. In such case, this License incorporates the limitation as if
written in the body of this License.

@item
The Free Software Foundation may publish revised and/or new
versions of the Lesser General Public License from time to time.
Such new versions will be similar in spirit to the present version,
but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library
specifies a version number of this License which applies to it and
``any later version'', you have the option of following the terms and
conditions either of that version or of any later version published by
the Free Software Foundation. If the Library does not specify a
license version number, you may choose any version ever published by
the Free Software Foundation.

@item
If you wish to incorporate parts of the Library into other free
programs whose distribution conditions are incompatible with these,
write to the author to ask for permission. For software which is
copyrighted by the Free Software Foundation, write to the Free
Software Foundation; we sometimes make exceptions for this. Our
decision will be guided by the two goals of preserving the free status
of all derivatives of our free software and of promoting the sharing
and reuse of software generally.

@center NO WARRANTY

@item
BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO
WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW.
EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR
OTHER PARTIES PROVIDE THE LIBRARY ``AS IS'' WITHOUT WARRANTY OF ANY
KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME

THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

@item
IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN
WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY
AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU
FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR
CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE
LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING
RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A
FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF
SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH
DAMAGES.
@end enumerate

@iftex
@heading END OF TERMS AND CONDITIONS
@end iftex
@ifinfo
@center END OF TERMS AND CONDITIONS
@end ifinfo

@page
@appendixsubsec How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest
possible use to the public, we recommend making it free software that
everyone can redistribute and change. You can do so by permitting
redistribution under these terms (or, alternatively, under the terms of the
ordinary General Public License).

To apply these terms, attach the following notices to the library. It is
safest to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least the
``copyright'' line and a pointer to where the full notice is found.

@smallexample
@var{one line to give the library's name and an idea of what it does.}
Copyright (C) @var{year} @var{name of author}

This library is free software; you can redistribute it and/or modify it
under the terms of the GNU Lesser General Public License as published by
the Free Software Foundation; either version 2.1 of the License, or (at
your option) any later version.

This library is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301,
USA.
@end smallexample

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a ``copyright disclaimer'' for the library, if
necessary. Here is a sample; alter the names:

@smallexample
Yoyodyne, Inc., hereby disclaims all copyright interest in the library
`Frob' (a library for tweaking knobs) written by James Random Hacker.

@var{signature of Ty Coon}, 1 April 1990
Ty Coon, President of Vice
@end smallexample

That's all there is to it!
#!/usr/bin/python
#
Copyright (C) 2013 Free Software Foundation, Inc.
#
This script is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 3, or (at your option)
any later version.

This script adjusts the copyright notices at the top of source files
so that they have the form:
#
Copyright XXXX-YYYY Free Software Foundation, Inc.
#
It doesn't change code that is known to be maintained elsewhere or
that carries a non-FSF copyright.
#
The script also doesn't change testsuite files, except those in
libstdc++-v3. This is because libstdc++-v3 has a conformance testsuite,
while most tests in other directories are just things that failed at some
point in the past.
#
Pass --this-year to the script if you want it to add the current year
to all applicable notices. Pass --quilt if you are using quilt and
want files to be added to the quilt before being changed.

```
#
# By default the script will update all directories for which the
# output has been vetted.  You can instead pass the names of individual
# directories, including those that haven't been approved.  So:
#
#    update-copyright.pl --this-year
#
# is the command that would be used at the beginning of a year to update
# all copyright notices (and possibly at other times to check whether
# new files have been added with old years).  On the other hand:
#
#    update-copyright.pl --this-year libjava
#
# would run the script on just libjava/.
#
# Note that things like --version output strings must be updated before
# this script is run.  There's already a separate procedure for that.

import os
import re
import sys
import time
import subprocess

class Errors:
  def __init__ (self):
    self.num_errors = 0

  def report (self, filename, string):
    if filename:
      string = filename + ': ' + string
    sys.stderr.write (string + '\n')
    self.num_errors += 1

  def ok (self):
    return self.num_errors == 0

class GenericFilter:
  def __init__ (self):
    self.skip_files = set()
    self.skip_dirs = set()
    self.skip_extensions = set()
    self.fossilised_files = set()
    self.own_files = set()

    self.skip_files |= set ([
        # Skip licence files.
        'COPYING',
```

```
            'COPYING.LIB',
            'COPYING3',
            'COPYING3.LIB',
            'LICENSE',
            'fdl.texi',
            'gpl_v3.texi',
            'fdl-1.3.xml',
            'gpl-3.0.xml',

            # Skip auto- and libtool-related files
            'aclocal.m4',
            'compile',
            'config.guess',
            'config.sub',
            'depcomp',
            'install-sh',
            'libtool.m4',
            'ltmain.sh',
            'ltoptions.m4',
            'ltsugar.m4',
            'ltversion.m4',
            'lt~obsolete.m4',
            'missing',
            'mkdep',
            'mkinstalldirs',
            'move-if-change',
            'shlibpath.m4',
            'symlink-tree',
            'ylwrap',

            # Skip FSF mission statement, etc.
            'gnu.texi',
            'funding.texi',
            'appendix_free.xml',

            # Skip imported texinfo files.
            'texinfo.tex',
            ])


def get_line_filter (self, dir, filename):
    if filename.startswith ('ChangeLog'):
        # Ignore references to copyright in changelog entries.
        return re.compile ('\t')

    return None

def skip_file (self, dir, filename):
```

```
        if filename in self.skip_files:
            return True

        (base, extension) = os.path.splitext (os.path.join (dir, filename))
        if extension in self.skip_extensions:
            return True

        if extension == '.in':
            # Skip .in files produced by automake.
            if os.path.exists (base + '.am'):
                return True

            # Skip files produced by autogen
            if (os.path.exists (base + '.def')
                and os.path.exists (base + '.tpl')):
                return True

        # Skip configure files produced by autoconf
        if filename == 'configure':
            if os.path.exists (base + '.ac'):
                return True
            if os.path.exists (base + '.in'):
                return True

        return False

    def skip_dir (self, dir, subdir):
        return subdir in self.skip_dirs

    def is_fossilised_file (self, dir, filename):
        if filename in self.fossilised_files:
            return True
        # Only touch current current ChangeLogs.
        if filename != 'ChangeLog' and filename.find ('ChangeLog') >= 0:
            return True
        return False

    def by_package_author (self, dir, filename):
        return filename in self.own_files

class Copyright:
    def __init__ (self, errors):
        self.errors = errors

        # Characters in a range of years.  Include '.' for typos.
        ranges = '[0-9](?:[-0-9.,\s]|\s+and\s+)*[0-9]'

        # Non-whitespace characters in a copyright holder's name.
```

```
name = '[\w.,-]'

# Matches one year.
self.year_re = re.compile ('[0-9]+')

# Matches part of a year or copyright holder.
self.continuation_re = re.compile (ranges + '|' + name)

# Matches a full copyright notice:
self.copyright_re = re.compile (
    # 1: 'Copyright (C)', etc.
    '([Cc]opyright'
    '|[Cc]opyright\s+\([Cc]\)'
    '|[Cc]opyright\s+%s'
    '|[Cc]opyright\s+&copy;'
    '|[Cc]opyright\s+@copyright{}'
    '|@set\s+copyright[\w-]+)'

    # 2: the years.  Include the whitespace in the year, so that
    # we can remove any excess.
    '(\s*(?:' + ranges + ',?'
    '|@value\{[^{}]*\})\s*)'

    # 3: 'by ', if used
    '(by\s+)?'

    # 4: the copyright holder.  Don't allow multiple consecutive
    # spaces, so that right-margin gloss doesn't get caught
    # (e.g. gnat_ugn.texi).
    '(' + name + '(?:\s?' + name + ')*)?')

# A regexp for notices that might have slipped by.  Just matching
# 'copyright' is too noisy, and 'copyright.*[0-9]' falls foul of
# HTML header markers, so check for 'copyright' and two digits.
self.other_copyright_re = re.compile ('copyright.*[0-9][0-9]',
                                      re.IGNORECASE)
self.comment_re = re.compile('#+|[*]+|;+|%+|//+|@c |dnl ')
self.holders = { '@copying': '@copying' }
self.holder_prefixes = set()

# True to 'quilt add' files before changing them.
self.use_quilt = False

# If set, force all notices to include this year.
self.max_year = None

# Goes after the year(s).  Could be ', '.
self.separator = ' '
```

```
def add_package_author (self, holder, canon_form = None):
    if not canon_form:
        canon_form = holder
    self.holders[holder] = canon_form
    index = holder.find (' ')
    while index >= 0:
        self.holder_prefixes.add (holder[:index])
        index = holder.find (' ', index + 1)

def add_external_author (self, holder):
    self.holders[holder] = None

class BadYear():
    def __init__ (self, year):
        self.year = year

    def __str__ (self):
        return 'unrecognised year: ' + self.year

def parse_year (self, string):
    year = int (string)
    if len (string) == 2:
        if year > 70:
            return year + 1900
    elif len (string) == 4:
        return year
    raise self.BadYear (string)

def year_range (self, years):
    year_list = [self.parse_year (year)
                 for year in self.year_re.findall (years)]
    assert len (year_list) > 0
    return (min (year_list), max (year_list))

def set_use_quilt (self, use_quilt):
    self.use_quilt = use_quilt

def include_year (self, year):
    assert not self.max_year
    self.max_year = year

def canonicalise_years (self, dir, filename, filter, years):
    # Leave texinfo variables alone.
    if years.startswith ('@value'):
        return years

    (min_year, max_year) = self.year_range (years)
```

```
    # Update the upper bound, if enabled.
    if self.max_year and not filter.is_fossilised_file (dir, filename):
        max_year = max (max_year, self.max_year)

    # Use a range.
    if min_year == max_year:
        return '%d' % min_year
    else:
        return '%d-%d' % (min_year, max_year)

def strip_continuation (self, line):
    line = line.lstrip()
    match = self.comment_re.match (line)
    if match:
        line = line[match.end():].lstrip()
    return line

def is_complete (self, match):
    holder = match.group (4)
    return (holder
            and (holder not in self.holder_prefixes
                 or holder in self.holders))

def update_copyright (self, dir, filename, filter, file, line, match):
    orig_line = line
    next_line = None
    pathname = os.path.join (dir, filename)

    intro = match.group (1)
    if intro.startswith ('@set'):
        # Texinfo year variables should always be on one line
        after_years = line[match.end (2):].strip()
        if after_years != '':
            self.errors.report (pathname,
                                'trailing characters in @set: '
                                + after_years)
            return (False, orig_line, next_line)
    else:
        # If it looks like the copyright is incomplete, add the next line.
        while not self.is_complete (match):
            try:
                next_line = file.next()
            except StopIteration:
                break

            # If the next line doesn't look like a proper continuation,
            # assume that what we've got is complete.
```

```
            continuation = self.strip_continuation (next_line)
            if not self.continuation_re.match (continuation):
                break

            # Merge the lines for matching purposes.
            orig_line += next_line
            line = line.rstrip() + ' ' + continuation
            next_line = None

            # Rematch with the longer line, at the original position.
            match = self.copyright_re.match (line, match.start())
            assert match

        holder = match.group (4)

        # Use the filter to test cases where markup is getting in the way.
        if filter.by_package_author (dir, filename):
            assert holder not in self.holders

        elif not holder:
            self.errors.report (pathname, 'missing copyright holder')
            return (False, orig_line, next_line)

        elif holder not in self.holders:
            self.errors.report (pathname,
                                'unrecognised copyright holder: ' + holder)
            return (False, orig_line, next_line)

        else:
            # See whether the copyright is associated with the package
            # author.
            canon_form = self.holders[holder]
            if not canon_form:
                return (False, orig_line, next_line)

            # Make sure the author is given in a consistent way.
            line = (line[:match.start (4)]
                    + canon_form
                    + line[match.end (4):])

            # Remove any 'by'
            line = line[:match.start (3)] + line[match.end (3):]

    # Update the copyright years.
    years = match.group (2).strip()
    try:
        canon_form = self.canonicalise_years (dir, filename, filter, years)
    except self.BadYear as e:
```

```
            self.errors.report (pathname, str (e))
            return (False, orig_line, next_line)

        line = (line[:match.start (2)]
                + ' ' + canon_form + self.separator
                + line[match.end (2):])

        # Use the standard (C) form.
        if intro.endswith ('right'):
            intro += ' (C)'
        elif intro.endswith ('(c)'):
            intro = intro[:-3] + '(C)'
        line = line[:match.start (1)] + intro + line[match.end (1):]

        # Strip trailing whitespace
        line = line.rstrip() + '\n'

        return (line != orig_line, line, next_line)

    def process_file (self, dir, filename, filter):
        pathname = os.path.join (dir, filename)
        if filename.endswith ('.tmp'):
            # Looks like something we tried to create before.
            try:
                os.remove (pathname)
            except OSError:
                pass
            return

        lines = []
        changed = False
        line_filter = filter.get_line_filter (dir, filename)
        with open (pathname, 'r') as file:
            prev = None
            for line in file:
                while line:
                    next_line = None
                    # Leave filtered-out lines alone.
                    if not (line_filter and line_filter.match (line)):
                        match = self.copyright_re.search (line)
                        if match:
                            res = self.update_copyright (dir, filename, filter,
                                                         file, line, match)
                            (this_changed, line, next_line) = res
                            changed = changed or this_changed

                        # Check for copyright lines that might have slipped by.
                        elif self.other_copyright_re.search (line):
```

```
                self.errors.report (pathname,
                                    'unrecognised copyright: %s'
                                    % line.strip())
            lines.append (line)
            line = next_line

    # If something changed, write the new file out.
    if changed and self.errors.ok():
        tmp_pathname = pathname + '.tmp'
        with open (tmp_pathname, 'w') as file:
            for line in lines:
                file.write (line)
        if self.use_quilt:
            subprocess.call (['quilt', 'add', pathname])
        os.rename (tmp_pathname, pathname)

  def process_tree (self, tree, filter):
    for (dir, subdirs, filenames) in os.walk (tree):
        # Don't recurse through directories that should be skipped.
        for i in xrange (len (subdirs) - 1, -1, -1):
            if filter.skip_dir (dir, subdirs[i]):
                del subdirs[i]

        # Handle the files in this directory.
        for filename in filenames:
            if filter.skip_file (dir, filename):
                sys.stdout.write ('Skipping %s\n'
                                  % os.path.join (dir, filename))
            else:
                self.process_file (dir, filename, filter)

class CmdLine:
  def __init__ (self, copyright = Copyright):
    self.errors = Errors()
    self.copyright = copyright (self.errors)
    self.dirs = []
    self.default_dirs = []
    self.chosen_dirs = []
    self.option_handlers = dict()
    self.option_help = []

    self.add_option ('--help', 'Print this help', self.o_help)
    self.add_option ('--quilt', '"quilt add" files before changing them',
                     self.o_quilt)
    self.add_option ('--this-year', 'Add the current year to every notice',
                     self.o_this_year)

  def add_option (self, name, help, handler):
```

```
    self.option_help.append ((name, help))
    self.option_handlers[name] = handler

def add_dir (self, dir, filter = GenericFilter()):
    self.dirs.append ((dir, filter))

def o_help (self, option = None):
    sys.stdout.write ('Usage: %s [options] dir1 dir2...\n\n'
                      'Options:\n' % sys.argv[0])
    format = '%-15s %s\n'
    for (what, help) in self.option_help:
        sys.stdout.write (format % (what, help))
    sys.stdout.write ('\nDirectories:\n')

    format = '%-25s'
    i = 0
    for (dir, filter) in self.dirs:
        i += 1
        if i % 3 == 0 or i == len (self.dirs):
            sys.stdout.write (dir + '\n')
        else:
            sys.stdout.write (format % dir)
    sys.exit (0)

def o_quilt (self, option):
    self.copyright.set_use_quilt (True)

def o_this_year (self, option):
    self.copyright.include_year (time.localtime().tm_year)

def main (self):
    for arg in sys.argv[1:]:
        if arg[:1] != '-':
            self.chosen_dirs.append (arg)
        elif arg in self.option_handlers:
            self.option_handlers[arg] (arg)
        else:
            self.errors.report (None, 'unrecognised option: ' + arg)
    if self.errors.ok():
        if len (self.chosen_dirs) == 0:
            self.chosen_dirs = self.default_dirs
        if len (self.chosen_dirs) == 0:
            self.o_help()
        else:
            for chosen_dir in self.chosen_dirs:
                canon_dir = os.path.join (chosen_dir, '')
                count = 0
                for (dir, filter) in self.dirs:
```

```
                if (dir + os.sep).startswith (canon_dir):
                    count += 1
                    self.copyright.process_tree (dir, filter)
            if count == 0:
                self.errors.report (None, 'unrecognised directory: '
                                    + chosen_dir)
    sys.exit (0 if self.errors.ok() else 1)

#-----------------------------------------------------------------------------

class TopLevelFilter (GenericFilter):
  def skip_dir (self, dir, subdir):
    return True

class ConfigFilter (GenericFilter):
  def __init__ (self):
    GenericFilter.__init__ (self)

  def skip_file (self, dir, filename):
    if filename.endswith ('.m4'):
      pathname = os.path.join (dir, filename)
      with open (pathname) as file:
        # Skip files imported from gettext.
        if file.readline().find ('gettext-') >= 0:
          return True
    return GenericFilter.skip_file (self, dir, filename)

class GCCFilter (GenericFilter):
  def __init__ (self):
    GenericFilter.__init__ (self)

    self.skip_files |= set ([
        # Not part of GCC
        'math-68881.h',
        ])

    self.skip_dirs |= set ([
        # Better not create a merge nightmare for the GNAT folks.
        'ada',

        # Handled separately.
        'testsuite',
        ])

    self.skip_extensions |= set ([
        # Maintained by the translation project.
        '.po',
```

```
            # Automatically-generated.
            '.pot',
            ])

        self.fossilised_files |= set ([
            # Old news won't be updated.
            'ONEWS',
            ])

class TestsuiteFilter (GenericFilter):
    def __init__ (self):
        GenericFilter.__init__ (self)

        self.skip_extensions |= set ([
            # Don't change the tests, which could be woend by anyone.
            '.c',
            '.C',
            '.cc',
            '.h',
            '.hs',
            '.f',
            '.f90',
            '.go',
            '.inc',
            '.java',
            ])

    def skip_file (self, dir, filename):
        # g++.niklas/README contains historical copyright information
        # and isn't updated.
        if filename == 'README' and os.path.basename (dir) == 'g++.niklas':
            return True
        return GenericFilter.skip_file (self, dir, filename)

class LibCppFilter (GenericFilter):
    def __init__ (self):
        GenericFilter.__init__ (self)

        self.skip_extensions |= set ([
            # Maintained by the translation project.
            '.po',

            # Automatically-generated.
            '.pot',
            ])

class LibGCCFilter (GenericFilter):
    def __init__ (self):
```

```
    GenericFilter.__init__ (self)

    self.skip_dirs |= set ([
        # Imported from GLIBC.
        'soft-fp',
        ])

class LibJavaFilter (GenericFilter):
  def __init__ (self):
    GenericFilter.__init__ (self)

    self.skip_dirs |= set ([
        # Handled separately.
        'testsuite',

        # Not really part of the library
        'contrib',

        # Imported from upstream
        'classpath',
        'libltdl',
        ])

  def get_line_filter (self, dir, filename):
    if filename == 'NameDecoder.h':
      return re.compile ('.*NAME_COPYRIGHT')
    if filename == 'ICC_Profile.h':
      return re.compile ('.*icSigCopyrightTag')
    return GenericFilter.get_line_filter (self, dir, filename)

class LibMudflapFilter (GenericFilter):
  def __init__ (self):
    GenericFilter.__init__ (self)

    self.skip_dirs |= set ([
        # Handled separately.
        'testsuite',
        ])

class LibStdCxxFilter (GenericFilter):
  def __init__ (self):
    GenericFilter.__init__ (self)

    self.skip_files |= set ([
        # Contains no copyright of its own, but quotes the GPL.
        'intro.xml',
        ])
```

```
        self.skip_dirs |= set ([
            # Contains automatically-generated sources.
            'html',

            # The testsuite data files shouldn't be changed.
            'data',

            # Contains imported images
            'images',
            ])

        self.own_files |= set ([
            # Contains markup around the copyright owner.
            'spine.xml',
            ])

    def get_line_filter (self, dir, filename):
        if filename == 'boost_concept_check.h':
            return re.compile ('// \(C\) Copyright Jeremy Siek')
        return GenericFilter.get_line_filter (self, dir, filename)

class GCCCopyright (Copyright):
    def __init__ (self, errors):
        Copyright.__init__ (self, errors)

        canon_fsf = 'Free Software Foundation, Inc.'
        self.add_package_author ('Free Software Foundation', canon_fsf)
        self.add_package_author ('Free Software Foundation.', canon_fsf)
        self.add_package_author ('Free Software Foundation Inc.', canon_fsf)
        self.add_package_author ('Free Software Foundation, Inc', canon_fsf)
        self.add_package_author ('Free Software Foundation, Inc.', canon_fsf)
        self.add_package_author ('The Free Software Foundation', canon_fsf)
        self.add_package_author ('The Free Software Foundation, Inc.', canon_fsf)
        self.add_package_author ('Software Foundation, Inc.', canon_fsf)

        self.add_external_author ('ARM')
        self.add_external_author ('AdaCore')
        self.add_external_author ('Ami Tavory and Vladimir Dreizin, IBM-HRL.')
        self.add_external_author ('Cavium Networks.')
        self.add_external_author ('Faraday Technology Corp.')
        self.add_external_author ('Florida State University')
        self.add_external_author ('Greg Colvin and Beman Dawes.')
        self.add_external_author ('Hewlett-Packard Company')
        self.add_external_author ('Information Technology Industry Council.')
        self.add_external_author ('James Theiler, Brian Gough')
        self.add_external_author ('Makoto Matsumoto and Takuji Nishimura,')
        self.add_external_author ('National Research Council of Canada.')
        self.add_external_author ('Peter Dimov and Multi Media Ltd.')
```

```
        self.add_external_author ('Peter Dimov')
        self.add_external_author ('Pipeline Associates, Inc.')
        self.add_external_author ('Regents of the University of California.')
        self.add_external_author ('Silicon Graphics Computer Systems, Inc.')
        self.add_external_author ('Silicon Graphics')
        self.add_external_author ('Stephen L. Moshier')
        self.add_external_author ('Sun Microsystems, Inc. All rights reserved.')
        self.add_external_author ('The Go Authors.  All rights reserved.')
        self.add_external_author ('The Go Authors. All rights reserved.')
        self.add_external_author ('The Go Authors.')
        self.add_external_author ('The Regents of the University of California.')
        self.add_external_author ('Unicode, Inc.')
        self.add_external_author ('University of Toronto.')

class GCCCmdLine (CmdLine):
    def __init__ (self):
        CmdLine.__init__ (self, GCCCopyright)

        self.add_dir ('.', TopLevelFilter())
        # boehm-gc is imported from upstream.
        self.add_dir ('config', ConfigFilter())
        # contrib isn't really part of GCC.
        self.add_dir ('fixincludes')
        self.add_dir ('gcc', GCCFilter())
        self.add_dir (os.path.join ('gcc', 'testsuite'), TestsuiteFilter())
        self.add_dir ('gnattools')
        self.add_dir ('include')
        self.add_dir ('libada')
        self.add_dir ('libatomic')
        self.add_dir ('libbacktrace')
        self.add_dir ('libcpp', LibCppFilter())
        self.add_dir ('libdecnumber')
        # libffi is imported from upstream.
        self.add_dir ('libgcc', LibGCCFilter())
        self.add_dir ('libgfortran')
        self.add_dir ('libgomp')
        self.add_dir ('libiberty')
        self.add_dir ('libitm')
        self.add_dir ('libjava', LibJavaFilter())
        self.add_dir (os.path.join ('libjava', 'testsuite'), TestsuiteFilter())
        self.add_dir ('libmudflap', LibMudflapFilter())
        self.add_dir (os.path.join ('libmudflap', 'testsuite'),
                      TestsuiteFilter())
        self.add_dir ('libobjc')
        self.add_dir ('libquadmath')
        # libsanitiser is imported from upstream.
        self.add_dir ('libssp')
        self.add_dir ('libstdc++-v3', LibStdCxxFilter())
```

```
    self.add_dir ('lto-plugin')
    # zlib is imported from upstream.

    self.default_dirs = [
      'gcc',
      'libada',
      'libatomic',
      'libbacktrace',
      'libcpp',
      'libdecnumber',
      'libgcc',
      'libgfortran',
      'libgomp',
      'libitm',
      'libmudflap',
      'libobjc',
      'libstdc++-v3',
      ]

GCCCmdLine().main()
```

libffi - Copyright (c) 1996-2012 Anthony Green, Red Hat, Inc and others.
See source files for details.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
``Software''), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively

when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you

received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or

otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software

Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) 19yy <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) 19yy name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Library General
Public License instead of this License.
Copyright (c) 2001, Thai Open Source Software Center Ltd, Sun Microsystems.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

Neither the names of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
/* gnu.classpath.tools.taglets.CopyrightTaglet
Copyright (C) 2001 Free Software Foundation, Inc.

This file is part of GNU Classpath.

GNU Classpath is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2, or (at your option)
any later version.

GNU Classpath is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with GNU Classpath; see the file COPYING. If not, write to the
Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA
02111-1307 USA.

Linking this library statically or dynamically with other modules is

making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version. */

```
package gnu.classpath.tools.taglets;

import java.util.Map;

import com.sun.tools.doclets.Taglet;

import com.sun.javadoc.Tag;

/**
*  A simple Taglet which handles Copyright information.
*/
public class CopyrightTaglet implements Taglet {

  private static final String NAME = "copyright";
  private static final String HEADER = "Copyright:";

  public String getName() {
    return NAME;
  }

  public boolean inField() {
    return true;
  }

  public boolean inConstructor() {
    return true;
  }

  public boolean inMethod() {
    return true;
  }
```

```
public boolean inOverview() {
  return true;
}

public boolean inPackage() {
  return true;
}

public boolean inType() {
  return true;
}

public boolean isInlineTag() {
  return false;
}

public static void register(Map tagletMap) {
  CopyrightTaglet copyrightTaglet = new CopyrightTaglet();
  tagletMap.put(copyrightTaglet.getName(), copyrightTaglet);
}

public String toString(Tag tag) {
  return toString(new Tag[] { tag });
}

public String toString(Tag[] tags) {
  if (tags.length == 0) {
    return null;
  }
  else {
    boolean haveValidTag = false;
    for (int i = 0; i < tags.length && !haveValidTag; ++i) {
      if (tags[i].text().length() > 0) {
        haveValidTag = true;
      }
    }

    if (haveValidTag) {
      StringBuffer result = new StringBuffer();
      result.append("<dl>");
      for (int i = 0; i < tags.length; i++) {
        if (tags[i].text().length() > 0) {
          result.append("<dt><i>Copyright &#169; " + tags[i].text() + "</i></dt>");
        }
      }
      result.append("</dl>");
      return result.toString();
    }
```

```
      else {
        return null;
      }
    }
  }
}
/* Permission.java -- The superclass for all permission objects
  Copyright (C) 1998, 2001, 2002, 2005  Free Software Foundation, Inc.

This file is part of GNU Classpath.

GNU Classpath is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2, or (at your option)
any later version.

GNU Classpath is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with GNU Classpath; see the file COPYING.  If not, write to the
Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA
02110-1301 USA.

Linking this library statically or dynamically with other modules is
making a combined work based on this library.  Thus, the terms and
conditions of the GNU General Public License cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module.  An independent module is a module which is not derived from
or based on this library.  If you modify this library, you may extend
this exception to your version of the library, but you are not
obligated to do so.  If you do not wish to do so, delete this
exception statement from your version. */

package java.security;

import gnu.java.lang.CPStringBuilder;

import java.io.Serializable;
```

```
/**
* This class is the abstract superclass of all classes that implement
* the concept of a permission.  A permission consists of a permission name
* and optionally a list of actions that relate to the permission.  The
* actual meaning of the name of the permission is defined only in the
* context of a subclass.  It may name a resource to which access permissions
* are granted (for example, the name of a file) or it might represent
* something else entirely.  Similarly, the action list only has meaning
* within the context of a subclass.  Some permission names may have no
* actions associated with them.  That is, you either have the permission
* or you don't.
*
* <p>The most important method in this class is <code>implies</code>.  This
* checks whether if one has this permission, then the specified
* permission is also implied.  As a conceptual example, consider the
* permissions "Read All Files" and "Read File foo".  The permission
* "Read All Files" implies that the caller has permission to read the
* file foo.
*
* <p><code>Permission</code>'s must be immutable - do not change their
* state after creation.
*
* @author Aaron M. Renn (arenn@urbanophile.com)
* @see Permissions
* @see PermissionCollection
* @since 1.1
* @status updated to 1.4
*/
public abstract class Permission implements Guard, Serializable
{
 /**
  * Compatible with JDK 1.1+.
  */
 private static final long serialVersionUID = -5636570222231596674L;

 /**
  * This is the name assigned to this permission object.
  *
  * @serial the name of the permission
  */
 private String name;

 /**
  * Create an instance with the specified name.
  *
  * @param name the permission name
  */
```

```
public Permission(String name)
{
  this.name = name;
}

/**
 * This method implements the <code>Guard</code> interface for this class.
 * It calls the <code>checkPermission</code> method in
 * <code>SecurityManager</code> with this <code>Permission</code> as its
 * argument.  This method returns silently if the security check succeeds
 * or throws an exception if it fails.
 *
 * @param obj the <code>Object</code> being guarded - ignored by this class
 * @throws SecurityException if the security check fails
 * @see GuardedObject
 * @see SecurityManager#checkPermission(Permission)
 */
public void checkGuard(Object obj)
{
  SecurityManager sm = System.getSecurityManager();
  if (sm != null)
    sm.checkPermission(this);
}

/**
 * This method tests whether this <code>Permission</code> implies that the
 * specified <code>Permission</code> is also granted.
 *
 * @param perm the <code>Permission</code> to test against
 * @return true if perm is implied by this
 */
public abstract boolean implies(Permission perm);

/**
 * Check to see if this object equals obj. Use <code>implies</code>, rather
 * than <code>equals</code>, when making access control decisions.
 *
 * @param obj the object to compare to
 */
public abstract boolean equals(Object obj);

/**
 * This method returns a hash code for this <code>Permission</code>. It
 * must satisfy the contract of <code>Object.hashCode</code>: it must be
 * the same for all objects that equals considers to be the same.
 *
 * @return a hash value
 */
```

```
public abstract int hashCode();

/**
 * Get the name of this <code>Permission</code>.
 *
 * @return the name
 */
public final String getName()
{
  return name;
}

/**
 * This method returns the list of actions for this <code>Permission</code>
 * as a <code>String</code>. The string should be in canonical order, for
 * example, both <code>new FilePermission(f, "write,read")</code> and
 * <code>new FilePermission(f, "read,write")</code> have the action list
 * "read,write".
 *
 * @return the action list for this <code>Permission</code>
 */
public abstract String getActions();

/**
 * This method returns an empty <code>PermissionCollection</code> object
 * that can store permissions of this type, or <code>null</code> if no
 * such collection is defined. Subclasses must override this to provide
 * an appropriate collection when one is needed to accurately calculate
 * <code>implies</code>.
 *
 * @return a new <code>PermissionCollection</code>
 */
public PermissionCollection newPermissionCollection()
{
  return null;
}

/**
 * This method returns a <code>String</code> representation of this
 * <code>Permission</code> object. This is in the format:
 * <code>'(' + getClass().getName() + ' ' + getName() + ' ' + getActions
 * + ')'</code>.
 *
 * @return this object as a <code>String</code>
 */
public String toString()
{
  CPStringBuilder string = new CPStringBuilder();
```

```
  string = string.append('(');
  string = string.append(getClass().getName());
  string = string.append(' ');
  string = string.append(getName());

  if (!(getActions().equals("")))
    {
      string = string.append(' ');
      string = string.append(getActions());
    }

  string = string.append(')');
  return string.toString();
 }
} // class Permission
/* Permission.java -- Information about an ACL permission
  Copyright (C) 1998 Free Software Foundation, Inc.

This file is part of GNU Classpath.

GNU Classpath is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2, or (at your option)
any later version.

GNU Classpath is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with GNU Classpath; see the file COPYING.  If not, write to the
Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA
02110-1301 USA.

Linking this library statically or dynamically with other modules is
making a combined work based on this library.  Thus, the terms and
conditions of the GNU General Public License cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module.  An independent module is a module which is not derived from
```

```
or based on this library.  If you modify this library, you may extend
this exception to your version of the library, but you are not
obligated to do so.  If you do not wish to do so, delete this
exception statement from your version. */

package java.security.acl;

/**
* This interface provides information about a permission that can be
* granted.  Note that this is <em>not</em> the same as the class
* <code>java.security.Permission</code>.
*
* @version 0.0
*
* @author Aaron M. Renn (arenn@urbanophile.com)
*/
public interface Permission
{
 /**
  * This method tests whether or not a specified <code>Permission</code>
  * (passed as an <code>Object</code>) is the same as this permission.
  *
  * @param perm The permission to check for equality
  *
  * @return <code>true</code> if the specified permission is the same as this one, <code>false</code> otherwise
  */
 boolean equals (Object perm);

 /**
  * This method returns this <code>Permission</code> as a <code>String</code>.
  *
  * @return A <code>String</code> representing this permission.
  */
 String toString();
}
The software in this package is distributed under the GNU General Public
License (with a special exception described below).

A copy of GNU General Public License (GPL) is included in this distribution,
in the file COPYING.  If you do not have the source code, it is available at:

   http://www.gnu.org/software/classpath/

 Linking this library statically or dynamically with other modules is
 making a combined work based on this library.  Thus, the terms and
 conditions of the GNU General Public License cover the whole
 combination.
```

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

GNU Classpath also contains other (example) code distributed under other terms. External libraries included with GNU Classpath may also be distributed under different licensing terms.
The location and the exact terms of this other code is mentioned below.

Directory examples.
All example code is distributed under the GNU General Public License (GPL).
Each example source code files carries the following notice:

GNU Classpath examples are free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2, or (at your option) any later version.

GNU Classpath examples are distributed in the hope that they will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with GNU Classpath examples; see the file COPYING. If not, write to the Free Software Foundation, 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

For more information see the README file in the examples directory.

Directory native/fdlibm
fdlimb contains general algorithms useful for runtimes and compilers to support strict double and float mathematical operations.

fdlibm files carry the following notices:

Copyright (c) 1991 by AT&T.

Permission to use, copy, modify, and distribute this software for any
purpose without fee is hereby granted, provided that this entire notice
is included in all copies of any software which is or includes a copy
or modification of this software and in all copies of the supporting
documentation for such software.

THIS SOFTWARE IS BEING PROVIDED "AS IS", WITHOUT ANY EXPRESS OR IMPLIED
WARRANTY. IN PARTICULAR, NEITHER THE AUTHOR NOR AT&T MAKES ANY
REPRESENTATION OR WARRANTY OF ANY KIND CONCERNING THE MERCHANTABILITY
OF THIS SOFTWARE OR ITS FITNESS FOR ANY PARTICULAR PURPOSE.

Copyright (C) 1993 by Sun Microsystems, Inc. All rights reserved.

Developed at SunPro, a Sun Microsystems, Inc. business.
Permission to use, copy, modify, and distribute this
software is freely granted, provided that this notice
is preserved.

The upstream for fdlibm is libgcj (http://gcc.gnu.org/java/), they sync
again with the 'real' upstream (http://www.netlib.org/fdlibm/readme).

The documentation of some of the files in org/ietf/jgss/ is derived from
the text of RFC 2853: Generic Security Service API Version 2: Java Bindings.
That document is covered under the following license notice:

Copyright (C) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to
others, and derivative works that comment on or otherwise explain it
or assist in its implementation may be prepared, copied, published and
distributed, in whole or in part, without restriction of any kind,
provided that the above copyright notice and this paragraph are
included on all such copies and derivative works. However, this
document itself may not be modified in any way, such as by removing
the copyright notice or references to the Internet Society or other
Internet organizations, except as needed for the purpose of developing
Internet standards in which case the procedures for copyrights defined
in the Internet Standards process must be followed, or as required to
translate it into languages other than English.

The limited permissions granted above are perpetual and will not be
revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an
"AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING
TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT

NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Directory external/w3c_dom
Bindings for the Document Object Model (DOM) as published by the World Wide Web Consortium. All files are distributed under the following W3C Software Short Notice:

Copyright (c) 2004 World Wide Web Consortium,

(Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved. This work is distributed under the W3C(r) Software License [1] in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

[1] http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in

advertising or publicity pertaining to the software without specific,
written prior permission. Title to copyright in this software and any
associated documentation will at all times remain with copyright
holders.

Directory external/sax
Simple API for XML, a standard application interface for processing XML.
All files are distributed with the following short notice:

NO WARRANTY! This class is in the Public Domain.

The www.saxproject.org explains:

Copyright Status

SAX is free!

In fact, it's not possible to own a license to SAX, since it's been
placed in the public domain.

No Warranty

Because SAX is released to the public domain, there is no warranty
for the design or for the software implementation, to the extent
permitted by applicable law. Except when otherwise stated in writing
the copyright holders and/or other parties provide SAX "as is" without
warranty of any kind, either expressed or implied, including, but not
limited to, the implied warranties of merchantability and fitness for
a particular purpose. The entire risk as to the quality and
performance of SAX is with you. Should SAX prove defective, you assume
the cost of all necessary servicing, repair or correction.

In no event unless required by applicable law or agreed to in
writing will any copyright holder, or any other party who may modify
and/or redistribute SAX, be liable to you for damages, including any
general, special, incidental or consequential damages arising out of
the use or inability to use SAX (including but not limited to loss of
data or data being rendered inaccurate or losses sustained by you or
third parties or a failure of the SAX to operate with any other
programs), even if such holder or other party has been advised of the
possibility of such damages.

Copyright Disclaimers

This page includes statements to that effect by David Megginson, who
would have been able to claim copyright for the original work.

SAX 1.0

Version 1.0 of the Simple API for XML (SAX), created collectively by the membership of the XML-DEV mailing list, is hereby released into the public domain.

No one owns SAX: you may use it freely in both commercial and non-commercial applications, bundle it with your software distribution, include it on a CD-ROM, list the source code in a book, mirror the documentation at your own web site, or use it in any other way you see fit.

David Megginson, sax@megginson.com
1998-05-11

SAX 2.0

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

Two files in gnu/xml/aelfred2 (SAXDriver.java and XmlParser.java) were originally derived from code which carried the following notice:

Copyright (c) 1997, 1998 by Microstar Software Ltd.

AElfred is free for both commercial and non-commercial use and redistribution, provided that Microstar's copyright and disclaimer are retained intact. You are free to modify AElfred for your own use and to redistribute AElfred with your modifications, provided that the modifications are clearly documented.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of merchantability or fitness for a particular purpose. Please use it AT YOUR OWN RISK.

The files in resource/gnu/java/locale were generated by gnu.localegen from the files distributed from date provided by CLDR.
All these files are distributed under the following terms:

Copyright (C) 1991-2005 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

The file native/jni/midi-dssi/gnu_javax_sound_midi_dssi_DSSISynthesizer.c contains two functions (get_port_default and set_control) derived from example code in the DSSI distribution (http://dssi.sourceforge.net). The original DSSI example code is distributed under the following terms:

Copyright 2004 Chris Cannam, Steve Harris and Sean Bolton.

Permission to use, copy, modify, distribute, and sell this software for any purpose is hereby granted without fee, provided that the above copyright notice and this permission notice are included in all copies or substantial portions of the software.

The files in java/util/concurrent and subdirectories (with the
exception of CopyOnWriteArrayList.java), along with
java.util.NavigableMap, java.util.NavigableSet, java.util.Deque and
java.util.ArrayDeque, are taken from JSR166 concurrency materials
maintained by Doug Lea and distributed under the following terms:

Written by Doug Lea with assistance from members of JCP JSR-166
Expert Group and released to the public domain, as explained at
http://creativecommons.org/licenses/publicdomain

CopyOnWriteArrayList.java has been reimplemented for GNU Classpath,
and is distributed under the same terms as other GNU Classpath files,
as specified at the top of this file.

Directory external/relaxngDatatype
RELAX NG Pluggable Datatype Libraries. All files are distributed under
the following notice:

Copyright (c) 2001, Thai Open Source Software Center Ltd, Sun
Microsystems. All rights reserved.

Redistribution and use in source and binary forms, with or
without
modification, are permitted provided that the following
conditions are met:

Redistributions of source code must retain the above
copyright
notice, this list of conditions and the following
disclaimer.

Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials
provided
with the distribution.

Neither the names of the copyright holders nor the names of
its
contributors may be used to endorse or promote products
derived
from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT
NOT

LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

* m4/ax_func_which_gethostbyname_r.m4

Copyright 2005 Caolan McNamara <caolan@skynet.ie>
Copyright 2005 Daniel Richard G. <skunk@iskunk.org>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307,
USA.

As a special exception, the respective Autoconf Macro's copyright
owner gives unlimited permission to copy, distribute and modify the
configure scripts that are the output of Autoconf when processing the
Macro. You need not follow the terms of the GNU General Public License
when using or distributing such scripts, even though portions of the
text of the Macro appear in them. The GNU General Public License
(GPL) does govern all other use of the material that constitutes the
Autoconf Macro.

This special exception to the GPL applies to versions of the
Autoconf Macro released by the Autoconf Macro Archive. When you make
and distribute a modified version of the Autoconf Macro, you may
extend this special exception to the GPL to apply to your modified
version as well.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete

machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further

restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest

to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into

proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
Jan 21, 2002

The libgcj library is licensed under the terms of the GNU General Public License.

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

You should have received a copy of the GNU General Public License along with libjava; see the file COPYING. If not, write to the Free Software Foundation, 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change

free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.
^L
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The

former contains code derived from the library, whereas the latter must
be combined with the library in order to run.
^L

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other
program which contains a notice placed by the copyright holder or
other authorized party saying it may be distributed under the terms of
this Lesser General Public License (also called "this License").
Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data
prepared so as to be conveniently linked with application programs
(which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work
which has been distributed under these terms. A "work based on the
Library" means either the Library or any derivative work under
copyright law: that is to say, a work containing the Library or a
portion of it, either verbatim or with modifications and/or translated
straightforwardly into another language. (Hereinafter, translation is
included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for
making modifications to it. For a library, complete source code means
all the source code for all modules it contains, plus any associated
interface definition files, plus the scripts used to control
compilation
and installation of the library.

Activities other than copying, distribution and modification are not
covered by this License; they are outside its scope. The act of
running a program using the Library is not restricted, and output from
such a program is covered only if its contents constitute a work based
on the Library (independent of the use of the Library in a tool for
writing it). Whether that is true depends on what the Library does
and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's
complete source code as you receive it, in any medium, provided that
you conspicuously and appropriately publish on each copy an
appropriate copyright notice and disclaimer of warranty; keep intact
all the notices that refer to this License and to the absence of any
warranty; and distribute a copy of this License along with the
Library.

You may charge a fee for the physical act of transferring a copy,

and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library

with the Library (or with a work based on the Library) on a volume of
a storage or distribution medium does not bring the other work under
the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public
License instead of this License to a given copy of the Library. To do
this, you must alter all the notices that refer to this License, so
that they refer to the ordinary GNU General Public License, version 2,
instead of to this License. (If a newer version than version 2 of the
ordinary GNU General Public License has appeared, then you can specify
that version instead if you wish.) Do not make any other change in
these notices.
^L
Once this change is made in a given copy, it is irreversible for
that copy, so the ordinary GNU General Public License applies to all
subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of
the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or
derivative of it, under Section 2) in object code or executable form
under the terms of Sections 1 and 2 above provided that you accompany
it with the complete corresponding machine-readable source code, which
must be distributed under the terms of Sections 1 and 2 above on a
medium customarily used for software interchange.

If distribution of object code is made by offering access to copy
from a designated place, then offering equivalent access to copy the
source code from the same place satisfies the requirement to
distribute the source code, even though third parties are not
compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the
Library, but is designed to work with the Library by being compiled or
linked with it, is called a "work that uses the Library". Such a
work, in isolation, is not a derivative work of the Library, and
therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library
creates an executable that is a derivative of the Library (because it
contains portions of the Library), rather than a "work that uses the
library". The executable is therefore covered by this License.
Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file
that is part of the Library, the object code for the work may be a
derivative work of the Library even though the source code is not.

Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.
^L
6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if

the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.
^L
7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute
the Library except as expressly provided under this License. Any

attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.
^L
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that

system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.
^L
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

^L

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper

mail.

You should also get your employer (if you work as a programmer) or
your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James
Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and
all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts

as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be

introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the

users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an

appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the

library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any

patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR

OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or
your school, if any, to sign a "copyright disclaimer" for the library,
if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James
Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get

it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with

the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the

Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the

Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time.

Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

1.12 git 1.8.3.1

1.12.1 Available under license :

Copyright (C) 2010 David Barr <david.barr@cordelta.com>.
All rights reserved.

Copyright (C) 2010 Jonathan Nieder <jrnieder@gmail.com>.

Copyright (C) 2005 Stefan Hegny, hydrografix Consulting GmbH,
Frankfurt/Main, Germany
and others, see http://svn2cc.sarovar.org

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
  notice(s), this list of conditions and the following disclaimer
  unmodified other than the allowable addition of one or more
  copyright notices.
2. Redistributions in binary form must reproduce the above copyright
  notice(s), this list of conditions and the following disclaimer in
  the documentation and/or other materials provided with the
  distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDER(S) ``AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER(S) BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

While most of this project is under the GPL (see COPYING), the xdiff/
library and some libc code from compat/ are licensed under the
GNU LGPL, version 2.1 or (at your option) any later version and some
other files are under other licenses. Check the individual files to
be sure.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid
distributors to deny you these rights or to ask you to surrender these
rights. These restrictions translate to certain responsibilities for
you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis
or for a fee, you must give the recipients all the rights that we gave
you. You must make sure that they, too, receive or can get the source
code. If you link other code with the library, you must provide
complete object files to the recipients, so that they can relink them
with the library after making changes to the library and recompiling
it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the
library, and (2) we offer you this license, which gives you legal
permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in

non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does

and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based

on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and

therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the

user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any

particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Note that the only valid version of the GPL as far as this project
is concerned is _this_ particular version of the license (ie v2, not
v2.2 or v3.x or whatever), unless explicitly otherwise stated.

HOWEVER, in order to allow a migration to GPLv3 if that seems like
a good idea, I also ask that people involved with the project make
their preferences known. In particular, if you trust me to make that
decision, you might note so in your copyright message, ie something
like

This file is licensed under the GPL v2, or a later version
at the discretion of Linus.

might avoid issues. But we can also just decide to synchronize and
contact all copyright holders on record if/when the occasion arises.

Linus Torvalds

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and

modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such

interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such

an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not

excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify

it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.
Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,

execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another

language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other

circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY

FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.13 graphviz 2.30.1 :21.el7

1.13.1 Available under license :

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">

<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
<title>Eclipse Public License - Version 1.0</title>
<style type="text/css">
 body {
   size: 8.5in 11.0in;
   margin: 0.25in 0.5in 0.25in 0.5in;
```

```
  tab-interval: 0.5in;
  }
 p {
  margin-left: auto;
  margin-top:  0.5em;
  margin-bottom: 0.5em;
  }
 p.list {
  margin-left: 0.5in;
  margin-top:  0.05em;
  margin-bottom: 0.05em;
  }
 </style>

</head>

<body lang="EN-US">

<h2>Eclipse Public License - v 1.0</h2>

<p>THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR
DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS
AGREEMENT.</p>

<p><b>1. DEFINITIONS</b></p>

<p>"Contribution" means:</p>

<p class="list">a) in the case of the initial Contributor, the initial
code and documentation distributed under this Agreement, and</p>
<p class="list">b) in the case of each subsequent Contributor:</p>
<p class="list">i) changes to the Program, and</p>
<p class="list">ii) additions to the Program;</p>
<p class="list">where such changes and/or additions to the Program
originate from and are distributed by that particular Contributor. A
Contribution 'originates' from a Contributor if it was added to the
Program by such Contributor itself or anyone acting on such
Contributor's behalf. Contributions do not include additions to the
Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii)
are not derivative works of the Program.</p>

<p>"Contributor" means any person or entity that distributes
the Program.</p>

<p>"Licensed Patents" mean patent claims licensable by a
Contributor which are necessarily infringed by the use or sale of its
```

Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance
with this Agreement.

"Recipient" means anyone who receives the Program under
this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each
Contributor hereby grants Recipient a non-exclusive, worldwide,
royalty-free copyright license to reproduce, prepare derivative works
of, publicly display, publicly perform, distribute and sublicense the
Contribution of such Contributor, if any, and such derivative works, in
source code and object code form.

b) Subject to the terms of this Agreement, each
Contributor hereby grants Recipient a non-exclusive, worldwide,
royalty-free patent license under Licensed Patents to make, use, sell,
offer to sell, import and otherwise transfer the Contribution of such
Contributor, if any, in source code and object code form. This patent
license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor,
such addition of the Contribution causes such combination to be covered
by the Licensed Patents. The patent license shall not apply to any other
combinations which include the Contribution. No hardware per se is
licensed hereunder.

c) Recipient understands that although each Contributor
grants the licenses to its Contributions set forth herein, no assurances
are provided by any Contributor that the Program does not infringe the
patent or other intellectual property rights of any other entity. Each
Contributor disclaims any liability to Recipient for claims brought by
any other entity based on infringement of intellectual property rights
or otherwise. As a condition to exercising the rights and licenses
granted hereunder, each Recipient hereby assumes sole responsibility to
secure any other intellectual property rights needed, if any. For
example, if a third party patent license is required to allow Recipient
to distribute the Program, it is Recipient's responsibility to acquire
that license before distributing the Program.

d) Each Contributor represents that to its knowledge it
has sufficient copyright rights in its Contribution, if any, to grant
the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor

includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF

LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

</body>

</html>
Common Public License Version 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that

Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.
Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each

Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the

Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the

responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

```
digraph G {
 rankdir="LR";
 label = "label demo";
 labeljust="left";
 subgraph cluster0 {
   a;
   b;
   a->b;
   label = "cluster0";
   labelloc="bottom"
 }
 subgraph cluster1 {
   c;
   d;
   c->d;
   label = "cluster1";
   labelloc="top"
 }
}
digraph G {
 rankdir="LR";
 label = "label demo";
 subgraph cluster0 {
   a;
   b;
   a->b;
   label = "cluster0";
   labeljust="left";
 }
 subgraph cluster1 {
```

```
  c;
  d;
  c->d;
  label = "cluster1";
 }
}
```

This package was originally debianized by Stephen M Moraco <stephen@debian.org>
on Fri, 31 Dec 1999 15:37:17 -0700.

It was downloaded from http://www.graphviz.org/

Upstream Authors:

John Ellson, Lucent (TCL/tk and codegen extensions)
Emden Gansner, AT&T Research (dot, neato)
Eleftherios Koutsofios, AT&T Research (dotty, lefty, spline-o-matic)
John Mocenigo, AT&T Research (Grappa)
Stephen North, AT&T Research (dot, neato)
Gordon Woodhull (Montage)
and
David Dobkin (spline-o-matic)
Vladimir Alexiev (codegen improvements)

Copyright:

Copyright 1994-2007 AT&T Corp.

License:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b)

above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The Debian packaging was started by Stephen M Moraco <stephen@debian.org> and
then was taken over in late 2006 by Cyril Brulebois, placing his work under the
same terms as graphviz itself.
2006-2008, Cyril Brulebois <cyril.brulebois@enst-bretagne.fr>

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that neither the name of Emin
Martinian nor the names of any contributors are be used to endorse or
promote products derived from this software without specific prior
written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your

freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.
^L
Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that

any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes
a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and

modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.
^L

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation
and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.
^L
Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file

that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.
^L
6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system,

rather than copying library functions into the executable, and (2)
will operate properly with a modified version of the library, if
the user installs one, as long as the modified version is
interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at
least three years, to give the same user the materials
specified in Subsection 6a, above, for a charge no more
than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy
from a designated place, offer equivalent access to copy the above
specified materials from the same place.

e) Verify that the user has already received a copy of these
materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the
Library" must include any data and utility programs needed for
reproducing the executable from it. However, as a special exception,
the materials to be distributed need not include anything that is
normally distributed (in either source or binary form) with the major
components (compiler, kernel, and so on) of the operating system on
which the executable runs, unless that component itself accompanies
the executable.

It may happen that this requirement contradicts the license
restrictions of other proprietary libraries that do not normally
accompany the operating system. Such a contradiction means you cannot
use both them and the Library together in an executable that you
distribute.
^L
7. You may place library facilities that are a work based on the
Library side-by-side in a single library together with other library
facilities not covered by this License, and distribute such a combined
library, provided that the separate distribution of the work based on
the Library and of the other library facilities is otherwise
permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work
based on the Library, uncombined with any other library
facilities. This must be distributed under the terms of the
Sections above.

b) Give prominent notice with the combined library of the fact
that part of it is a work based on the Library, and explaining
where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.
^L
11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made

generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.
^L
14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE

LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

^L

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.14 httpd-tools 2.4.6 :67.el7_4.6

1.14.1 Available under license :

```
# GENERATED FROM XML -- DO NOT EDIT

URI: license.html.en
Content-Language: en
Content-type: text/html; charset=ISO-8859-1
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en"><head><!--
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
          This file is generated from xml source: DO NOT EDIT
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
X
    -->
<title>The Apache License, Version 2.0 - Apache HTTP Server</title>
<link href="./style/css/manual.css" rel="stylesheet" media="all" type="text/css" title="Main stylesheet" />
<link href="./style/css/manual-loose-100pc.css" rel="alternate stylesheet" media="all" type="text/css" title="No
Sidebar - Default font size" />
<link href="./style/css/manual-print.css" rel="stylesheet" media="print" type="text/css" /><link rel="stylesheet"
type="text/css" href="./style/css/prettify.css" />
<script src="./style/scripts/prettify.js" type="text/javascript">
</script>

<link href="./images/favicon.ico" rel="shortcut icon" /></head>
<body id="manual-page" class="no-sidebar"><div id="page-header">
```

Modules | Directives | FAQ | Glossary | Sitemap
Apache HTTP Server Version 2.4

<img title="<-" alt="<-" src="./images/left.gif" />

Apache > HTTP Server > Documentation > Version 2.4 <h1>The Apache License, Version 2.0</h1>

Available Languages: en

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

Definitions

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity
on behalf of whom a Contribution has been received by Licensor and
subsequently incorporated within the Work.

Grant of Copyright License. Subject to the terms
and conditions of this License, each Contributor hereby grants to You
a perpetual, worldwide, non-exclusive, no-charge, royalty-free,
irrevocable copyright license to reproduce, prepare Derivative Works of,
publicly display, publicly perform, sublicense, and distribute the
Work and such Derivative Works in Source or Object form.

Grant of Patent License. Subject to the terms
and conditions of this License, each Contributor hereby grants to You a
perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable
(except as stated in this section) patent license to make, have made,
use, offer to sell, sell, import, and otherwise transfer the Work,
where such license applies only to those patent claims licensable
by such Contributor that are necessarily infringed by their
Contribution(s) alone or by combination of their Contribution(s)
with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a
cross-claim or counterclaim in a lawsuit) alleging that the Work
or a Contribution incorporated within the Work constitutes direct
or contributory patent infringement, then any patent licenses
granted to You under this License for that Work shall terminate
as of the date such litigation is filed.

Redistribution. You may reproduce and distribute
copies of the Work or Derivative Works thereof in any medium, with or
without modifications, and in Source or Object form, provided that You
meet the following conditions:

<ol class="lo-A">
  You must give any other recipients of the Work or
  Derivative Works a copy of this License; and

  You must cause any modified files to carry prominent notices
  stating that You changed the files; and

  You must retain, in the Source form of any Derivative Works
  that You distribute, all copyright, patent, trademark, and
  attribution notices from the Source form of the Work,
  excluding those notices that do not pertain to any part of
  the Derivative Works; and

  If the Work includes a "NOTICE" text file as part of its
  distribution, then any Derivative Works that You distribute must
  include a readable copy of the attribution notices contained
  within such NOTICE file, excluding those notices that do not
  pertain to any part of the Derivative Works, in at least one
  of the following places: within a NOTICE text file distributed
  as part of the Derivative Works; within the Source form or
  documentation, if provided along with the Derivative Works; or,
  within a display generated by the Derivative Works, if and
  wherever such third-party notices normally appear. The contents
  of the NOTICE file are for informational purposes only and
  do not modify the License. You may add Your own attribution
  notices within Derivative Works that You distribute, alongside
  or as an addendum to the NOTICE text from the Work, provided
  that such additional attribution notices cannot be construed
  as modifying the License.

<p>You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.</p>

Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

```
  </ol>

  <p class="centered">END OF TERMS AND CONDITIONS</p>

  <p class="centered">APPENDIX: How to apply the Apache License to your
  work.</p>

  <p>To apply the Apache License to your work, attach the following
  boilerplate notice, with the fields enclosed by brackets "[]"
  replaced with your own identifying information. (Don't include
  the brackets!)  The text should be enclosed in the appropriate
  comment syntax for the file format. We also recommend that a
  file or class name and description of purpose be included on the
  same "printed page" as the copyright notice for easier
  identification within third-party archives.</p>

  <div class="example"><pre>Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

  http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.</pre></div>
</div>
</div>
<div class="bottomlang">
<p><span>Available Languages: </span><a href="./en/license.html" title="English"> en </a></p>
</div><div class="top"><a href="#page-header"><img src="./images/up.gif" alt="top" /></a></div><div
class="section"><h2><a id="comments_section" name="comments_section">Comments</a></h2><div
class="warning"><strong>Notice:</strong><br />This is not a Q&A section. Comments placed here should be
pointed towards suggestions on improving the documentation or server, and may be removed again by our
moderators if they are either implemented or considered invalid/off-topic. Questions on how to manage the Apache
HTTP Server should be directed at either our IRC channel, #httpd, on Freenode, or sent to our <a
href="http://httpd.apache.org/lists.html">mailing lists</a>.</div>
<script type="text/javascript"><!--//--><![CDATA[//><!--
var comments_shortname = 'httpd';
var comments_identifier = 'http://httpd.apache.org/docs/2.4/license.html';
(function(w, d) {
  if (w.location.hostname.toLowerCase() == "httpd.apache.org") {
     d.write('<div id="comments_thread"><\/div>');
     var s = d.createElement('script');
     s.type = 'text/javascript';
```

```
      s.async = true;
      s.src = 'https://comments.apache.org/show_comments.lua?site=' + comments_shortname + '&page=' +
comments_identifier;
      (d.getElementsByTagName('head')[0] || d.getElementsByTagName('body')[0]).appendChild(s);
   }
   else {
      d.write('<div id="comments_thread">Comments are disabled for this page at the moment.<\/div>');
   }
})(window, document);
//--><![]]></script></div><div id="footer">
<p class="apache">Copyright 2013 The Apache Software Foundation.<br />Licensed under the <a
href="http://www.apache.org/licenses/LICENSE-2.0">Apache License, Version 2.0</a>.</p>
<p class="menu"><a href="./mod/">Modules</a> | <a href="./mod/directives.html">Directives</a> | <a
href="http://wiki.apache.org/httpd/FAQ">FAQ</a> | <a href="./glossary.html">Glossary</a> | <a
href="./sitemap.html">Sitemap</a></p></div><script type="text/javascript"><!--//--><![CDATA[//><!--
if (typeof(prettyPrint) !== 'undefined') {
   prettyPrint();
}
//--><![]]></script>
</body></html>
```

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache HTTP Server
Copyright 2013 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Portions of this software were developed at the National Center
for Supercomputing Applications (NCSA) at the University of
Illinois at Urbana-Champaign.

This software contains code derived from the RSA Data Security
Inc. MD5 Message-Digest Algorithm, including various
modifications by Spyglass Inc., Carnegie Mellon University, and
Bell Communications Research, Inc (Bellcore).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

APACHE HTTP SERVER SUBCOMPONENTS:

The Apache HTTP Server includes a number of subcomponents with
separate copyright notices and license terms. Your use of the source
code for the these subcomponents is subject to the terms and
conditions of the following licenses.

For the mod_mime_magic component:

```
/*
* mod_mime_magic: MIME type lookup via file magic numbers
* Copyright (c) 1996-1997 Cisco Systems, Inc.
*
* This software was submitted by Cisco Systems to the Apache Group in July
* 1997.  Future revisions and derivatives of this source code must
* acknowledge Cisco Systems as the original contributor of this module.
* All other licensing and usage conditions are those of the Apache Group.
*
* Some of this code is derived from the free version of the file command
* originally posted to comp.sources.unix.  Copyright info for that program
* is included below as required.
* ---------------------------------------------------------------------------
* - Copyright (c) Ian F. Darwin, 1987. Written by Ian F. Darwin.
*
* This software is not subject to any license of the American Telephone and
* Telegraph Company or of the Regents of the University of California.
*
* Permission is granted to anyone to use this software for any purpose on any
* computer system, and to alter it and redistribute it freely, subject to
* the following restrictions:
*
* 1. The author is not responsible for the consequences of use of this
* software, no matter how awful, even if they arise from flaws in it.
*
* 2. The origin of this software must not be misrepresented, either by
* explicit claim or by omission.  Since few users ever read sources, credits
* must appear in the documentation.
*
* 3. Altered versions must be plainly marked as such, and must not be
* misrepresented as being the original software.  Since few users ever read
* sources, credits must appear in the documentation.
```

```
*
* 4. This notice may not be removed or altered.
* -------------------------------------------------------------------------
*
*/
```

For the modules\mappers\mod_imagemap.c component:

"macmartinized" polygon code copyright 1992 by Eric Haines, erich@eye.com

For the server\util_md5.c component:

```
/*****************************************************************************
* NCSA HTTPd Server
* Software Development Group
* National Center for Supercomputing Applications
* University of Illinois at Urbana-Champaign
* 605 E. Springfield, Champaign, IL 61820
* httpd@ncsa.uiuc.edu
*
* Copyright (C) 1995, Board of Trustees of the University of Illinois
*
*****************************************************************************
*
* md5.c: NCSA HTTPd code which uses the md5c.c RSA Code
*
*  Original Code Copyright (C) 1994, Jeff Hostetler, Spyglass, Inc.
*  Portions of Content-MD5 code Copyright (C) 1993, 1994 by Carnegie Mellon
*     University (see Copyright below).
*  Portions of Content-MD5 code Copyright (C) 1991 Bell Communications
*     Research, Inc. (Bellcore) (see Copyright below).
*  Portions extracted from mpack, John G. Myers - jgm+@cmu.edu
*  Content-MD5 Code contributed by Martin Hamilton (martin@net.lut.ac.uk)
*
*/


/* these portions extracted from mpack, John G. Myers - jgm+@cmu.edu */
/* (C) Copyright 1993,1994 by Carnegie Mellon University
* All Rights Reserved.
*
* Permission to use, copy, modify, distribute, and sell this software
* and its documentation for any purpose is hereby granted without
* fee, provided that the above copyright notice appear in all copies
* and that both that copyright notice and this permission notice
* appear in supporting documentation, and that the name of Carnegie
* Mellon University not be used in advertising or publicity
```

* pertaining to distribution of the software without specific,
* written prior permission. Carnegie Mellon University makes no
* representations about the suitability of this software for any
* purpose. It is provided "as is" without express or implied
* warranty.
*
* CARNEGIE MELLON UNIVERSITY DISCLAIMS ALL WARRANTIES WITH REGARD TO
* THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY
* AND FITNESS, IN NO EVENT SHALL CARNEGIE MELLON UNIVERSITY BE LIABLE
* FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN
* AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING
* OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
* SOFTWARE.
*/

/*
* Copyright (c) 1991 Bell Communications Research, Inc. (Bellcore)
*
* Permission to use, copy, modify, and distribute this material
* for any purpose and without fee is hereby granted, provided
* that the above copyright notice and this permission notice
* appear in all copies, and that the name of Bellcore not be
* used in advertising or publicity pertaining to this
* material without the specific, prior written permission
* of an authorized representative of Bellcore. BELLCORE
* MAKES NO REPRESENTATIONS ABOUT THE ACCURACY OR SUITABILITY
* OF THIS MATERIAL FOR ANY PURPOSE. IT IS PROVIDED "AS IS",
* WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES.
*/

For the srclib\apr\include\apr_md5.h component:
/*
* This is work is derived from material Copyright RSA Data Security, Inc.
*
* The RSA copyright statement and Licence for that original material is
* included below. This is followed by the Apache copyright statement and
* licence for the modifications made to that material.
*/

/* Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All
  rights reserved.

  License to copy and use this software is granted provided that it
  is identified as the "RSA Data Security, Inc. MD5 Message-Digest
  Algorithm" in all material mentioning or referencing this software
  or this function.

License is also granted to make and use derivative works provided
that such works are identified as "derived from the RSA Data
Security, Inc. MD5 Message-Digest Algorithm" in all material
mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is"
without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
documentation and/or software.
*/

For the srclib\apr\passwd\apr_md5.c component:

/*
* This is work is derived from material Copyright RSA Data Security, Inc.
*
* The RSA copyright statement and Licence for that original material is
* included below. This is followed by the Apache copyright statement and
* licence for the modifications made to that material.
*/

/* MD5C.C - RSA Data Security, Inc., MD5 message-digest algorithm
*/

/* Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All
rights reserved.

License to copy and use this software is granted provided that it
is identified as the "RSA Data Security, Inc. MD5 Message-Digest
Algorithm" in all material mentioning or referencing this software
or this function.

License is also granted to make and use derivative works provided
that such works are identified as "derived from the RSA Data
Security, Inc. MD5 Message-Digest Algorithm" in all material
mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either
the merchantability of this software or the suitability of this
software for any particular purpose. It is provided "as is"
without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this
documentation and/or software.
*/

```
/*
* The apr_md5_encode() routine uses much code obtained from the FreeBSD 3.0
* MD5 crypt() function, which is licenced as follows:
* ----------------------------------------------------------------------------
* "THE BEER-WARE LICENSE" (Revision 42):
* <phk@login.dknet.dk> wrote this file.  As long as you retain this notice you
* can do whatever you want with this stuff. If we meet some day, and you think
* this stuff is worth it, you can buy me a beer in return.  Poul-Henning Kamp
* ----------------------------------------------------------------------------
*/
```

For the srclib\apr-util\crypto\apr_md4.c component:

```
* This is derived from material copyright RSA Data Security, Inc.
* Their notice is reproduced below in its entirety.
*
* Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All
* rights reserved.
*
* License to copy and use this software is granted provided that it
* is identified as the "RSA Data Security, Inc. MD4 Message-Digest
* Algorithm" in all material mentioning or referencing this software
* or this function.
*
* License is also granted to make and use derivative works provided
* that such works are identified as "derived from the RSA Data
* Security, Inc. MD4 Message-Digest Algorithm" in all material
* mentioning or referencing the derived work.
*
* RSA Data Security, Inc. makes no representations concerning either
* the merchantability of this software or the suitability of this
* software for any particular purpose. It is provided "as is"
* without express or implied warranty of any kind.
*
* These notices must be retained in any copies of any part of this
* documentation and/or software.
*/
```

For the srclib\apr-util\include\apr_md4.h component:

```
*
* This is derived from material copyright RSA Data Security, Inc.
* Their notice is reproduced below in its entirety.
*
* Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All
* rights reserved.
*
* License to copy and use this software is granted provided that it
```

* is identified as the "RSA Data Security, Inc. MD4 Message-Digest
* Algorithm" in all material mentioning or referencing this software
* or this function.
*
* License is also granted to make and use derivative works provided
* that such works are identified as "derived from the RSA Data
* Security, Inc. MD4 Message-Digest Algorithm" in all material
* mentioning or referencing the derived work.
*
* RSA Data Security, Inc. makes no representations concerning either
* the merchantability of this software or the suitability of this
* software for any particular purpose. It is provided "as is"
* without express or implied warranty of any kind.
*
* These notices must be retained in any copies of any part of this
* documentation and/or software.
*/

For the srclib\apr-util\test\testmd4.c component:

*
* This is derived from material copyright RSA Data Security, Inc.
* Their notice is reproduced below in its entirety.
*
* Copyright (C) 1990-2, RSA Data Security, Inc. Created 1990. All
* rights reserved.
*
* RSA Data Security, Inc. makes no representations concerning either
* the merchantability of this software or the suitability of this
* software for any particular purpose. It is provided "as is"
* without express or implied warranty of any kind.
*
* These notices must be retained in any copies of any part of this
* documentation and/or software.
*/

For the srclib\apr-util\xml\expat\conftools\install-sh component:

#
install - install a program, script, or datafile
This comes from X11R5 (mit/util/scripts/install.sh).
#
Copyright 1991 by the Massachusetts Institute of Technology
#
Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that

copyright notice and this permission notice appear in supporting
documentation, and that the name of M.I.T. not be used in advertising or
publicity pertaining to distribution of the software without specific,
written prior permission. M.I.T. makes no representations about the
suitability of this software for any purpose. It is provided "as is"
without express or implied warranty.
#

For the test\zb.c component:

/* ZeusBench V1.01
    ===============

This program is Copyright (C) Zeus Technology Limited 1996.

This program may be used and copied freely providing this copyright notice
is not removed.

This software is provided "as is" and any express or implied waranties,
including but not limited to, the implied warranties of merchantability and
fitness for a particular purpose are disclaimed. In no event shall
Zeus Technology Ltd. be liable for any direct, indirect, incidental, special,
exemplary, or consequential damaged (including, but not limited to,
procurement of substitute good or services; loss of use, data, or profits;
or business interruption) however caused and on theory of liability. Whether
in contract, strict liability or tort (including negligence or otherwise)
arising in any way out of the use of this software, even if advised of the
possibility of such damage.

  Written by Adam Twiss (adam@zeus.co.uk). March 1996

Thanks to the following people for their input:
 Mike Belshe (mbelshe@netscape.com)
 Michael Campanella (campanella@stevms.enet.dec.com)

*/

For the expat xml parser component:

Copyright (c) 1998, 1999, 2000 Thai Open Source Software Center Ltd
  and Clark Cooper

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to

the following conditions:

The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

==========

1.15 httpd-tools/GPLv3 2.4.6 :67.el7_4.6

1.15.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you
want it, that you can change the software or use pieces of it in new
free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you

these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of

works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major

Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the

written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the

terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work

in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this

License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE

USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

1.16 kafka 1.3.5

1.16.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2015 David Arthur

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.17 kibana 6.5.4

1.17.1 Available under license :

```
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { getBundledNotices } from './bundled_notices';

const concatNotices = notices => notices.map(notice => notice.text).join('\n');

export async function generatePackageNoticeText(pkg) {
 const bundledNotices = concatNotices(await getBundledNotices(pkg.directory));

 const intro = `This product bundles ${pkg.name}@${pkg.version}`;
 const license = ` which is available under ${
   pkg.licenses.length > 1
     ? `the\n"${pkg.licenses.join('", ')} licenses.`
     : `a\n"${pkg.licenses[0]}" license.`
 }`;

 const moreInfo = bundledNotices
   ? `\n${bundledNotices}\n`
   : `  For details, see ${pkg.relative}/.`;

 return `${intro}${license}${moreInfo}`;
}
MIT
{
 "name": "bar",
 "version": "1.0.0",
 "license": "MIT"
}
Notice me!
```

```
{
 "name": "foo",
 "version": "1.0.0",
 "license": "Apache 2.0"
}
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { resolve } from 'path';
import { readFileSync } from 'fs';

import { generateBuildNoticeText } from './generate_build_notice_text';

const NODE_MODULES = resolve(__dirname, '__fixtures__/node_modules');
const NODE_DIR = resolve(process.execPath, '../..');
const NODE_VERSION = '8.11.3';
const PACKAGES = [
 {
   name: 'bar',
   version: '1.0.0',
   licenses: ['MIT'],
   directory: resolve(NODE_MODULES, 'bar'),
   relative: 'node_modules/bar',
 },
 {
   name: 'foo',
   version: '1.0.0',
   licenses: ['Apache 2.0'],
   directory: resolve(NODE_MODULES, 'foo'),
   relative: 'node_modules/foo',
 },
];
```

```
describe('src/dev/build/tasks/notice_file/generate_notice', () => {
 describe('generateBuildNoticeText()', () => {
  let notice;
  beforeAll(
    async () =>
      (notice = await generateBuildNoticeText({
        packages: PACKAGES,
        nodeDir: NODE_DIR,
        nodeVersion: NODE_VERSION,
        noticeFromSource: 'NOTICE_FROM_SOURCE\n'
      }))
  );

  it('returns a string', () => {
    expect(typeof notice).toBe('string');
  });

  it('includes *NOTICE* files from packages', () => {
    expect(notice).toEqual(expect.stringContaining(
      readFileSync(resolve(NODE_MODULES, 'foo/NOTICE.txt'), 'utf8')
    ));
  });

  it('includes *LICENSE* files from packages', () => {
    expect(notice).toEqual(expect.stringContaining(
      readFileSync(
        resolve(NODE_MODULES, 'bar/LICENSE.md'),
        'utf8'
      )
    ));
  });

  it('includes the LICENSE file from node', () => {
    expect(notice).toEqual(expect.stringContaining(
      readFileSync(resolve(NODE_DIR, 'LICENSE'), 'utf8')
    ));
  });

  it('includes node version', () => {
    expect(notice).toEqual(expect.stringContaining(
      'This product bundles Node.js v8.11.3'
    ));
  });

  it('includes the noticeFromSource', () => {
    expect(notice).toEqual(expect.stringContaining(
      'NOTICE_FROM_SOURCE'
```

```
    ));
  });
 });
});
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { resolve } from 'path';
import { readFile } from 'fs';

import { fromNode as fcb } from 'bluebird';
import glob from 'glob';

export async function getBundledNotices(packageDirectory) {
 const pattern = resolve(packageDirectory, '*{LICENSE,NOTICE}*');
 const paths = await fcb(cb => glob(pattern, cb));
 return Promise.all(
   paths.map(async path => ({
     path,
     text: await fcb(cb => readFile(path, 'utf8', cb)),
   }))
 );
}
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
```

```
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { resolve } from 'path';
import { readFileSync } from 'fs';

export function generateNodeNoticeText(nodeDir, nodeVersion) {
 const licensePath = resolve(nodeDir, 'LICENSE');
 const license = readFileSync(licensePath, 'utf8');
 return `This product bundles Node.js v${nodeVersion}.\n\n${license}`;
}
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { readFileSync, writeFileSync } from 'fs';
import { resolve } from 'path';

import getopts from 'getopts';
import dedent from 'dedent';
import { ToolingLog, pickLevelFromFlags } from '@kbn/dev-utils';

import { REPO_ROOT } from '../constants';
import { generateNoticeFromSource } from './generate_notice_from_source';

const unknownFlags = [];
```

```
const opts = getopts(process.argv.slice(2), {
 boolean: [
   'help',
   'validate',
   'verbose',
   'debug',
 ],
 unknown(flag) {
   unknownFlags.push(flag);
 }
});

const log = new ToolingLog({
 level: pickLevelFromFlags(opts),
 writeTo: process.stdout
});

if (unknownFlags.length) {
 log.error(`Unknown flags ${unknownFlags.map(f => `"${f}"`).join(',')}`);
 process.exitCode = 1;
 opts.help = true;
}

if (opts.help) {
 process.stdout.write('\n' + dedent`
   Generate or validate NOTICE.txt.

     Entries are collected by finding all multi-line comments that start
     with a "@notice" tag and copying their text content into NOTICE.txt.

   Options:
     --help      Show this help info
     --validate  Don't write the NOTICE.txt, just fail if updates would have been made
     --verbose   Set logging level to verbose
     --debug     Set logging level to debug
 ` + '\n\n');
 process.exit();
}

(async function run() {
 const path = resolve(REPO_ROOT, 'NOTICE.txt');
 const newContent = await generateNoticeFromSource({
   productName: 'Kibana source code with Kibana X-Pack source code',
   directory: REPO_ROOT,
   log,
 });

 const currentContent = readFileSync(path, 'utf8');
```

```
 if (currentContent === newContent) {
   log.success('NOTICE.txt is up to date');
   return;
 }

 if (!opts.validate) {
   log.success('Wrote notice text to NOTICE.txt');
   writeFileSync(path, newContent, 'utf8');
   return;
 }

 log.error('NOTICE.txt is out of date, run `node scripts/notice` to update and commit the results.');
 process.exit(1);
}()).catch(error => {
 log.error(error);
 process.exit(1);
});
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

export { generateNoticeFromSource } from './generate_notice_from_source';
export { generateBuildNoticeText } from './generate_build_notice_text';
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
```

```
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import vfs from 'vinyl-fs';

const NOTICE_COMMENT_RE = /\/\*[\s\n\*]*@notice([\w\W]+?)\*\/g;
const NEWLINE_RE = /\r?\n/g;

/**
* Generates the text for the NOTICE.txt file at the root of the
* repo which details the licenses for code that is copied/vendored
* into the repository.
*
* @param  {Object} options
* @property {string} options.productName Name to print at the top of the notice
* @property {ToolingLog} options.log
* @property {string} options.directory absolute path to the repo to search for @notice comments
* @return {string}
*/
export async function generateNoticeFromSource({ productName, directory, log }) {
 const globs = [
   '**/*.{js,less,css,ts}',
 ];

 const options = {
   cwd: directory,
   nodir: true,
   ignore: [
     '{node_modules,build,target,dist,optimize}/**',
     'packages/*/{node_modules,build,target,dist}/**',
     'x-pack/{node_modules,build,target,dist,optimize}/**',
     'x-pack/packages/*/{node_modules,build,target,dist}/**',
   ]
 };

 log.debug('vfs.src globs', globs);
 log.debug('vfs.src options', options);
 log.info(`Searching ${directory} for multi-line comments starting with @notice`);

 const files = vfs.src(globs, options);
 const noticeComments = [];
 await new Promise((resolve, reject) => {
```

```
  files
    .on('data', (file) => {
      log.verbose(`Checking for @notice comments in ${file.relative}`);

      const source = file.contents.toString('utf8');
      let match;
      while ((match = NOTICE_COMMENT_RE.exec(source)) !== null) {
        log.info(`Found @notice comment in ${file.relative}`);
        noticeComments.push(match[1]);
      }
    })
    .on('error', reject)
    .on('end', resolve);
});

let noticeText = '';
noticeText += `${productName}\n`;
noticeText += `Copyright 2012-${(new Date()).getUTCFullYear()} Elasticsearch B.V.\n`;

for (const comment of noticeComments.sort()) {
  noticeText += '\n---\n';
  noticeText += comment
    .split(NEWLINE_RE)
    .map(line => (
      line
        // trim whitespace
        .trim()
        // trim leading * and a single space
        .replace(/(^\* ?)/, '')
    ))
    .join('\n')
    .trim();
  noticeText += '\n';
}

noticeText += '\n';

log.debug(`notice text:\n\n${noticeText}`);
return noticeText;
}
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
```

```
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { write, read } from '../lib';

export const UpdateLicenseFileTask = {
 description: 'Updating LICENSE.txt file',

 async run(config, log, build) {
   if (build.isOss()) {
     log.info('Copying Apache 2.0 license to LICENSE.txt');
     await write(
       build.resolvePath('LICENSE.txt'),
       await read(config.resolveFromRepo('licenses/APACHE-LICENSE-2.0.txt'))
     );
   } else {
     log.info('Copying Elastic license to LICENSE.txt');
     await write(
       build.resolvePath('LICENSE.txt'),
       await read(config.resolveFromRepo('licenses/ELASTIC-LICENSE.txt'))
     );
   }
 },
};
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
```

```
* under the License.
*/

import licenseChecker from 'license-checker';

export function callLicenseChecker(options = {}) {
 const {
   directory,
   dev = false
 } = options;

 if (!directory) {
   throw new Error('You must specify the directory where license checker should start');
 }

 return new Promise((resolve, reject) => {
   licenseChecker.init({
     start: directory,
     production: !dev,
     json: true,
     customFormat: {
       realPath: true,
       licenseText: false,
       licenseFile: false
     }
   }, (err, licenseInfo) => {
     if (err) reject(err);
     else resolve(licenseInfo);
   });
 });
}
```

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The MIT License (MIT)

Copyright (c) 2014 Steven Skelton

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to

use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
the Software, and to permit persons to whom the Software is furnished to do so,
subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS
FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR
COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { getInstalledPackages } from '../src/dev/npm';
import {
 assertLicensesValid,
 LICENSE_WHITELIST,
 LICENSE_OVERRIDES,
} from '../src/dev/license_checker';

export default function licenses(grunt) {
 grunt.registerTask('licenses', 'Checks dependency licenses', async function () {
   const done = this.async();

   try {
     const dev = Boolean(grunt.option('dev'));

     assertLicensesValid({
       packages: await getInstalledPackages({
```

```
        directory: grunt.config.get('root'),
        licenseOverrides: LICENSE_OVERRIDES,
        dev
      }),
      validLicenses: LICENSE_WHITELIST
    });
    done();
  } catch (err) {
    grunt.fail.fatal(err);
    done(err);
  }
});
}
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { writeFileSync } from 'fs';
import { resolve } from 'path';
import { getInstalledPackages } from '../src/dev/npm';
import { LICENSE_OVERRIDES } from '../src/dev/license_checker';

import {
 isNull,
 isUndefined
} from 'lodash';

const allDoubleQuoteRE = /"/g;

function escapeValue(value) {
 if (isNull(value)) {
   return;
 }
```

```
 return `"${value.replace(allDoubleQuoteRE, """)}"`;
}

function formatCsvValues(fields, values) {
 return fields.map((field) => {
   const value = values[field];

   if (isNull(value) || isUndefined(value)) {
     return null;
   }

   return value.toString();
 })
   .map(escapeValue)
   .join(',');
}

export default function licensesCSVReport(grunt) {
 grunt.registerTask('licenses:csv_report', 'Report of 3rd party dependencies', async function () {
   const fields = ['name', 'version', 'url', 'license'];
   const done = this.async();

   try {
     const file = grunt.option('csv');
     const directory = grunt.option('directory');
     const dev = Boolean(grunt.option('dev'));

     const packages = await getInstalledPackages({
       directory: directory ? resolve(directory) : grunt.config.get('root'),
       licenseOverrides: LICENSE_OVERRIDES,
       dev
     });

     const csv = packages.map(pkg => {
       const data = {
         name: pkg.name,
         version: pkg.version,
         url: pkg.repository || `https://www.npmjs.com/package/${pkg.name}`,
         license: pkg.licenses.join(',')
       };

       return formatCsvValues(fields, data);
     }).join('\n');

     if (file) {
       writeFileSync(file, `${fields.join(',')}\n${csv}`);
       grunt.log.writeln(`wrote to ${file}`);
```

```
    } else {
      grunt.log.writeln(csv);
      grunt.log.writeln('\nspecify "--csv [filepath]" to write the data to a specific file');
    }

    done();
  } catch (err) {
    grunt.fail.fatal(err);
    done(err);
  }
});
}
```

Kibana source code with Kibana X-Pack source code
Copyright 2012-2018 Elasticsearch B.V.

This product includes code that was extracted from angular-ui-bootstrap@0.13.1
which is available under an "MIT" license

The MIT License

Copyright (c) 2012-2016 the AngularUI Team, http://angular-ui.github.io/bootstrap/

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product uses Noto fonts that are licensed under the SIL Open
Font License, Version 1.1.

Pretty handling of logarithmic axes.
Copyright (c) 2007-2014 IOLA and Ole Laursen.

Licensed under the MIT license.
Created by Arne de Laat
Set axis.mode to "log" and make the axis logarithmic using transform:

```
  axis: {
      mode: 'log',
      transform: function(v) {v <= 0 ? Math.log(v) / Math.LN10 : null},
      inverseTransform: function(v) {Math.pow(10, v)}
  }
```

The transform filters negative and zero values, because those are
invalid on logarithmic scales.
This plugin tries to create good looking logarithmic ticks, using
unicode superscript characters. If all data to be plotted is between two
powers of ten then the default flot tick generator and renderer are
used. Logarithmic ticks are places at powers of ten and at half those
values if there are not to many ticks already (e.g. [1, 5, 10, 50, 100]).
For details, see https://github.com/flot/flot/pull/1328

This product bundles angular-ui-bootstrap@0.12.1 which is available under a
"MIT" license.

The MIT License

Copyright (c) 2012-2014 the AngularUI Team, https://github.com/organizations/angular-ui/teams/291112

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product bundles bootstrap@3.3.6 which is available under a
"MIT" license.

The MIT License (MIT)

Copyright (c) 2011-2015 Twitter, Inc

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

This product bundles childnode-remove which is available under a
"MIT" license.

The MIT License (MIT)

Copyright (c) 2016-present, jszhou
https://github.com/jserz/js_piece

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

This product bundles geohash.js which is available under a
"MIT" license. For details, see src/ui/public/utils/decode_geo_hash.js.

This product includes code that is based on flot-charts, which was available
under a "MIT" license.

The MIT License (MIT)

Copyright (c) 2007-2014 IOLA and Ole Laursen

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product includes code that was extracted from angular@1.3.
Original license:
The MIT License

Copyright (c) 2010-2014 Google, Inc. http://angularjs.org

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

require('../src/setup_node_env');
require('../src/dev/notice/cli');
```

Source code in this repository is variously licensed under the Apache License Version 2.0, an Apache compatible license, or the Elastic License. Outside of the "x-pack" folder, source code in a given file is licensed under the Apache License Version 2.0, unless otherwise noted at the beginning of the file or a LICENSE file present in the directory subtree declares a separate license. Within the "x-pack" folder, source code in a given file is licensed under the Elastic License, unless otherwise noted at the beginning of the file or a LICENSE file present in the directory subtree declares a separate license.

The build produces two sets of binaries - one set that falls under the Elastic License and another set that falls under Apache License Version 2.0. The binaries that contain `-oss` in the artifact name are licensed under the Apache License Version 2.0.

```
.kuiNotice {
 padding: 40px 60px 48px;
 margin: 20px;
 background-color: $noticeBackgroundColor;
 line-height: $kuiLineHeight;
}
$noticeBackgroundColor: #FFF;
$noticeTitleFontSize: 22px;
```

```
@import "notice";
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import React from 'react';

import {
 GuideDemo,
 GuideSandbox,
 GuideSandboxCodeToggle,
 GuideSectionTypes,
} from '../../components';

const html = require('./notice_sandbox.html');

export default props => (
 <GuideSandbox>
   <GuideDemo
     isFullScreen={true}
     html={html}
   />

   <GuideSandboxCodeToggle
     source={[{
       type: GuideSectionTypes.HTML,
       code: html,
     }]}
     title={props.route.name}
   />
 </GuideSandbox>
);
```

```
<div
 class="kuiView"
 style="background-color: pink"
>
 <div
   class="kuiViewContent kuiViewContent--constrainedWidth"
   style="background-color: pink"
 >
   <div class="kuiNotice">
     <h1 class="kuiTitle kuiVerticalRhythm">
       <span class="kuiIcon kuiIcon--error fa-warning"></span>
       Notice title
     </h1>

     <p class="kuiText kuiVerticalRhythm">
       You can use the Notice component to present content in the center of the content area,
       within a container with a narrow, fixed width.
     </p>

     <p class="kuiText kuiVerticalRhythm">
       The pink color is just for demonstration purposes.
     </p>
   </div>
 </div>
</div>
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

exports.assert = function assert(truth, message) {
 if (truth) {
   return;
 }
```

```
 const error = new Error(message);
 error.failedAssertion = true;
 throw error;
};

exports.normalizeWhitespace = function normalizeWhitespace(string) {
 return string.replace(/\s+/g, ' ');
};

exports.init = function (context, program, initStep) {
 try {
   return initStep();
 } catch (error) {
   if (error.failedAssertion) {
     context.report({
       node: program,
       message: error.message
     });
   } else {
     throw error;
   }
 }
};
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

const { RuleTester } = require('eslint');
const rule = require('../disallow_license_headers');
const dedent = require('dedent');

const RULE_NAME = '@kbn/license-header/disallow-license-headers';
```

```
const ruleTester = new RuleTester({
 parser: 'babel-eslint',
 parserOptions: {
   ecmaVersion: 2015
 }
});
ruleTester.run(RULE_NAME, rule, {
 valid: [
   {
     code: dedent`
       /* license */

       console.log('foo')
     `,

     options: [{
       licenses: [
         '// license'
       ]
     }],
   },
   {
     code: dedent`
       // license

       console.log('foo')
     `,

     options: [{
       licenses: [
         '/* license */',
       ]
     }],
   }
 ],

 invalid: [
   // missing license option
   {
     code: dedent`
       console.log('foo')
     `,

     options: [],
     errors: [
       {
         message: '"licenses" option is required',
```

```
    }
  ]
},

// license cannot contain multiple block comments
{
  code: dedent`
    console.log('foo')
  `,

  options: [{
    licenses: [
      '/* one *//* two */'
    ]
  }],
  errors: [
    {
      message: "'licenses[0]" option must only include a single comment',
    }
  ]
},

// license cannot contain multiple line comments
{
  code: dedent`
    console.log('foo')
  `,

  options: [{
    licenses: [
      `// one\n// two`
    ]
  }],
  errors: [
    {
      message: "'licenses[0]" option must only include a single comment',
    }
  ]
},

// license cannot contain expressions
{
  code: dedent`
    console.log('foo')
  `,

  options: [{
    licenses: [
```

```
        '// old license',
        dedent`
          /* license */
          console.log('hello world');
        `
      ]
    }],
    errors: [
      {
        message: "'licenses[1]" option must only include a single comment',
      }
    ]
  },

  // license is not a single comment
  {
    code: dedent`
      console.log('foo')
    `,

    options: [{
      licenses: [
        '// old license',
        '// older license',
        `console.log('hello world');`
      ]
    }],
    errors: [
      {
        message: "'licenses[2]" option must only include a single comment',
      }
    ]
  },
 ]
});
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
```

```
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

const { RuleTester } = require('eslint');
const rule = require('../require_license_header');
const dedent = require('dedent');

const RULE_NAME = '@kbn/license-header/require-license-header';

const ruleTester = new RuleTester({
 parser: 'babel-eslint',
 parserOptions: {
   ecmaVersion: 2015
 }
});
ruleTester.run(RULE_NAME, rule, {
 valid: [
   {
     code: dedent`
       /* license */

       console.log('foo')
     `,

     options: [{ license: '/* license */' }],
   },
   {
     code: dedent`
       // license

       console.log('foo')
     `,

     options: [{ license: '// license' }],
   }
 ],

 invalid: [
   // missing license option
   {
     code: dedent`
       console.log('foo')
     `,

     options: [],
```

```
  errors: [
    {
      message: '"license" option is required',
    }
  ]
},

// content cannot contain multiple block comments
{
  code: dedent`
    console.log('foo')
  `,

  options: [
    { license: '/* one *//* two */' }
  ],
  errors: [
    {
      message: '"license" option must only include a single comment',
    }
  ]
},

// content cannot contain multiple line comments
{
  code: dedent`
    console.log('foo')
  `,

  options: [
    { license: `// one\n// two` }
  ],
  errors: [
    {
      message: '"license" option must only include a single comment',
    }
  ]
},

// content cannot contain expressions
{
  code: dedent`
    console.log('foo')
  `,

  options: [
    {
      license: dedent`
```

```
      /* license */
      console.log('hello world');
    `
    }
  ],
  errors: [
    {
      message: '"license" option must only include a single comment',
    }
  ]
},

// content is not a single comment
{
  code: dedent`
    console.log('foo')
  `,

  options: [
    { license: `console.log('hello world');` }
  ],
  errors: [
    {
      message: '"license" option must only include a single comment',
    }
  ]
},

// missing license header
{
  code: dedent`
    console.log('foo')
  `,

  options: [{ license: '/* license */' }],
  errors: [
    {
      message: 'File must start with a license header',
    }
  ],

  output: dedent`
    /* license */

    console.log('foo')
  `
},
```

```
  // strips newlines before the license comment
  {
    code: '\n\n' + dedent`
      /* license */

      console.log('foo')
    `,

    options: [{ license: '/* license */' }],
    errors: [
      {
        message: 'License header must be at the very beginning of the file',
      }
    ],

    output: dedent`
      /* license */

      console.log('foo')
    `
  },

  // moves license header before other nodes if necessary
  {
    code: dedent`
      /* not license */
      /* license */
      console.log('foo')
    `,

    options: [{ license: '/* license */' }],
    errors: [
      {
        message: 'License header must be at the very beginning of the file',
      }
    ],

    output: dedent`
      /* license */

      /* not license */

      console.log('foo')
    `
  },
]
});
```

```
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

const babelEslint = require('babel-eslint');

const { assert, normalizeWhitespace, init } = require('../lib');

module.exports = {
 meta: {
   fixable: 'code',
   schema: [{
     type: 'object',
     properties: {
       licenses: {
         type: 'array',
         items: {
           type: 'string'
         }
       },
     },
     additionalProperties: false,
   }]
 },
 create: context => {
   return {
     Program(program) {
       const licenses = init(context, program, () => {
         const options = context.options[0] || {};
         const licenses = options.licenses;

         assert(!!licenses, '"licenses" option is required');
```

```
        return licenses.map((license, i) => {
          const parsed = babelEslint.parse(license);

          assert(!parsed.body.length, `"licenses[${i}]" option must only include a single comment`);
          assert(parsed.comments.length === 1, `"licenses[${i}]" option must only include a single comment`);

          return normalizeWhitespace(parsed.comments[0].value);
        });
      });

      if (!licenses || !licenses.length) return;

      const sourceCode = context.getSourceCode();

      sourceCode
        .getAllComments()
        .filter(node => licenses.includes(normalizeWhitespace(node.value)))
        .forEach(node => {
          context.report({
            node,
            message: 'This license header is not allowed in this file.',
            fix(fixer) {
              return fixer.remove(node);
            }
          });
        });
    },
  };
}
};
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/
```

```
const babelEslint = require('babel-eslint');

const { assert, normalizeWhitespace, init } = require('../lib');

function isHashbang(text) {
 return text.trim().startsWith('#!') && !text.trim().includes('\n');
}

module.exports = {
 meta: {
   fixable: 'code',
   schema: [{
     type: 'object',
     properties: {
       license: {
         type: 'string',
       },
     },
     additionalProperties: false,
   }]
 },
 create: context => {
   return {
     Program(program) {
       const license = init(context, program, function () {
         const options = context.options[0] || {};
         const license = options.license;

         assert(!!license, '"license" option is required');

         const parsed = babelEslint.parse(license);
         assert(!parsed.body.length, '"license" option must only include a single comment');
         assert(parsed.comments.length === 1, '"license" option must only include a single comment');

         return {
           source: license,
           nodeValue: normalizeWhitespace(parsed.comments[0].value)
         };
       });

       if (!license) {
         return;
       }

       const sourceCode = context.getSourceCode();
       const comment = sourceCode.getAllComments().find(node => (
         normalizeWhitespace(node.value) === license.nodeValue
```

```
      ));

      // no licence comment
      if (!comment) {
        context.report({
          message: 'File must start with a license header',
          loc: {
            start: { line: 1, column: 0 },
            end: { line: 1, column: sourceCode.lines[0].length - 1 }
          },
          fix(fixer) {
            if (isHashbang(sourceCode.lines[0])) {
              return undefined;
            }

            return fixer.replaceTextRange([0, 0], license.source + '\n\n');
          }
        });
        return;
      }

      // ensure there is nothing before the comment
      const sourceBeforeNode = sourceCode.getText().slice(0, sourceCode.getIndexFromLoc(comment.loc.start));
      if (sourceBeforeNode.length && !isHashbang(sourceBeforeNode)) {
        context.report({
          node: comment,
          message: 'License header must be at the very beginning of the file',
          fix(fixer) {
            // replace leading whitespace if possible
            if (sourceBeforeNode.trim() === '') {
              return fixer.replaceTextRange([0, sourceBeforeNode.length], '');
            }

            // inject content at top and remove node from current location
            // if removing whitespace is not possible
            return [
              fixer.remove(comment),
              fixer.replaceTextRange([0, 0], license.source + '\n\n'),
            ];
          }
        });
      }
    },
  };
}
};
# THIS IS AN AUTOGENERATED FILE. DO NOT EDIT THIS FILE DIRECTLY.
# yarn lockfile v1
```

```
dedent@^0.7.0:
 version "0.7.0"
 resolved "https://registry.npmjs.org/dedent/-/dedent-0.7.0.tgz#2495ddbaf6eb874abb0e1be9df22d2e5a544326c"
 integrity sha1-JJXduvbrh0q7Dhvp3yLS5aVEMmw=
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

module.exports = {
 rules: {
   'require-license-header': require('./rules/require_license_header'),
   'disallow-license-headers': require('./rules/disallow_license_headers'),
 },
};
{
 "name": "@kbn/eslint-plugin-license-header",
 "version": "1.0.0",
 "private": true,
 "license": "Apache-2.0",
 "peerDependencies": {
   "eslint": "^5.6.0",
   "babel-eslint": "^9.0.0"
 },
 "dependencies": {
   "dedent": "^0.7.0"
 }
}
NOTE: Watcher must be enabled to view cluster alerts. If you have a Basic
license, Top Cluster Alerts are not displayed.
PNG
```

IHDR.GsRGB@IDATx`VO
{#KPDGmk[kZWu{C{%7||29W?/w9{_\u_|BQ>;T+0F#`hd7$$D%Q*i0F#`IMF#`0Fq0q#`0FLq9
o0F#`S\G;[-0F#`IMF#`0Fq0q#`0FLq9
o0F#`FJzrl<|^N~v IjZFIdfHFJd7t.\b<*)W;"v&Ki=]C3fm9 k]Z\$YRXTK
d9FK]E}0SJjSE
v#`86^q6g~3k,/#p
5M?k=#}U2kk&/M*CyDjk*TiQ"zB9$=`^Sz'|,_IFx?O.zq,U)bvR/I59i/N9uLTdgfAv@iK{R%;F#PB02l
ekJiE=JU z@:U 1edw GJKWITx>UX'79RJ<m[Fx%<sQLGpL$+FAJ%IQu"#H
CAWJso%.3OrX\"bbkHnm'Y{ZF{nyGd2Zch-TA mKx6F&PXrtK*+R+= Mesnl"Np-{e-
1SBBA=%88Hd4LYUJm}CTR)3TQZ=I2
k8ousu%vR#_/wHxfr~rF e|'r`6>Ar}dbBnF=w,)Q--s[oe|lS&O#->g;(:]kPEAA2s*.ddv*y*\Y;D6Q Pl+JJatn)!]cBY3jH
BQL[YZ_$Xzvh#7$[d}kl+wJI-
w7oYEk;y(ktn,YZ#{ITP)m3lw1\1eg7i,,TN/ eU6Z`\>$^~+?V,?&l4JT]R3]Vj3oPE&2~h/4^
;v!e0wgA4,O$]{f,i>rF^Eygf k-g
!C6n1ee#-UkgN/%lyeek\p^vLGoM"F.Wz]]f_'q[57'IqX9wXoJi s])9u*hzLYOGT(aLoO1up$$>%2lWlOQ-
qei\7I|8E(kwIZqU;*
y[U! 6o{2Ytt_9-$&&S~uD?Z 0F#pj?B"9Q(Wcb!F;Zj]*vqb!FLR[$!JFV\,^^mAe<?o\BI-w7JK0}Ve`yxr*),9O;JOyD-
_%#-M4y[CHLL[?\,MD+[>Z([%<o/ =YxifEj. .T
!,R%Y*aY=/1]JC0@Rs$FT,;GZv,~6vSVm&RT$w@KW%o/M]vR%M qOd 6_.)
ON-B6h\D3OINxSf7mS5
7n/)^Z.}}IV%3(D'syru2/KK`h/})!A*&ergEKnQL~{<6/)ysI\a]p ;S~ZY\BJ/|)/?
Lvy2FNCW.>\FIQ0{tyr!n.JPA?'7N(zOvP]U?Uz:9eNftkTz|E
RJCTi[GfGJUHSr$?.\+WDPCo^J~hDuz]YL(^)K>D.~wn-
C:4nbUb\#4GJ>o~2rTiqI>Z4f?X)YK70i4Hhgg=.#dmy$3%MUA| `#`@,.*2~mzK2>:<.xQ\Z*\<<e2oo%
d]]R|rGfGr%OuAq:wH8z]2]tG#!Mf|-w.LR{@%js.2gm.KPQ~4'
r2pF#:bi2RYr^d^F,gY[d%L]
TP^DTM-
&tt:@O)VW]&kX<~|ylZ)PbU(rvxrxe,mK;>l[[7)kh[jZ6=Q~Y])uCYf#/+8V,*jYu"<c*R'~FbwoaV|d!aH:PV;EDNrU
z*m%3o/
=[{z
e/m.orUJ}`ZWvYq{Ter&|sX^*eC36EO_:]mj
HemK32?N-d$<}PZ
VX'/nb`|}LO.WYqf2#c'9?v|\ZJeZgTxYxim&v^!gL,{wQ&Ff1AlUe4)n\p85Y52I(d'i!ooI1CzY
RzJXa<2|z~[+Jfm/zO];6J;W9<}H\ArVgwX(gQ~1 Bjvn#az(q<vy@o?O~}Yy^^0FGv
B#ztS8;k*=C(Tg:Tq,_&Qm\u}W0vgCRW*y:SeGmVWBB?G"n6!T<u
~ieX7ITYn_{7\, :yHtUVT!ro.G35Be7(Ut Epua":Z&v+sGnyE*Q*V.h;/I*
OhZhLg}KTda*'/G6L"]P$t[d_UJ9iFy;J>pF ISM(=CH\ym
UUP<?S\BT+ZW'bfVUZ3 G7?ArSpl20aw8.1JwB:Q
vGGBJU*+!N_jHZ6dT\g Q2EGQvfl_;r|/DP:Gw2PQj(Y*.r<\fyA?|DJK;>$9,Mn(9+SZ8`n~{od73xDAJ
>b2'"#Mg,yUJB}n|.zEx)-dMP%:evjR{^rP'6C=&qKlQ{=$"JNQk { {)|k0F NqJQZ^u;^L0;Vy%J
w
n'?\GHtNVIr{wP9c$UHZ>d`1-pU
Vn;[!Ewm+_~&eWGA!=[nV+SjHKd`"TZ*NR$H%h-7j)-EN9_SY{
eGgKVA,I%+6=GwGOi.1jUsZ\xTD]
LvsnhcUi;k\)eNyu3T?0g9%NmZ+3ul<w9u[(F4>d|\fCo7Uy-VNuS
j(Z}c$a8|"ZCQ3\]PyGDV%z<\dg2w$m[vNh~XXbWvOmc[u2LbIUJUtVm7C$'4R+K
/'.l,O](4WU6K~xthR$[y; D=Lp\ZlZ"/!DZ!I,<\eLktozg+/

#e<U<jCS7Rum
{]WTq~#`CV\0Z%Pk&2K;Sz_su1osOWy\T ,\Me]uF8deO`L BTM_Y s8|<g@"FNu5P/KuR=Jsy$EK56gI5B
J.:O~tL[kR^#=I./SrD^]P+^Ib!~kTdmJD
^{~9HKmIzs{_kCm~enx,_'Swj(M%|.ypjYoucT}BZFvPnp|D4m5EfZU Oom\yE]%R_:<8.+/ O+ygzY[1[bTxI>
[Wnpl|kmdEyAUT^ZJ[eK1{LO=U4x2GY_ Um*9W_R=DD_.h9 wD]TWU
e#`0uDIr8d y`P]Zs
jexG*.LG`NE*`Ws<wWUDd'&cABAUUc6w,GK
;h
IEIp- gPIr_u*_KeS]^U_9C;:]x
wP%u[e5d$uR"?'HQT@g2UePb.6sXZw$jSZ /\nJX|x<4u$^ww8[]5.mtwQrGV
StMQ)[I(G4n!O,N*q(p}:Uz6W.K Zo-q}dHOPCim%]H!2zP;nbrqo8nbS<uS\9l}1wd`o&?4BjLzPJ

*!5eR jX7Y7Dn"GF/T_&_bOr3Ru>*FMxh+Iix;yZvoF;-]"W8K7g(u_T51=m:
jyr)mTA(7G+3tsz#ac?0F%sGB=Df/)7{XyKUx`Phh'oMP6;ox/ya"9I;Xu15\.@VwHi1VZ~/ct{RU|l}xL&^Y}Ui}+=
s>M`LN"D^U9a(<:q!/K%<KNnzd`\DYyo^=416=eBx-cBbT1oyU %:8_gJ!"a=y;<d`]L=/Y-.MN-
TD]p#UQu1&i)Jw_<AQm:^*s>O\.Ov1BwV+
g.%
i3nqgLYE.{.;H_fS cd*5N2<sF<"ys:U`_[z<XKx{uDJ#M]hBU*Pa#\h/3X<bJrAw?O'/^1LrL~wus][KI_}](M*
o}XR,ZGHnVnX;jrd;^mZ&eyf"i/:U%2Vwrf6D?L[,(Ua,PNE%Xt(H[`FKdh<pnT'tSkt.u_N_an]j=hVY?]Kcife}=k/|R
#NFly 'e_F#`'S9k[RSqaOGEJ*@pL:dt'+J54 2]P-XBN]DTaP+*XCu-HB,rrdO
+LQ^EETQub=uPl<yb:;w,~\l}dYnq<LnwJlMXdu<Pu/UruB{Pv%Zr
uP>IgVw`dX-8_k]H#`Dg|WHl @*en;Psz ed)Vt'Nb[Fd"jtBR.E=N@y.NbrfXhU+R-
cn.DzmoE($Inn!=FkNtQvmKXJy/AqAsU7}iL`>0(G$uLxi&G>vmDOWa:Z7(WvgV=DS;anJiH&2
&J:BrPwTjeR5Y!V+Twb1FbU25O(>mZJ|lef}b^-WjbU&zRD@Z#t[31[:EF-KWm?E""uBJHSzRkhvi*{]:FZw-
{OdkH;;y,9ja+rOuiZ HPd(TOn{PLskpzX({e[(GPm)X7%&F]m"'-OHd^FtOk>$\U1F#@BB]Vbu07(%\v/(GRq97zQ
yZQD^"/`~#pRX%=ule dg7<2\v5F#pJq=a5e + 84B?N~pLJS:t|GLo\qj#Pz [,uG U;A5Li0F!hSU})!a:^e .+=%]o#W1D.
`:^E|s;ui?8OfJQp@i(w]L-A#`0
 W*_wyexl{@X^A
gq!^Os/=.Z&F#` :{>@;#`0F4'Tqa]ktM -JCh+0F#`0U8aXE^P-
7TQH;l0F#`@&Pk9vpMtOg{/;fcO4F#`0FKI/CM7Bk,'e//
;9&~4Sr~qq#`0F#`Jd@Vj_oT+m{$G_ Q{C{OJf#{oF#`0FZsSK
#ChTb[z$KCY)2(1lyEOU;bG0F#`@Bn}p#KBTUbD0/\,_lJKM
0F#`0GrU#LD>9%jb0F#`8L:.f.7)XnI#`0F#m'PK^AJ.m[c7~G^jAtK2,#`0F#`j$PS2=DOK3GH0g#`0F#`@uMq]/,/(*
FvSKKTDsyW'7)6o#`0F#`*7%-+G.r2S;<~$Tn;b0F#`@9zQ\2sr)-
T&eJIS{\k}F#`0FT&PKN~X@E.X.|}ieqUKV6Q0F#`%PK~a$RRUE
UY Tw_\<^Z4>l0F#`0{Q`1_KJd3<7@,/w{&`-VE&P:v#`0FEN2KJV%KCb.d.^8v~rXAo#`0F#`m32%
R%r]wc#`0FFJ[\mso{b)":%e|( 1X9#`0F#X|#%C=lIAiAPYrzxY]j#`0Fo1cV\xdn{_A,`t5
Ml~>v#`0F4&xWu!EEM6Zf@v2?V{1e Fv#`0F4Go{NbXn;1MR 0F#`0@JK&;e^%OXQ3F#`0F4NV\db_(-y;9-
tK$fI#`0F#pR.fcHRa{Sw6Q-br0F#`Lv-Yb~_K#V-[L+s#`0FF@Kj(XV|?%Uw[rX|,.OZ0F#`@%Tq}-/xj%Bahtm-
l#`0FKvKH$"<|UhJRcZt0F#`8INqQa_rjJZZr$FX0F#`0FS_#`0FBL
!F#`0FTM#`0F#`S\HCX10F#`).U3F#`0F44b#`0F#P5S\fcg0F#`h Lqi
a0F#`0FjT#`0F#@@a0F#`@B>u
!RX\rVm0F#` SRRoS\JK%22R%0F#`0
@HHHRRR~bhh Ta;`0F#`@$XE |a(JU*jyc#`0F@u-
?9Ji0F#`5S\u[0F#`AF#`0FQ0Q7U#`0FLq99Ji0F#`5S\u[0F#`AF#`0FQd$(4V)c|oeF#`0F[D~4^XX$c[EFGIXHdd]R
RJL[*F#`0F
:On9{thR.?<tj<i2o6&a#`0F#`Vq).Hy<d_T$[]ddn2I1cv;_|WT]Lyni0F#`@eu{Xs]]uMEEfKN?F3dZ`lKF#F#`0F]tM
N$(-} S}G +9T/M/^.mZ#%`0F#`8LnUT:m-^xk3Q{UjDZFe

WN>57?.~]Gz~#`0F#`jm;lyj,e[qvr2ap/t7
y+sFR3X|>0F#`hdnDjiQdL2;OPZDD^{E/uK6.#`0F#`@9:S\:D{InIHRxB+F
dO-['r`;ef[_^@2F#`0FRB^(IP%E3
j^I]P,]_=,y1F#`0Fxl;L/N1/TfDC[|$edqxNZG6H"9O.Yy:0F#`0G3"UF#qmTyQe#Q,XLWK})4uE-
,{tdF#`0FrucqQ=c]b
gM' Y9y2l:l&zN/1L>kWk6KI0k #`0F#`@Q\BBe~3g~XJZv<ZbNii
s\&]b[{(jm)[+SQBa0F#`@'Pg>Y%}[^ dn\ZPX6]D 5B6MwJ`0F#` TU"<dsu`y0E
uKtDs7^7B:mOg?]f/m0F#`@'Pw(u
_n\K3j)QD_ILMg>%OO%,c#`0F#GnWdr#/EC>$ZMnw;
C_X}1iO#`0F#`W\HYa_!/^-u{fMt]~.q)dJEB5{v`0F#`@i
XPp+zT`0F#`
S\\,[\m0F#`@#"`E#jl0F#`8Y r#`0F4"4#`0F@q9YX0AY
R$!!!JQQQbrV*H) Z>abi@%`Km+o8\?@u;?P

$;;["""<X_|SeYRR"~o9/999
;*S=P}}>EAM-Lg222-[kk1@0~ZF^ 0z5kVl:'BF^^W>FxlV'5B~c1d5.}nU~_;kK>}[aaO~O8?#2x
r={H=rvx_|!O>'|vi?S4sd(3ZkyY^s5rw><BF +/~!^z7k],7xC{9O'#FO?
Q@[l?\[Uh:S'=z:B0e1cgmAyy=ax*]0'c=[7o)Yz&eGhW{}rWTLyU%I
_+o9~?)N\yC*M~#7F\P.p8s>@nf:,O$H3p&&&K/$w]tE55YN.J/Sw)).-8
Bys=~12vW[NvvYi3,=lQ.]TiAw8~Z@m b%e7$/bm3s!3\c*PM@KdQ@~il.fHdq
cE9sk.p*,><bB@BBQ
2/8qW>Iy
'),&Nh'~z}~HJJ;`(cccgg>_e@aU3Fpd7[egF@)/ }J
pj5`u(k8IsB}<71VNc3P^7[d ((0' !3rH<;7ry]vcW',nA#1/{.qxK*kHY9Z0hhY\#ptZW
j
"1A>A]<_=u5j@VJMMt>,
k?pap7,6U 1 3P8G
L|H:(0[OW6r"8&O,K,{]XZ@,?e:.pq;P O(%@tMV]-/i68u.9,-
|GSNxpG#8\^$M++i#_n<5/EqMuri.EU}kk`6`U+/K.iQe?(/@i1:u9oW`'1}q:7>,[/54TZB+8I
)sg|%E:h(*r3uI.pzOd`Z].Bo7xqmE;94+Vxluko'f]',
G`#OG$D767*U#9sggq7+oL6;dU2PVSwt-NJwPFP^rJ1]`9h .$j*IHH8bbbd "iqy._u>:v#rt+c
+@IDAT6U;r"R!CH}=E,Xfv!OcN:ywGzY<}v3glMJx/ebW_}%i=;W ._tH{iQ6#]|2"t;G\6rnsyp7y eUrQ~S-
==ey<,wE6lw-I&>vWEZ>PQ
/J w^R%eMF
}FA0A5D <$TZL~={`\
:f@IaZ984pB yPZpu<vAu30#>30@\p.G]wu3.SRRA|x)#_x@FY_r-x[gan+|#Jz >.~3&.vAmW(O[Xp7r'|z
w@EchK96`e-y'%~r
O>x:(X$W3[Ji !];||:_i/D}\6YDpMC0o \ho\{M<KyRo-q)wEHKOCcNA;~?p?6c"=;vc'0G}Sr8OG1q^%
#mEbxUhQ|apnv/}J~O]Hcyq
e_W^bc?g)uhB4@
d0!:KOUK,F*yaYgSS`!m.{uo 3`@d [oJ^z-"Ao?{d1P.%>
.7qo%k"|Aosf[7&}>[lG=Q|"O 7X7xRV(&?<8.+pB/Nagrioqhkkwi\Harae!I9\-',3'_{w/t8o}-y|OQ!H=5L&\:]w m5X=ro
.jeB1>O|~k7|ci>(sC:m26aR>i#}{oiS6C&DhwxQ'kHU>;I=OHM:Do.8hs\M(=%et'~s_Wh
V6#ps.$aG
U)AzKh3@r;\^yoe5-<7;` ZfN}3K
Vf 0^f!d~3p#K>05
s`N!

Bl}{aao^Eb.D`
oZN`
{7^Mq-!a^~^ifq-wQ5PWfcQBiw #\M09goZj\D)'J("C_Dt=PPli<p
1G8R5\OxUg ;,*%xNH%q'19v_Y7;R];A4rMK0dBr]S
4Hs/3
xPjws5#q/Q/Awek/L<(?e&\p&CA#p-n|^-}xF\ve^+k&X\/x|MyhgVU,Cpr,9^;n#`Kut8FE:]%Zob(-SG_~qE
$Y:""(3k`f4?#ObE1`#T1]^VB5-\+z[4H{,SVf).RuYx!h!2 p2L-uN8D9@A0$}"CC {&MA

U)):C`=B6S.5f_tGc)-W0H,y ._r2SbWNve('8"r=mM/Ns/Bn6A7<f9Fr},>1
9'oM(?{4ps|(w(\z_}>J!?"/e
16Po>Y#}1Gq\KYq)*|QY#tyX9|PJ.=z'XCaY#PnlL8O?
..V@ :`":S\u$$h
nA_Tb7ad'VRRXFu,=Z6 XNiG$1?m]3 %T5`0BpBxg-=#k*=00;M.W_]fF/poAO?7KF},AA7,``nJR*`Q.>Nh
]~#Haa"`Ga).N<y?od:ip=@bf~C .(<y[\7~;O?r"(|Qv!{C%co,,b}&XPm>J}y5 Ms-
,:sPZ?c/`!9"|p!P/x!rr#v37e;@HpV(X(.vv0F^/"z,%X^GGd8 (zQZ\MQR".zXphy
6?A6?3J>xY^\IV-zlF#p#p<VirUrRZ#XZWR{C5~3h{
.72<IgA
N8d8xVIf?4+PI9C@MAA?P<|#0pu\Yj"sMZvPB!}G v.<;_0G9FGL\C[!H
~6qcio/Lp t
-/e '('V fyKoc~Z}oxf]Uu8i:3R-\"^bB{c,,]^
Hu{,q'@ix+ cKcYGu'M)|to]`BoF<mi&ZkXC\Z
k,oP][Dr}OKZ#,S&^,jiZ75&a5*!5f,2xgs_B3|n)B%;x)3U#kD]&3eOfd`o?J8!UZKDT'pwfRC^Ghw#'@r=-@h~|cOm
K,N`'=pruwE0D)uyCqqor"xbCUh^"}rPM0XPg2-B>W
D${kbbIB]o\&o>o<k
K&u,a#ushFq-H--u'v4TVDNE>A8P70#F}cMH6G/9vq
~_;P`bz'7\
Nb#p=BC 4qfYqq7M]#b+*.Fhu"23!Iaq5Op$}@YhO,1
}uX{RUN)/64r9%VYrO]?
+6g?>Kp:M"URr;sU\@%f7d=/",85
[2w:[10@~@yX+,u+,yN<`(.4Qy'q3D|I[*$~j&ic:{6:p5_$rsO+7~*-Oik7]M]X X&f \ buAw
]SW(N89BP5K.
D`g*f sqpx|X:2Mz,h.IZp=}hSviOxUzgW7vkc8LWe6&@u}IY:z?;lJ^=K96
aP@PRXc$y`muRF!WStx_vov)'q3R5>AG5k@}3SClL._/8fo#B:GRXQ2T& Jt;}Kq%.k$}4U[Wu<MfYDm
q'e!
w.v'B=5[b/X0B:u5c.`,f&L Ff{;FYbk\7SJ,])#nBj[!J(\\]9qI7(;()3eUupIc'+,{+phQ> Nq=k<?RUU7/XC~;vm);9Ko[
Da{^C`be01wQ^^#>G?;=pY]g^oJu-?RujNI{k`c
r0|~3A?.3p]y4Y&&!|3xP~xa(K`G*(&YBeYBqa'%\jh7=@7nH T+'5X#X~z5A>2xX
.Q,n}LmKoTdI~DY]__;QY.^(JL\P@X?D<Nx&p<O
@1fXeYE]q%Uauakihgp?6}=NKeneFD_tl8

,*(>.W]/uf(-W^0X~N O(n]2b]a-u'P',)1@@r`KM
tCV!?B<)951I7kxh@N[&w7.S4q~>?}yedKVe=e6<VW[}(-u!"an8}+@8O:Xj`@{aWn-e*CXU}7z-5m
en+)fS]_phMqiEe=\G@>vcH\ZR+*Z.;PZyCT+x+\67@y!FpnCe[MCzjr[u5r R/>f H"P?'D QWL1~/cc
hXR4!"_&PZq
vCp^R$_Vy<PQ)'#o9i+mDiSiS~G:Yf}rpl)7\9eCF\{+q7pvo_}|G/\9IzrgJ$C"s.?BergJsK.Oh'\DN_&H\po#MT&)F$e
@HR?w4W+Ls--v^yOpcaf',2\pZ\[J<fXqQxnv<b%T4Ae9kBn>Kf@?qAb>BLm_oe&3
`uE'moP68FYh'/^!4X#sXk+;.+;TpUsAaWXA(SO=AoRNjAGe6E#b c%}K'2PFSvB\#,SJI`y'tu?}
w,,|XX^{s4{$zQ_9Ow>}0+?mOz&w\~o+(Lp13-bWC\i'Of{>!!a":ge+u&.t2#P P[#,)Q*DIC<.Y@-K8p

NfF(bcG=e*E,EgevW7 wk>BB<euuq?a[| ,E*%kY*6m\{Br!;!h38#o{# VtL(RI{<
WNFsY?yoV@pL4Bc=u %aIn1w_&\FoOt3aA2@:VWuVWQz_Y4O/AY`RB:>'`EX~ @nX+_(C({(J!$,QTNbB
1@_/(m"v& (/Xs`G@qn{.o#PMq UUvpg|-]+0m^(2.F@A(fpgPFbf
A%`vbF)b]<*:Uk;%S|A8u>8xL'D9f!YO,5
[ff8G T'oF\@#OI4]YHv&-4+070DM.uqH?D:wyg?A8s]vH?O@g7uiy'
ooFesmE_b]wQ]YO2qO@'B
Z*TiommZ
Z-NpwH$c9ox$$! sa23]7s9{SqYMm28q+9*9s=x%i6QRW.,s;!
{XH6rY<qgOddSV]3|5=|
xBPSi%9?|^j./0GUab8<atE Yb*wF_nI#R#hq6:~==(FO_8.'+#u.P B@>If9+f6K,QVIy
<M~Z3`QySY[l+$+<0/Z7OjYuv*/u~V:s$2 (@Mf?6]\zx'l3uX'+e`Wyq_W99~0+l\:pCWJ=Ovy#<A- 4
jtZq|~/Lsa^gMO93O\e?zDa8=s-]g&VuR[rT'K/`?XeM3Rxfopyx6=(BuxZ;S(:mpdm9.I5oX&^_58?}\p^ya}]XJ3?
:p[/V7f8-|X4j|gw;#s~3,o~2Gy-}oq< Bn"GLo_@VxH+;AO;Nxqg?`(m<Mui!W#EX [N5$Z&3<gs%O]=-/?pz,
Is9\>y835s^<['l^Su|un3^f^nU&zuv3C8?p8-soO-K5zDef?\.yq:p[g<dnj=zp~|}
SV=8aG5afXCqj|]q[dMker5.j;qo/n|p;;m{?s<z,-|}p|5[2Tdm04lP^=weWkrXiRBQW2.=}YmWwfqzn;
+\??mj}e^*["-
O|6\9'*3HL^JY(j$CR:TWN,_xAPTZ[V:W_n!^[y1]m6u8?V=WWPK9:enj1QPWtw<{wqejp>j=K;aGm[eiJqnnT>A
}Ej9j\50#qQn*u/p]`>,]m5/ac|M'
.$Y5U^uK:!p@Vx{jWK]@|^v]sf)LRfUX2~tgeWfu2U>esc)2qL[lqy[mCO/Bn%`V'.$W(K;g_j>N!
B@{mgyfrrk]\dT=I)B@! =F6AcEZ,&U$B@! B%`3E1],B@! Boq{Hj! B@! @JB@! B@.")! B@!PmDp6B@! "
HJ>B@! B@T\
d,B@! b+B@! F@jC+! B@!`+"#B@! @JB@! B@.")! B@!PmDp6B@! " HJ>B@! B@T\
d,B@! b+B@! F@jC+! B@!`+"#B@! @3o?#! z4_[\%eH(\F@prtVLqEGJw h_dr3GPkW1g
]fZ3xr=LW$&#2ni$]A$ETCdFz:F-/8`_Wdql9}:4S
,'$e"neTkp
V$'=o9?0
m=qQ$]+"qsl,YZ4
b4<$s9h:-IPx'/%C`; H!D^xw`<'.7hU6U`9 M5N:303O
Mh3O{<arU&@Q8SYN=i55\H1Ns2swk7 e,Y'Inw
vh:~nQiMu2Z,3"%6>
rX'VQ{X-7uETSUU=]`(8FKQ|B@VZ~M,h4bL"6KqI(a-oi_y37&b<&m;}~=:O;rRnVl][Le$
z22^Ke{NL\OY
:`33P2tF=Ecz@itv3at;6*fzw S2:t=Iq9)bS#NsV3YQz;63PH>+%X(GU;P1Ty}%x$F&{JJb,nY*"G-
$+;hE5S{%ri^j;E[4l^hAj;o oODr]<[$1Dk@&FC!u]<IliQVuB@ .z8gW?$m7w#-#{We&dQgec|9m7,
?).2<o`f< [bHAW2vv%Ac@f`h:]#p5Zl_<<!rs(l[W }btS3tYe3Nmc"aW2o :5MqdV$90GOCaP~p"y'vc=:Ikl
Xh^ik*d6PJGql;!MC.a0=zZw-_WYkg]Ohpm'^.ZP7+0j9CuGu:
F%ZbOi\c63a&\Mk~Z7'aC{q&M5eyuI0*"pZ7.=0WW8AkW !F #q*N5kwnVCiqbJ{unM\a`:{&&*:
0$azH{?j
/B@"?mh4)zZ]|~2hras&4 G74~F|8/]SV:},mj$! kbMEpth\v|8@?;gbBh7he@x8bIa(Y~Nht\nOyL3a``wX,H@S
xdG}Q!?bI[`Kcoq
s8LBu;qt,\~ZOtEhH8`oq6.=(>~W`OW@+LM{WhrYN($lL8qYdjq%.1|;@cfcJ[bX\\qx.=wUI$xF+iW1{7\@ONEHl
ceg7ll:rhQea:fvgQR8|7q!,X'%@[v5c$]rAh1d9?`te`C:geB@E6=iS{(e|q7~l&g?= A
1XCLR:a ,'wnLSe5Z ! n198\=izw<571fc"<9+\f-w s[iMj('ZE&=idIKI8u"1W}CMfr7}
^GpNuSDzhH=)0D!ot$p~huxfeGH<n h
o@LZ.|/G!I$_:O*jSg3nt%Iu~
'Q
vBf%Xv#a8&=k@9R9Xs~G0{[8k&siG9;HPB8!#|Q`3qJ"&?zPw`h`t3#]e`/bO#~
3ZM:7L=' tmG*WN%}pZLC3D2w [Cv//! &`ll{F'R2f"VIOD+NAuq9)

.EaxjX/WoG>u#;CB@Tl}[x+~
<sZ(((cEAVP\kPchhVkSRuF$(ihHgW4Kv[])N{zxY>Oh#>/>]B!87ITtjHZJwL%R(5#"2
:\Iax.]_FM4#aU
\VM]i! m.
7V@ngCNtD4t48"K'[y)#Kc'i)
+``6(^&rt^3y mR|v@CF^cI-^NhV
/hn\iOGe88y]f8*r
H,uh
l-:V&%6Qapd:>ND]:"^~QUB@..T>$pQ n,s/GIsxjhO|I~-{bNZB~8M >w/3drS'l!?A4i7:
MVERjy=famhO[+U2m*3PE\B;?aMHB:L+4-VlyHr2+#tD<8,Dk=]J[DCt-
~qmxBY{jxf=Zep|Mh~(N|*:3Aq{Gx:+JlMZ<W@."s{{8K
Kul4<]6n~pI$T
UM[E+Gx;=_Wgr7$tG9Z[w.<|k)Z#rr,|m'{TB@T'rF[/("!d[iBNZ=GqOlj[B@4?u{<}#HM#1qDlK|nQhRvb!B
j,{5pr Dj9 h5VN;Qcv?i$2)%)DO>b3W\1r$u=ozhl]qS$\4o>h84|4i.YNu3T':7v)3e{:nZ-
aMEy=~8[&J,I@PrnT=BA^7Oc?<P
;YTE$tw#CI092Y>;l$_cUhb@MQv9c?KjB@T0$(r5#iKXkr?XZs[v?>~h$V<V/|nTyP^+hI&B@j$`Zot{7a#;ja>+w
hibvOBRr-9/zvYv#JLyUYP[xeah9B4Ey!"*q1QO;;EZ ^9c]>'{$9Z^^6\F]`r
s/(zJ/*fy4okJAyvB4k&mD@IDATK"p>Wy)Rm?oA.a?$pl{ZO8K.rc<-
1WP{g;q
WcjG}`"A8>,pGLG"m6|H"!}"iCM{Cbk9TO63O;s~mp t9"-8t6R
_"lqIJT[-*igx|",|ax(&XFm29jNc#Kibh!-4fs0GrEFJJ6-;9#?O&q'2cv|~ 9_*P8?8QQwH,:4"M1Xr
v3#OS4n&1er.VhRB;Z6T'G~&1vsrP42f<ihQ;L}p<B#}=G#}O>)~^ YMIiZdFPIs c-
gIsQ:cP^{X#d$A2LKcF(C`0vjM"#j7i6B#:o3Y[`Xvs I+0`e'J"V5]oTJ5@8yh:
!zZa:sX4XiA
o#[Wazr(xv6}jqTx][M}~:|h848$ff,"vL|TgLY#%bRc;}je-
|:e4^;'"b_#u}(Qe?wFV^V6B@TVsMy^frO5k)>da'3=>Lz0z(.CG7Z)g(pvu'24sF*ALgP$zz*fA9UnIu4i7|A@mJ+4'
4@Oox"oNEYdL+<=g6zs{1w*3%.WLv|Lp:F9F8%p|jL\iR -q`(}~.Rt|xiITfq`dEx;Q{ksp*R>]Tv'\fDNkb1Q"<#z/"Vf
E B+lVO0S')4Ar{-zxfa>S]L,>/B6xGx\t j#xQ8[|Y _i[vi\l9yt
@5&P`&ds_nD?B@?/hxsl[<4Q4jJd&>5eia39|7Yx6OeA52b9r
R^Myx0CR*N"x[><UJ*p:~jJt4#J1r7"A>ii/qNy'EL=Me`1/()X%p{
nKmq;r/A))0u!,m3N&4..$rPSX
u!A!p Fp!)36cd_4Gb0WdT<=@+O(iVh7Xzl$Y8! L'yW&%~:&:D e^!jeEeRRJ-
4y(8Wt)WV\5;Vet-qo,-HsGY0RJ@(ZyB@)\AhV'3?Qs\zl<0woj\h[rB =_CabwO4L!(Ev%o/?31O0: .7ea7b"8U!
@h5Vm;wVRa"9)$`;Ofa'#b19Bi`UVWl-$&FB@!pQ B@{mElj{uFo\<Ygl ,=pX,8liB@! U#`{A$6EeKHGNjrC
$(:B@! >hd,z|ly?B@! B@@ .eS%A! B@!P_hD! B@;K@;[JB@! B
Dp4I"B@! % - ! B@!P"T$B@! BB@! \M! B@!pg rgyKiB@! B@T.U&IB@! Dp4! B@! @@*@$B@!
B@Y"YRB@! U KI! B@! ,\,o)M! B@*
$B@! w.w&B@! @RhD! B@;K@[R
vKB@! Bl'|0,
&rtvDM7pDf ?Og"Gf-3W! B@!CFhQZddAzu@B TN_CMOWcGQP! B@! BZ?&l NMdfQ ,gEo.B@! BBfAF*%K-
|ujeLB@! BBfzY\c>|>r9
/9y%e#'|_T-LE! B@*uUY{9rBx+X~z=:5cjnrs)$1+E! B@<.[XdJ:#v5 nPr h-Ph~7?v&s"2B@! {l#MhP%5(*i *)YZ
rH! B@:.qIa9gs(KYf1{X3B@! B@6\-2xf|aFow&CSAtS(,}R! B@!%`~"SnMxvKAzn>~I_|bIy#A!
B@!PmzXdRZ}Xb.%^}(%hPo
+Z%+q\! B@! l'%pGG~xpm98}QbF>1HQ+7u[!L&j]]! lCZl{8! -i#eFu
.\K^m"0_&D/oYf6;OAYrFdeexNB@p,xzzvWSAASB@*~ska7<5=<
IpqSfz.,w&=?A[jpvvF```[+B@i4VBKzz:AX B@!`;AyW-g.~d6my{y
Yy.,e!"NNN]]V3B q<q`-/%Lv$uoV! .Sqqqm^JCzV.Mh^-[cxtAnB@TDL|K[M_UE !

pzAzn?,XhQ_]}^Wun 5B@j-J^[%! B@!P55.B@!p\u@~:JB@ TG+KB@12]! JWB@! w!\"UB@! B$\JoB@!
B@Dp/TI! B@Dp)C ! B@!p.(R%! B@! J$&B@!]H@HB@! (I@<B@! w!\"UB@! B$\JoB@! B@waJBz
MB{8;;A+{18jJH"! &4tNNnO7wggWnV|uf &6Y|,3s8[CD<6i,z_<eAF7}GA^zM\M!P@vjN;p9$&r}78t .^Xv3
8w;g."PvJ5f]sBalR%SwE?g<fBO
@y1K "O\{f,x!t
o/UfW;yv^MJ4xV)x\O;*v&2 LUr
3g}|5vKk.B0b70t(5CF=0p-x+#5q iT_(@dT lPP?}W~1bFaiy;1~`20t}0T*
Ic*RUN/[/~u6u8sRS*< .fb*zIw9.99zQ}c{{-4)xx&x9r|6;`jp/<7?zw! V/?zC0g$8ZQ1Mde<[dWfzjs]W<xPA};ao
>[E?<8fxnh.^g^~U5>ZK&*we6~La03xu[G[R:haWd2`Y{)Ec4tQ^_r*#;7>p DdfI#D,Oo?ie)& ?_G
\=|PDektptrsq<q
p ;QSN ?'#W <=X@&I)0Q=]<PTfgf!d*TX ?yp'6'5Y|<J]W'E K1s5j w@jRrziQVxZwSfdW-|HLH@nNnS/$v8SE"
BcAanI}n`DKHC
gKwv8C$H+knv2o3BuBZiW&#O59!&>&;q5v!<Hs'G{GxzWr-:87T'|'[
CmrsJ=#Q'6/-X<v5z7Of2KIH1n>T"{IKN!2r4gQ{F7=hp(Knn

H_R4,Y:4@Ec^t]:SKXmi
ZGW8PAj]{w5vUOFwbUJ7nJ{c i@ux A^W<4]D$+Fg^rRm0e8vAn?6a=4#NNh-$ZTX:=a={ ,,y
D)gO@JGAKfcIGp9)KH(]-
QE3eK<P]~}1L8uY_"mE_td]`bS1}n:~|u=zR{x/d&7gHq4t8`Ud>Pt17k1bm/O(yvtCH^xhtdqh?=wG;,.D'B@TL9<;V
2^Gzp:L:M<|fUg_O[h<v^)]13qQ<2aO\F}]+?p?_^Tnr>v_MzcEBGhjhI!"*q1 9~"p[cPhM~rBxxP{cV"J
G/+Bo.TLLIxn0/PthE1cr1dDb H!S=g\<}@m{qh>x)/?y#AJc1qo4E5V<Z70=+9W?)Z
)B?U}06ny!2G,x>?^|ql _3QOZr.&7EsriObu_o^4,:n1'NBC~Q>&'&9!]CcX8L&-@,[+ZsA0k~l[?Xt:h|x5
xt?K7_8EG}gIp5+dnguH:%50Z7x*V:}Nj.dJ* g5p mG{m~Zyb!R{I|Zu)#Px]OLQniIWOFZ97uj2+-
4'&<)F^{!?,[6nv!@9/q'of~!5#B[?gQ` a,tjhgS9W-C}4fu[oV0g-!"FC:AmxL}yw30uT?~Bl~.v,
8s1jk&R\2KAKli U{9uN=QN]0)G?SM-/-iSba>t5=h.9${57szk
vW,UyDVlc08cX7hRFz$"_XhwGpa%:+vo7MCM{{;E@f&rI[;{W3:{`O`ehWSGnVY@z=nUR&!LcFG<`w1`[q`?0
ub,\Gs~n-`"go-LBqcSp4o
g<.-02O9e7h@0b':
ag`E`xVZ>>M3GYA9v\MB@ y 'gr7?/<*};6|p>|nj 5d"W:u}uQbBQ:}EU@5N-u&8+S'b
prJ,[>BEkOxX7w>5 |p*?z9i(ga/w'*URI$Ca? c6w! n>X34!c;pvxx-.M_jR1f5g1n9~:EdJX]fyb`e#{#s(7tM_FL:kC
[~^vB5id>ki/ ecR |C8w!=<l"GLb
mo_{/q`.GGN|;1vk.~x@2QJwqMXeLt~\-'O*qb?q%!!-vc!0gfcL}0ctEYD'%4Zcyhdx'O|Bz_4-fLqch?!]7hn\:^
6/}[ER+LBOBS [.
#"ND(l\z-|f,)A 75v(\y
u)\' o}n?L|Lwm:_'
Cn&B4m@={AfzLc#+BG&?<}.oTqk.|Zn6<ty7Zi[5-LF=WZLZ?O
bbN65"1;wxj:Vlk]r^Ws &-G>G\$l{K(|2UhckJBr~}Kt"FNz;.[
lO' "27E|r8_
O {@`E
|cr+Gn(rhk5Wa&$\7%#iAa]'z}iIM~'i3Td(0cmEh,=0e#hZeHhJJ+V^u(B{v@^J$n7-
}jaJdN_l$wVL;70Q3z9!>.!L~2n+Mz=K@8p\-a5iz5p1~Ksr$m_Yc|=x4:p%@Jf]/PdYz |{0LH$'*m9voW6;
o?-SG&m?>4C`U~zk5JjJUiW-nr*#
G@^#gGmo1^[GO[h}"iosv8vL83}zBSq)Y5C[&g.F>!PE
?CXFoY__'ugK3mCry#S6snKu]u.Y\&eCsrSj&{W91j"7<t-]A~mwL./FF8:9/?qV:g'j>lY1M
S=H`L 0uL4Z8g -;*Zwa}8O$
cdNCuK,YhamK&,[a5GXhQXvLi73ih20<?TQQPLd2?G
R)lD$&tK0H+Md\@k^WzGkQT ^gAc&a45n{6.+o ~AkkI:UfmV]K/KG"SlPdYMs~b+LCE-u+.V
-

J.=J87M2%9\>=EDiR;YR^>Xz6,[\#a_7o>f^E9,p:Y#Zub@1ws0O7E2m^6+o<Jw'd?Ww}zxb:{7GmH.]n0P'0P9XB"
E,x[$e249/>qEp1/=e:bhn;$7_[:x{%$8ca>h^6N.ULO87h$;,fb}@ nL&h_G7V&5j{ad %\?z)7j0l^I'h!2 'DZk
Kt }:Q)H*bS"o2b~CZ>>9MssQHE~p7)z}=&qa$&s0t]RS*_xp]RhJrH:l<mii^';3B$R/6:yijGtSb9ct?K[Zp6I*y
S){x+Y1?Q<.aXLFmgISxL9J! n}'MDirZxnOg#Ji(3SJbZ>E
0"zve*-{4&G&{K}``Q<MZgwmFbqG_1E8L@[=~3dBB?V~ZIg1t}K9jz62;>YMk"E
Z8,Qd/T]yU3_{ar'taL1__5.Yvo5esOWmQ8~5^z$8wGn@.Pdc;Wf"Q(|+z0[Pihbrf/~X8E3Dq%~Ph?tehg>AT\zflm4
ZKYyAQ1/#p-134o
WD:Rg9|QdPQ:cu-J(7Z5l#NB&r<}5~y542c";
i5+V"BNm^%zKQa6SGvG49BduS4g}h@/__H2`+q[
vfmHHl{luA#\R#m]"\.*'SEi @% %SwI\,l=/._79HRO&yl>U0"*+4z|~>kvJ$Ur2DC/3&Cn@Qhx%bSBr<4B>8N8By
3iwhSXF_E/W^vpg}1vQ42Y8Dw>|~Z,,eG)4l+eTGqMAkgiU(H%"ZwG)i/SIpn lb&cpK)<=mH@+o~\Yq}\8{@#{!
L\0qxrh6zc_cg9uU'/#UeV0d4

&j|}qL8M~7K>GI[bgq%_Vs?zhzx'qo?m[;@\R`r?KtCGuo'c`xq`0=`)e3M12kVh0cP!a`r-l
v/: ;y/ahX|6 Ql^M{
;i.jZp+kho_{<Z^8{}7OIs/cJ7v8sm"! H@?]B^Jv?4h2j4C=&U^!$l"_]v@=8Il
w
j_C5wMkH#XYy.~k%hQ 4(p/L
? jvBFlI1j$k2O/g.e2V#)5=KWawQoCqwww)-ZCqww#gv.Ke'3s%vk]?{sMJQ
x\?-[zB3Fj'ePR<rCV~}YGZxts(HcX\I}j~)KPE'0o`}B=gwiGhuH@L171"CQg1"c#(*G_i V<)6MN)(
CNi1{9j,zwl+8!xM\(LqeEWx-^y@#f]p+-v@&F3/lN^OTjKfJSQRYfDvSyylJi5`Hx<{X'7h#l
I
o~
=M{6yCW8eG^x>cw04wD.C1HovCeae/7zw3fMBhX4<_,S75]'B;Y%'{F{i*g, J74j4WC6-y:!{Etj
4w.(CJy4]yO^qh/~d>c'b0nJtgA`N6x!'Qik4S:oshM"np[[oR3HP`
1FL<%,]y5!7^nM(3y@xli4 Dy41p$SC'"S]`<};ZNdR0y$Esfv_I!ar:Y) 20 (OA1foz(]
">? !kA|=Y?omydyy?ptf)1/#mdIg0[<xc!xs8UAY{c1c&!g9t/\ ?|?r0#y13Zj8|ypH.CG\crh"iIS)VA]$KN]pudqq9q+
J~nB{tFE<=SGwh*YIA`h'Hq[a{d/TkjoNMyvTR
*68JU?!jy;;jRVy{,j,&\ sj49L8V96MdHNh;[H'amb|F0,eXD$x%]2),()%b#yXS2%_c uvu3 \v6io$K?#hok#UI]
"%=y<'=HM-x E
$C^,@"&<Z2fK@'X?q%8,p5216,I37Y3S7A-
"FLwh$HsfLRta'h2*=.s%Km[R!HMQBIvMQaFXYP%HD`P$m`*.D(`SF|sh zh@ITl`NzN6vTZ]
K]3UW[{Es3GKkVJFG?G}6>M$Shfms783ZjA.FwviQBEcciBv,l_!ms3bNs_V6!iL"E3$&? H2%
Sg!p)&JD{Kso=`MEP@J9dDN 6Fa.=V~002TR*~[S/Db1@IDATY&I0B6<-^5$8$mtJf%R/VC054
Ki8{JV}%!I9?_Q`&$0ft|*q'$|D$s(/IVv_"%) `bjN39IIuE!J:8,R nvMYWJIj}2Jr"B&+4p_Rt*BO]8Sb#+
u
lt,zh3-=$RKYvI{7 NB{tbud(?H/GK9yEBFIc(P$^R?R:{{UOC(P @@ XrW/g>m>9a(A0.,dF^8fg"]+%;/
BGzaO1*ZRnaN)RLQF>m>)UI
(P @C W/h2kJc}=IiII0%~aZEuUo>0!9 2fI(P @@ Lm
y2gUJ~|]ES:aCzTC8xsJ%@&_KV(P @@?=I/J$A3.Iu6=[13Z870.H@@
(P @ge\Xr9[<~Gs1+P @@
(Co^rMY,dSf_wJm
(P @@
}!=HlOzm$
(P @@
B C76tvz(P @@
(Ac\uR:@@
(P @@?q @@

(P @@C@a\TJF
(P @@
)(?y]
(P @@

R2*P @@
(P OA@a\! bxvswbov'DGF *:_ZlT"DDD4Ymj\?/A5HS-
P'S='FFXX?6_:"qRH><{aOChHkMsCh?'n=%5=bxm8TrY#-
h\;/?|9_.X89*/[}SmUE{p\tY\wp~h\u0cLl8x{Y]Bb}cur;smI\"o\&c^>D.P~@m\\ `~plcX8ezj>^BXwFqx5B#u:,gJ
B?]`,Cl<?EPXa
xxTxt$v@
aA2(1 ,)'"!.vy`'qpx#+^_DHj$/Ea/n<>eN66csDREW#bBCQPs$Czb4TU,/,Ax?8/x;=35/+_NETsqPj% 44&07CsX-
,U`o=}JT&&0wx28a4A"ix!,n0,4041

9\V"^T``o}H&8##os1eJ)7Hc=ILNu${FtHX8L`bz~IP|-HaBL5sdD(`ec*pH o
"Kw>
P9-aa.y
L=G47kK-"yY#"<L%'M &[4J,Ja4 3T`[sDa4xq|q1>j=Mc
$1`

b\>R1&VVLSFNRtZXnW~%"&*:. w QhpuE>}& Y%Hb|,HZX8|1\oD&vuR{OcS0N-Fw,E
oh"BCBLafK,*&= 8"F82WGDp\IdXv/*hK3J4 &aU^~5$[QJ\'M\H/[whUK="2
F&(^0UDl]K^\q?K^01OZG 4
0D,[^qC

`8R0hrOKI10'`|3=@i)seKDDQ.<:S~_X|b|B#/XCMI?RE#Ok2\DGro1jlj*"mVL7Gb/
-=[k^Vc L7f
f1<zKw`U>1_byq*,C,#N
Yw@S@+XOx}"c0%eJ{!H?mB %CLqr:t!(_8TKm
!E1` (ocEMQT)^
ocAWA>2}r&?< 'q']bl0@~:O$Xw;&M gi#!?d.]GdZ#a`b&CR9sfNs[qtMI?)>of- 60"ObIUA2DzJ)Ck<a2s
X3q"Wu+.>-DQ<Xz=^4D&1KsfS-K129
1&ss4^'jjMo_`BS,\^MMFZ6&DbxKG\P&PT=Qh9Yk{9kO!gv':rpnzxh_p'H|VGL2_?=x`4(S&<w5A>2R%c(ZUq/L
vU=\;y0~rOZoIy>b}2$J2}A>_&N;p?L?*cd{z3BSUGbi2 $Da1kLu
Fbv5c1rGoL@4^LS
&v]{IMwMgwKc~l
6@r=/nE(L]7o\C#Iqu6AR}{vXzvx2m/C*:gjQr.De
.C]H\nbdmLO_JB"aI!?qE$ 8=C:6+1n<csQ:JL9tQ/7A0b@ry|.G
MQeghYJ*];/!+w5 .3Rn,-j#s,^kW5h3E,c3
{';^\[P^
X} qe&Vp"8Mmi:a
K.2J=! {gl2D +Xoyr/HDE0iT|q>.Ko +`/-P
-_gQ85GPL$rrl9tBF|2>:Y[<'-/
yq>#L P3k"lE+7pJ8y-aznaX
XK }F="|=b=TzB~W cdii
1n=j1:+G|nIs%BU\c@!>0h://8cVm;#8pfC~ >["(Y9IXdpeRs3eNg7 c'F8cL@ <H]xoi]7qpC0dPoz^+%IdI=2[\5[ps+0|"
'D@?PKh!'=;j$ E1o<=^>053Rw.WM&5>}k@/htmV/ :hrACF}=wX5?C.qtTy{sf{h(&NQ5/& /)K^ivoG'c=3X2?
B~C07<V"Rm5g}>'>\"1Wp!n6sb}XgBl '+G}h;#UCdwKOsah?

x7r>r1qwN0zLMp6c
@@b:i6&M;HFIGxF+OG>RK6@I4AW0?MV@qd}m(>u=nFhF
h
[2uV}6F`q'(5#|C#Ok!a*BQ8_{CA$z!q".'U3bK'&a7>N[e<9[68 &[agR(e ~3((>|{ja#HT8w*v~FD`QxE!>2s6P5z4
9`8~AkWG8o3;sIF3Dq/>Hu"M&ra6qb8x{}<OX01\wF9pW(2=0_op%``oB>Cw.uH8pj?)X Qy1dYL%zNp=P
0tp4(cw.Cn"oe~Zdvf=okP^[L8 ms&Qs 5v"+FcRBn p#w]/oDs
wzI#pzYrgwGsn &JDN
,MvR0a-U@=yRLi5u7Nr/50fkdjMaxu[*h~t=zc.{3m|{:ypl?
zPtVh"v]x]{aW0a4J7.kl
K\D0n4Q^?i:8b.
bQpxc8S<|.-JKeJe=cB+d(@#^]?YT4Il(ZFzB92 (6+^cG]3+HHx=yqa8e-
:b1A?~
F,>Gai;/<nOuG'qvnkrHvxyq/Tt`QJS;7K
Zc!{L2es j~3??.mY_N?G$\
q}`J9qAd(Qs;cNdTD]F]QZ$-V4KIR"&NQ`-9"9$Xba<BM`x"qa[B*
/bt;:,9
j)bX
T3D$#^s)GJ9e$C*{7>e1@C]J}q':\p~8y"N r)`~4B]D\:fz'rIit{hbiB>k&r9Rgqx1h4 K"uEZ1yC>0[o
Q;=_aBrU E&dBgR&V`
Z?qbtjZZ*8
M{=SBDq!"EVGbCyzwh,jj{F{Swmf4MD46lZ2sA>
/+S{zn?Dsi7<mB<~CW#DI#>}G@-oxv%T|a`hw9*TG[<*g)C>3c;c?=DXx>oEZ).]|jpEd]op7:U7p
FE{6%krL_P0
"`p2-r@"ya4Oeg8~4jlvX#K%R'F)7M&; K453m:klNJ,$w<_j6.<c&l\' P*aut9[WXU[* ;gb-[45
S}h[7=?^~+{r cyj_H+Vf#IK/W.K6No.F4'^_~J(_6Gjauh2QP6~lLX>{&\>>3V-VNe\ K
{u[F5i!>:p?26o(X<\la+*U,j
{LoPHz<iTIk(|"70nI=/!F,;
dmLG{AIj~FP<V`Q6T| A<`!
xa$D'4MJeEF&@mCy^{Mz'L[B#MIn ,e"kW>N^w!X"a&,jMxD+W&RX$qHoa
iQ=NG:23\l7|#Q8]4<B.3Q<U1QgKB%(%}r/[^CM-
xl.wK>;lq+Q=!(.z/eT@}Zb$ht6~fR>+K"w@Ho?}g7Smh`/puM\$5Bx8
bW7n_mMg(!`N<suJ"C^Y:(|uK35}E|>51eE:]:Wb xt4>`2Lh~} YQ<(< "Yli3z!>Qxh
<|$=/vz7mO@(mw"vu<2ftI]="R Ab"qp,\fJo7(Y,VLAtT^C'Q.LsMI7($=DT/"h{|rR "~Z2G]:0O4h}(NFurhI-
d,D~qo%'Nq)LL,Q(&KVl/x/9Wk3Ne2
Mf|{m{/Y)B^"6BHSLhfsp?iz@.r4w8k.il)IOT{x*hX\I=*/G8NjShS!Ga$2>.td*$0Ct7W;?b(C$
u\!K$[!q-zB(}yZOx9%`{t4-fZRp(qv6yAfZDv0|BJQIo@i#J5!{|!k8Rpb
+H[wP7=0rhl>C?x_)N>Ik0ga!jl)ynq21$g+VntArwdjY~"~hs[2; 9uvjEdE]~sHWEi(+q)sr(#@M.T+r^e`tK3[8:Wg9b
K6of$hW#5,P8^H!vHgeE7HCqGI35`'L}<8p?c@  o[:B`mGGgaxlAFyqn swLCGJ=ktIISg,18u0wBYw(0}0
^aL3cdti*Q]<,6H<}[[[Z-.1-r4$mwnS?oqDzp]'^1
+n$C3!>,o)/n
#[V>gO})'[;K:i-fkT<;Af}JpBh^Ud#!M|I`"zPF-
WTEDeDZFIx$;^~bZ\NP`ap=_}&giNdV\S[V^zwN<yVk0*TmjWC.+9^!aMgiLZ)lMR+TB3Gg8iS E-JVtx.<}-
L;qjHBrMSNGC1m
P(;}dTc~uowz0A" |Bnk5u!w!-;dWj\jGd%^z4yK;q?atZHxs06IG"+YL_h7YqLgA,) }N;IL!I9x%8)f}=<~C1VYP&xx
h+CH3eJat8#6}I$N\dqM+jVL_?Nj
$8g8G:;5QA *Vl /EbYi8Dj5S**=tI` "R ." dKH7!  ag&0&H.%[QE;T-QBi^X9b5h|4M!`8Z-
C$S42W5$xG)E>IIO%OagkKaXwo{7!b!3*o[_$tJ[;tAK>Yyd>)1Fo
`%MFYy-aib%Xao\C+Bj?Te*2R TQ+-Zc.\-Bn8}T/pMj%f

akdU{)(@R'yK{,0cGzMD{@[L.w ^uahi
3[-A]PkJurh__]64VzDG5Y:f#IjGnm1>RR@ ol$$3`yd1aY5BD&dD4V_o<zH2,} (,#,)7T,3mK\P7'v4{i[ssP;-Z;R2u

x%k!/i:?F-RM{L82w6hs8fFJB*3YPr@ C>w4$MD38D2AiC9{nn+~d32%LWP$ER22
S#)JU6h++W5~i ~;_8sYEkJLj#"B"&>muw<}M&q;eeS_dCc6y#e~Tm/pK0b$X:Ye{<C$%Rlw3hzy7gRmf-'IT,
Sy/_N4Qe+h.\lRj!{f@?kJ+"4y,|rc:$x|Snz|
T&.2/n>F:[f#CMHau++SC%<#&T`Pd!lY:LM=V?{0ej+`Xe[ZP"*K&[PLAN?\:`~R0B05
_n$*SC4gfI/*mA(xC!&JV%cZ.}n=Z^>8jjc5xPG0O=db2 OI>Vn2Wo
SGT7WE[60}N4!4Sfj~347Km7(sQ.@GNVRR/T*+dzWN .p)P;f4klkdiNj3^<>]vWX 7@9t$CZ||r
/QQ!U=a:0v,{GP-I)+i_-C'2vV!m+0{^IMPBh%:$o6.3xr|opq~mK`w/bf)^3[Yx.go`Y4+J.-_;7i]/!$j'!Q|=w&>o
5PI]x\5GA*;!1t6O_{nzj7'];X
GN_`XwH7H&=svn|md/\9}MuX!t9Q*OFr#7M\RkZm1PNu!<'3l/!X{xX4w=3$CbLwiM<y;WCZx&i_rF;{MD`X'/+
v-sqx>yHuBjK
x_Gx&Nh\TX3b>JU.JeKcpc<}xuND%bpA~6cFBJQR53G;\gn_q8szHTIU{+WbTby|^>#p3X2elA!*vxsn>7LYc~2e
(0?=_;\q 8vg/p
Ks0M4:![M
~l:L
g7cu S:]>z6n?xb8w4Z<pqH
ivis
svC<-wNzx2AJ z"?{{7PTj=x_/~$au}T4
y=\YzDU8{p','MgwQy\O^<>a3x$+cXf=#OOi&iFMkCogu
Kaz9!~(TD%rdJ&x>O0{$M
|dI8m}l}Ir1>lZ=YKl!+C4rEH-2IHIef=F<u|||l6;Jvs
0x]<mJA 'f{obg"B[gIj;<{%eDG~ZB)U2]
oz1(]4'{Gi~~GJ\jz6TrAgR*A`/|#Q`6mfE8{tJ8#Uio:x1hTy&yh\\h%p!jugqhBI7IN?O^!kgCiE3x)5RN6ne:Hs(S8&sR
;_~9Yma/dQ`Ym#}SR-YZ%d:)`o5v0}-T8:h&vuN"z1?GrCLQTn`).9_x:Db>1e<:#L+kVDvZ}\<e~bc3T#c,['0y`)`j
5$VE8st)`h%j/
Vk^-
2DK<%{W|j>RRHH\26cIHg8Rn[jE'O.(SN%"wp,ZP)(in%6OOYWRK,Y6a}5DSo[VD1U`\;NTk>,6B<m,S#R0wK
%4Sf
n]p3Ca<q]&GfMI2
x(PZ6efTPAH[EQ8}
^.2=c!.vSFE>rqVu94
"cf@G>LZtZt!Ng[C}0nZzIR2Q-"[R2+q~s-SGaEqp),3A5pjUWi,h]RYUzebR+J]d1)J}
vA~S8A&i~D"p@\36z9X+KgPj3HTle0o
p{1/X.\5"L=3{WE0o~Jx}_y3lX3^3Ky!&B(CsFTkfXQRT(Y88c~ZO^
-P~XNuO18l)+P.s;fL4]6rp:g3ha,^O_CiqY"R9Kx50S<CWm?
+##A8$3F"e-D hS8j%c89;u9'r3v0@fu1kITA%iT-`L#-
!+LBiNOb;S1VG)De.`lBAUCo//G8.{l^tU%?sJfzg9t%,LUvB+7-m<fq0.N0SmNI
VnZ[.i"0S=[;3&z+U&?'768#o_$Vm+<~&svbR3}YD2G?I1asVk\}Y%5[
[Ei IALiI^}RF=1?
+uN])]P3XF\1n(G'=[o4h<5**w>KF\O/#TAcd?L&cMaeEg+$E17AHUh>(R7$s:FBB"RD5'6]
QC~NEsakAW!I<I8+$ xn>E$ZO5t/Hh"OM
J!7#%uv")4VIuqC7r=0Le(#QS"qC]+!/BVQ6rq,"QjIOt$9#mMzDp&c~C ;1jE#+(c3oik;1ZC~9"8Q8w4Z;#%
'UD]1Ba1YU$0dBx@62S#o=@IDAT)Z]?h,X?T8D44!}2OUxO)><36):~EF^hWKkI; dEKv=KWGaX1d4(NS1w
Ah'*Y,xbY

<`=4W`Q5L_I:Q0Y8(z~pri122j +QaR9>^1Bm~$pnoiS! 0'~F?BK:hk(%!Bh#*,xmJ{(q|t m0

671X{/M(VSK|`s
49'8
s+,l8Rd";"0v>[gRc(%Bk8^)},hhKy;&h iTb^ saB*I4c=Q31jel5$`df`}aV?t*?yr Aqfif9qx<h6wA=.^ws899I{:&
k}io|\+"dDipx:/LdOn2BuQBw5PE-q+WH5@:cytZ.kgT]a&K('=I@FLHPOjI01-
jcDK(2D%5PK}59]U75YH8j'5QW.U?}MA*aN1-x!u!+uRJVBk<LvUk8Z]?<"<n2 `dkS<63z6"s#iQ'A Gwr
I}M1$}?('
=sqnT=bS&KgdzxO*XJj!D:]%0* tg3ufHz4Xx:IN8tqG;0k\%GRC~$'O6&9y-
Gi%w+~LO}ZT.+_!DHzQC"rAASeZsT\KICr[/\?uF.V\3) Ojl.M"6')>kiIRsZb"
M8{Wyp+T;XL616Kw?'HiGrEe`:lm\Mg{(|s8OtuX4>(:TCAtFP[/S}TK2UX>z$ntVI
>o0cH1dk>U\yCd(ult|r!:/_E~e2_6aZ#cRgkFMIR^JSL^\3"K(<h|UxbsfdM/(@&Sq7<U
0ssYrJA!g\|9nD3YO"2e_>7q!#bGR93'Gnsa<U?H9J$a%)Z0ClBZf_b>/T3
(yar`Q K<Gk6D[;KsX3vk40O
;w@LnO;r@@
(P @@Ooq##03SV9Q]3FQ.ovK6SgV(P @@
(!m7$2
a4JKo>CQH@@
(P @@Oo!)_(\ E ^x<'o")(P @@
6`Z[SW
;/;t+_S[
~6Dhi@Z=<t3S4GXOPRgKx-A-a*Iui*jR_|aP9?/Gh3o=1[qU)
(P @_
&b>A.=-1go<)~Ns
G{RXz/>wwbV5_r<w"n}
?(5~
lufZzt(j-Z`i<6GMysLY}k<*6lB(B
 {a+'"'E{-!'~gfMh/;;w;g5o+-nC*aye,^HRsWXda:%!
R]0eKB{Du}m[4Xa;#,yyH0K@,.rhk1DnXF!'M"]Yx8ko6DYyy0e$VGvD/}4KFwO@nN!rn3?D-
P`>|/3"Q\<F5;%R(C=J%|h$@$P X
r.Ln:7idof:dU!5-
iW*Qq74+M-O
!X
|KG[qPYd5[P"s>f|wy9P=7'92T>B^s0
l?Dza
,.wA=r)ce5{'>MrU>/X!_X$OcP:l
PiT0}"<,RY}%ucKGZfukMT(wM:_Vr(9;O+{Yxe=*2E~UV];Z\q*S<e?j*W3XN?>3r2)VSjm+28x^cKfZj
r|s}\Ch|2D+3/Fz.
UF\o=fMG\9'3i%gS]"7QnU`A)-/2dbUH 4/P:nu-?*(%^
eap_/}-Wi~.oSn
f25?}}'1ru!bcs:"7\l;9=Awhme??z6AJt8MAn#ye#&>
;5kb`[j(gNyCqx-8UhOba~jHO/Ek@)$1ni|3k5n
q<^rjGcwhqmT.E,Eyl;~5{/Z0gA+:M^?cVc*o?Z8EVqqv=HKL!>z +bmg-zJ6'qWqa-FD0g!CWl-
'E"N=E6/DA9,zZ6WM;c#IlgLND*?}_-\CO]H#^v3Q4)?kO?#0o2s62&vn5e(yq_*Fd!x`KHcw08y2lo2y}Z'p4e
hYojIvQkQk.DJz^v4ni9oz
oBDwNCLu/-u?'01yT2*F}}pUX|:$ec   B@yAa x\WW}{OWO'=SsTl8,{36cH+h5;V'Lh~]crxgF>SOax
#.BH_SepMx7?$Ft<hwd5Q3`qp6}iXbirC^^m0uSH!A%`bTlIOX86.,a4-tY;qw+_q6:F3F%cNxmD,N&mw
W6Q~2"?G3D_0~3dsr+,@=Qf7Fg60mDGt$n
);?";zT$SC~BJz^o8ky+<^ $j
Rs7_v#z;c^M 5!\0XW?;oMs~-N)RG-zZDoKp7`KY6cw

=4.hO^Gl~%0v8Z3c371xGy.X[7Ae{T5vs{pkzD?\=w^|c:{xw.]3BYl_3wurbwl'\Qs*m0{1q{|t(,---;O"
G_wxD;p(1JRU<8w~x
FT
qp0pw-Wq{tE2d,^}>s$
RO7;'hYlfYOx#= Es=kMkXZ>O_W:*| 4
+v\q5"ioK-ogNDX/IV=g@Q :cD1{^QhM40%G;:aepmZK8^0bTX7 q8]n
K]~V.^anNLso^`mc@nX]b7ZM'J~?!EsZZB
=j42cj"F)O)]l;e7|umg7!pCwh,:hqE+E`"V~xk['4TMPh&=`n/sC;7/VCVz-~e%n^$=
V4C=/2o?Gkq4#?vC?WZi[Oo?C4o];5=?W_4!`~q
<$< >B'o/<9R;zf]G
8Y^6?D%Pi9oHHf`O_a-yj[qG%3O/nndBFw<^lbB+-*'S6e3>}?YU<?_4oR*@zHaM2*f`4vL"s)lgRZT"-
G:wYAm=]2KN[?N\3OYunbA=zKI
5eMUBm}uRX<<h {p+A'ArH<[QT(7)1
m{}%\%I,8omb
+L*HRf?C=X!;_uf1EI7#cqXcp>:0&(ZEkGTH A^~Z$V;ekQ#.ty5E0d&2@'0(-;n/s$>3-iDLTzim8
^>]T./O6tEQSX@eNK,_?{
~;Ir2,MGa@U"{ #QVDKF; @
#POJn6m[VQQ3q/I*Y=TuW%F-i
(]<@2V#Z!lFGZ&VK3(n,,$Z@\|1;f^+O;-NqJ~O^ %(DiP8|VQMe(8jGxjv"0~Za##MtY.j>C:pbsC
0"K+D9E {[>/v0EP;fYT|;|yvN
[t`
({e>R_2/,3"E7
(-_DAdOZ+
:QA
0i]fr#H_~T~=Q(.\0u@@n41   @3U56&H[*D O6VtkkN;Q@,,>yp,
zKr3_uaSiW\"C=yPJ~92R`$8=\b}$@2Q.&21@36l*/2RQ\lU3 $$ .5KrW_^$
'\~M{5
o.jR;anR77>X:DTzOy<b*DJ/x7mLk=k8r_9
]<#[32E2E`xX|%nhpz2*wJ,r(H gn~+6ge-*>P;.\|x5lgS_wAHDc"*<7QeCK_qys;Vu:j";'# jJ@/#X*
R {!7W"q*IP\eweGH_2g{vB7O%Ybjv-^WG_~f0iKx~0F
9a}3!Fm5'?~-euu5E\1zXL}5<CE)Iu
).Bgg(l*O$'%wwzmlUe#.[wah~q/~=:QfIcS.`t7(c,+MV1/vju8,2i'<EY[-' _g1Cz9nb%o&-&vj+
}JF%cpxs0Q2S9,OG04&BBSa3bPX
6_|.V>#/o;<J}1icqbh
R;3."b0.-C1;o
:S7e7zDKI~3KMjC;5
lo>v,zneUx\{$J8][zw?6Qj}\9;' jK@I<_iS7$s\Z`
 em#jLDtx+W3eR(>{6HHc'eyU` X,a,KFF_;K+x5{US/n[#&OZk%A/{DD-1o~5-ViQnTWDd9b}`\{>^
y` 0>xlD[<`7F~$fNJ]G\?Edh0uRnrG@(^MulCzxZ&~ImD*eRsDHt]&JN8_ HK]F<<6DWi.M^QU?
oJSmQOV&{f@s,
v{q;qIdl/cZ7BaW,'eX>^qxe.^/.5RmI%."2=N++!-lOH-
]sF}o>=]Mz=o)RM1_W<R~dUi7~
xiS;w_`)Xxxn3D#>L|@gxwlxXg;K}{z2)CK<Q<E3&xU Q!HH
7,]^e,sem\"eU^0p)!YmA][7Yd$@$@$F
> LV$SzO6kA9N_B1W&M7rQEU@B/'{_w5(%$ctU(,uD_vMBY
7Wbj)d"2%^OuUYP6
u"/3/~R|]g
:q29WArE_*Lfax3+J6$*tmHeg)NoY9t-Q/'TKSPbj_J/O^r,%C,

bS%xxDQ^BTo2i6Y,7x*Uw*)B)qJ9QAYec3@U_l)7
L;,Z2y>@uXh*_PF^2E%}M#_)%KdwLW}C=h3T324W1xS zSm]9'e%}!44TKTU +X 3ioWZgU6#g
^ec(o)Kb.Eo|K?TX^^:+ o@]Mc[^e;Jg((,QiQ/kr m&QJ,h{:-H)eQD)wJW@|JK~K]m7 _>Fb4t8sz-R<+_]~~+6kn b5"
TT\jj$@$@$@$@$PPqFEQIHHHHRSk& jDK5,J$@$@$@$@5Z,7 T#T\QeQT gIHHHHR*
@M%@<M$@$@$@$@jTYHHHHj**.5Yn FT$@$@$@$@$PS Pq5r @5"@UE% JpA
nEzG]W$&nlpJ$@$@$@$@$@+.fzCtGZn6>My1
hY!!3&^g2HHHHH:EX1g;Lr D-1<}<\A/]'OOK 5 TRX{.'^96
' 37A>%"$kY#WJSUe#R{)2@b2@$@
,8rGg?5i:rryFqa*(-vF^SZReBQbqx%h?$p^HldyU;yO]\JMwZvs(FmG1s.$<@guG5G{ E^/P2_w5=;R>>\+a{:Y;4
a:, @E!iq:iRR$ EG-xzp[lZFCY%S99p~\J|sm/4
LEBj(wObbeGicGOdtz~^_CE/#R<eQkl'6 ;#
C@|exuY^<z>-p*+otF%*NGNzIwhn:7Y9D?#jXKByHE=Hj$=WMjWB_%P*G'+\q[TG-{0^\6+*?S)\YF-
5%R]]I&IEkqH* =`^+$U/)Hj~bA|FMRTLB
#0]6>1|*IJq)B$PVkLr2i[5(@$P3 V^
TWMf^*?jvL/pE[1i`7]
>^hoB
q_2jD$@z6EEJC@i VQ>f|*X ?ceu=M%upL
~Z8n@Y$@$@$@$@$@LqQ=OO=^X($9AcEbXZ}sdJHHHHH@).*SQ^D]aSkq,9MKCrq !.be
#U\TqXO&;adY!NW`C
IHHHHH,YUlHK/PXR* R? T$pauQ^Rs0~g{+'^-)V:(R$@$@$@$@$@gpuI^-cDyqf{l~*-
g~q)ZsYUfs~>RsduK$@$@$@U\Tdr,\%Y%Pi8U7CNO=?EV~V+$S'psH?V"@isHH!P
&`%S"R#pEL&k0(s6'*Qa+d[*|WkG
UIGB:1HH xUZ]i"3^=Z) 7o/EU:LDJ(.*#N@r ABQ1FPhFJJdek^p:)y0AY'-
yYf[fIXdR`vHJADkJ'hK>HO419S]VYrb\
u,WLA$@.A /=73WS(kkY[Y='?Vg
y9h&$NGcYsV4{FI'P!_2$#Ld3)>6]p8-Z&z$>JHyq8x2FQr=1_,q)~GtzxIOlPe*06^ZzB|dE6}IZ+vln5bhEMi&.O
X08]HHBEnGGiIq~4
RlWhrdeduKKA5 \t6E j$"91Ewxicu4Ao2&ZKWlQ=a4n0Q}.DVzm0 #}}v;WDV=%i[:I
U/_OVS57iY#oIcce; *..T:X"=rwjbciVLxM<q_N=;r3[AH&~1
x8<[FEC<1<E,In00Civ~RVND_|F|u5CLU^\^yRE_4X/Z&^zp)
!A9>[K,x^h(O#,Gu0y><6=F6kVnPQi pA9PKCphvkL-:G`~-aln;7<Rcto" --.i7l["-
aX[&}axl|?n:gk/?MWo@HDSLKSaXs-HCfn<_r+BB?Q$foqONAdalxb.h!i1\G+
ExscNs:oQ/>0"&\xo:zS)vALF7pq{wGwh5'
o;,S~a48\/Bxm-1ya0
ddQU&Ea.>Yq}[Daz&K0gqQ\H\@[~#?n.RV\;2]6F@Y>)1~,@#'VoY1$!=OC
qS(NiU+;EbkqG~nD%N]1`Mb:3 \X:IKaP:^G$z^PY"1cU36cN:h+O$[2R]'.U_47=;aKfI.ZDmWXt(.n-
!N2J>??G{<z,dXODMX!+ahnCcP'SZ67AkGLVC~<$
"PAB/^([s_zK?FGh.ng{n+NM/>i+aa,@2CQ@bc"+@DAXwCbV^Zf5OdnfT9C)@Nf]:i#V3U:wyp)EC!R?0Oz'ly
an9Z=bE
._?0%K\$eF5
dQeOdnX;DN*UTP+"?M
xkzp[pM~u=rEI6< "[cYX8>Z~2{
aV<gQ#2xO/F2/7:Rd~PJ:I\xU!:CwVZ*n]DBs~]LwQvEC%q:XILw$vD@j{*{UVV
se@I,i<=q*% 2ro=
0dwumJIPc%AS'{FM()g-q,f`FGbPwYIL~[zm |-O/l,O.=G$@$}1 F,28s(.yE6
AP_$<?Kok@O-iq*GNWaD<@PAd""@t9f}?X6D,2o3/]6sT&o7~ww:~JV<3|MVjd?nI/2mHSh2DIy[Glz"-
LQj:A+B0m"`r`
Y]%zCBb2$f nhP(`qdY]FFuK1xg|je&<zC0gfa0,& JC>9?XG|I-debLDB/VI2e71`Rli\=k~q &

nWdl5w59;K~68,XUZYJ,g
jHKxrlyH^2)J>/9Hs+2\g]CR+Y KYQE[SEb|\QHg`k;(}h"x
2}mu2oF4GC R6e^ZI~6j zbx`AJ}IDAT&at@[
L;,1AY7/kwcvy4e,<^#~3yU]p])Faw%*kQC:xqXN[a' ~>fr$@$*~zN*D']3dDxtr:fe&m.z$&}BWTRb}L9)xt"4-
&GZ(4Y%hP}#aZ5Uy[}A|.Cpw4\FDl@o<hl>yfm$$Mf1Zch!~fd|>6YV25e=m#a*]`h8><fG#>2&
+7q;|oFV"IG7MD>qr&Y|?4l>J#2oIt^\jV]:C<s+$@$"j1)M"nZDxkn|_e28%m}'l(>jd?Vr0us*DE6[=Ox|<KYs<B"1<
7. m
v"=21! ~_[F
?z5t(,<PK6 *+*6t*c%qxAM)DmnV1L}P
l,(eO9;&rrCS=8HH k{BpQVc`]v7Se KFd.if%G{9f]1XH~J)ixoL,AY|4 o\S:YM
`XAXY|`A
0\?*\FF
B.H@{R$! =6hTa \<S0gs8/H 2G *&@2; '@2G *&@2; 'p&TP3JD|%
REqX63'3rHHHHH*A(.h&6XZ6a5;q_U/bUZN%g g~(*!~v!n!JSZCE :6N <"@UI@Q:ZKKU
!N@_<OyBWEv(_B|wGS0e~&<+'Q&9m-/%xpTP#R3.K #_|m.[
F$p#2M<_=O$PYUD{~fP)kt*C-]cNlY+wgPZ^=hPoWr \`QZ]1"P$oAeJNzTqE(g$e dm$p|k&5e)=gp>3rRt^IiI|
HCoEAZu}i=_\j/2ybF.f~]#aS7C-_E:?EI:AFa;elHkC8?A5] ly
FGyd}P GJ[Y2\=_T-LEU:E&C#oVbLhMO
mZMLB_4)FS;>+u7
`X8\yyjYu/MZ[*E&MYx +$koiM&5RU"+,sVnwuKJ[`yDiPwome;a-\B&ucxpV%rIN< #2
3Ev`ZP W]krP.u:=+#B $;`Ln.%Kji$nRViQnRf.Q\T`LCq5;&ncnQU\zFdFR-=C/DN,HfPp3hndU^jJ?lU*uSE
%A-SF2$PUN=T#@$@$pIf0 Kt2Q6u^X
<P[(imR&6
"saee1uoV/ 87{pHLtDrE1ty{]'2.(<O.a(&%lx@$@$@$@$@$ OBJbR,ugHr&R?>chgSV<
(Gj\3IeEWZ#ZEP@V'c TqQ+751d:oy.yT=x6,^w P*FY&:|6z2[v,DZVc3>p
kJ(N$@$@$@$@$P T"scJ6|rlA<1hJ>IyXw<O,TbU2 U/n`j-u5^6|ow#;<hgJbNQ$@$@$@$@$@@+.*W
D9olq0N$@$@$@$@$@.pP)'v-=gHH.,"m/^`[]MHH.V+/GSA6LHKKUwf pRRR/Ba_$@$@
RmK_pq,.B];?~0f$@5z#++r!Fb@??(jeIj0Tf\{Nn05:l(HH ZXhR]WRHH,U!$HyrYpww4E5A6FT
%KHvV=B].l/mK$P8j J.8rHCpSq*rR^u
HH pIT\@?xU|9(8G6>tF$@$P)6]@++ jB0W)& @1.[HHHHH VBZyLL$@$@$@$@&.VX 8*(#;)/81VY
y*V%& jG]Q` y:gIHHHHR @#@9KL$@$@$@$@jWeHHHHj*.5Yb vNJNJb p
KRq1Lp3!$@$@$@$@$@;4I88 "@ 8#@
..UHHHHH.0HHHHHPqq0$@$@$@$@$@PqqFq$@$@$@$@$@.EKU! pF3*# p)T\\:(3T\Qa
KRAaHHHHH
HHHHH\
C$@$@$@$@gTG@$@$@$@$RTuP g88 "@ 8#@
..UHHHHH.0HHHHHPqq0$@$@$@$@$@PqqFq$@$@$@$@$@.EKU! pF3*# p)T\\:(3T\Qa
KRAaHHHHH
HHHHH\
C$@$@$@$@gTG@$@$@$@$RTuP g88 "@ 8#@
..UHHHHH.0HHHHHPqq0$@$@$@$@$@?.{IENDB`
PNG

IHDR-N<sRGB@IDATx`TU Pz
"X7mWw-Zu]ZV**CB*!:sd `dfox{k)),]s!>pt ,xa~Zw' @ff&@l7 tr-: bHHHHHHHl%@l=7 tr-:
bHHHHHHHl%@l=7 tr-: bHHHHHHHl%@l=7 tr-: bHHHHHHHl%@l=w
i*~3?:a[XTzq

CYa J^%%6s2s9d
c,nzkuO*%h8N-XSzpC=eR(-*CR7 8'9P-/B-uf_ha?7d"W~6T5Q-pELd?4UcB,[KlByc`CR&j,X{EKLo-
CGOazo^Kvm9BLxs vTLLk>}gN#1tK}
v9]IHHNEg#ybA
v`u%xogARqfg_g;sO,\7gOO[7M=Npu?gIgQc7Q^pwv8v#;^g":;tJa8t)Wzk 7Ln>r.~$N(>
)bT[Nv t.9l
zp@`oq){sFl4y>KfM7
8{#&..<$gerDymRR.iBKJtOvTO=+7_*y!
U6j xui`\ub)OW"*
_n=aF+u X%}qrcaM}(.' 6Ar
6%$"Q^fk
2=|]
;vR)z%4Q0u`xS-}HCr$0q29X;
{^W O6jt"aI{#,ExeF{e\T Mp{<VB#bKAnBZ#,k{SRD,GU_.*(k{Z\,bzu{%rs
ClzG_lErup2PWup!T#CvBy5^8o`~q(.]q9?7Lr@{vIg2
aDqL4Y+`~nR1r~fszj$Ag LR1wj^y\xO~i>! *Yj,DzVctNFNe="q~.u{Lf\0.xX!wpvw}qno:fHtpBt-pc&KHL
!#qY~7>>z@Z*x/FqmX=bw"54n[0cD? [JX</jFy~`;HHHHF<u8*d,;z]i/
e{-
E0=Pfnf.xU;"B_}eTZS$<<'xqdTS+V`|Pwbun5ePY_{!9v Pnz#l3*\a0Ip@V~.pV:+rq\{bEr:&yLj~gp(?}IbY
5UJ5k3}"MNO
^7U/UIH8+M:9RaRX%mzPJca{Tkj7+0X68/^*7WtE6:&o:D(iH[<9[o7Jex`.Lzs5Tt++,=+R(u$LL&9qq 0
BNw/=2)Tu+K,:j6uDGi8KmXS]v#LlL2,5c\C;;a~J7nn*)+^
-8`E"~8frIyd\ab23;M/'nEDr=1~8_\2I9?n)erc[5q\{gRQ'E sK`n#.@;>o-8a">2|V$b3&ws'ow
]P[WV.B{P
P,>_B(_/ 03~M)Oy&pgo=zs&<m4wjY: 9=*c#QpP&YUrN.dv`3~yx}dA$<>c<~=s~MC| (O5/_vSv
#%,Xs0"GFb?ab^5wjbLvRs21sTYa$Rlo:vhY.gaR?|-entG3C&OT {0Hdupu}k@xjO&JWO0l`<8*sa#7'-
dhI;OE>>4e,70kX\5MV#
'gJhl:naKAI?\VUxXcMNs[&d
lQQxqfejQ
uxyNt;}7c ?# np8XD{. b1 QoGO,chZ|-wO^f'B'1
C|p{cJcG,c?6gm7n;r,zzb}02S
(n%extI<.Wf
2_}3;&
@]#bLCq.&sD4[sqnx(sktgX*@_,ywv,KKR/4[$.I[qm
+K}uC#&a={`C
1d-IAXyyUU; Rz`vrgj<"|?#"bjs*&gf4[7bh1|H0Ywa'q">Lvb][2nQn5cmW<BxkfS0U?L^:B,;:A/PA(
uZoK$@$@$@m$x[~GMj&4%fNal!+>qxc[LPC1?>W#$O% XUCgwOLs":$!NM+P8le{Ey"1G-
Fwk.6=+D\D#WZ{} ='c.,4yzbWu+O|@'b(hu QA2,"!{rK>m1Qon@S(&$)r^xly\,"*8yZgW/bZ?_|=n95O
Za]7aLJxwQ".rWWcAq-zxTQ*JP%VdBCX|wQ*]Sjym,\7&B?
I`etmYVy==bV[2DrQ4
ZQ(e%P'lE`
vkC\p#X~&VgLz
;.chEXTR!MJZZ-Butm#c4$@|X<3NDK<TH|~rs/2nnWbd)Gu-zr!AVG6[9m2BYERF{|{B X]sX`/iQ!/|d Pa-
CTKRI(j>& h+R|`q*EgZ"s2?
><*
}=n<]'$X!+t
D<0+d@~#B7!1?*Xa2qE(sKr1_)}OCL:rdWzU`A!0Q'uu35HL[f..H6^Yej&1n)$*[N05Tb=l"ZZ,I&cWZi
"x?7J\Gkkca^[}

2z!%# $kC<M_Vt eL8eYYYu
:Io$H|Zq+YZF\R*G&j_,h!5S'Iq<_rP>n mvo&fd52{A;t*iM'MIuy *H,FdqqAQ+&w9A
MP xA@*E2eyk L9bQ%K&{q7P-bp~[mvp>*t?%NF"@B-K.RZ\s#1.Pcjy2wRm-gC[~LR./S&s>)+r!b]?FT7nb"
w*2tRU)O3F7xT2iqC"pj>i1UX4O[':qk5?L>s3N1m:nl
XC{P&M^;a^`/kq$ytao$]yuFRmR~.<XfOh89R4*p4|M_/K^H,%1K{AJZ+e;A}U5I'$'eO{-
%Cv}^s+Z)#gUXsg=a+X7 &m OVie]/)#q@5]B3V|5]0myE:<*45eW?jQg62LbkQL< n:/N6&lo4
zN=q|jwZ[woM,i"]wk^nETcpIcV[#Zr>N[^ @k?hS]ox4p-K`ahOG+x1LV$O'(z-
xW]2VL,yQj't|iCZ$"`0O$P(O<@VV&~NViAFZM>,STT$GVbOP%. k2r [Cp4FQAmEE|=_\5Jwh
o=`}"^'IYJv
=j>o&>ObaT%Sw{aSY!b!62/Br{F!255O+w5TfI)bX6/?2$-X?o7OlU"j*Q]RtW9$yn5aXx>7P"
]972l3e3rDXHR12]%a/AlJ2al]jZh"L9j;X,;kHAMFV)7
sD#(1MSK<.+u}ATZF~))92]1a`;B)zmr[w/<*Gee_5%e;
Iyw\y'.&z>*FZ`ZmI\Kr~u{64YIHHH@vK~fCN|NYd`kGqX}<m\^2:Q1~pSB<frd"4m5VYtZ/_r&6eY -K1jZ\
6vA\71z`4vX _/Kc<q//FbX6
n#Mk4j^HJ<rdsw>i'f
/daIppotPsCUneOtJ K&K "
hTyHneyE)OdaZQ)K
Y=D
Yexq{"do*Oj:%_2e)i>J\gv5Q)K`9]2nUa>bQ-,e+&E,@a]RQs/GG:dCc=t^H+?wiP2A7>d4.R]yo;7;X w}
~b1mpax~R#Zo8g8fWpEI[?Io,@ OUY!qEe`xlu6H=>ejS\.^2xqH#U2K[\wn?aiu`! b^ *ZY5R>Xao`zd|.r
s8'e/MmU-pXAS3Y(df7YB4=Gov$q/$?,-w@,"VGp]oi_ 3 @[8ZJ
petQVbY1Wdp7o['V'vm!}{KcL@O$ez'5LFh@T{xbL4ee!6T|0ZKuLr$UcFJ<9f#+65%2
[P>]<5!b*])pa! LW\^!A*]P.Zi.uNrHc<)3!uLIo`.)P?/2]$AVQ1yr$%KM?2a}R6
+UblbXyJ5
+-os}Mb Lh#9G8WXgeQ 9Qa+$db ah/"+/%
FwAIY)r6fKU)2HDw
x 6IIdSyb^$P'q%^"R+mZ&"u `[UI{(Kl,Fl<LHLA3=(M,e!LcYN6mm
21 B#{10mDU"HF#d;d<+O48KAX85C\(CU&lh-
XW2wePq)6k7U)(Q/=T/ns/4mNG|k
0T~#<Gj9.9>WGx,k17eQk EQ]K& `,
[abL
EX\;I8tg*k)]sLKu%V(6h\vX,}+uj<'_i}T',Vq!0,`Bdr&5ES\KK`0xk& 8AN)Z
trY}l!;lg6Lu!F>mjZB<t;ovD)#? Q6iSpW L`TWnX"5vigT Mt!i,DL\}e?o0BToqr~%@tt8N@8K|/R$\^M5
U:ho`s`qVFvV8&L
'H[3'=C,0 =##bG*VN5Z]5HHHH'iD]VJNlB]@"#ZO8Q1Dc"-w&vH,IbU/& Y8%
8t
B}x>D8L3Z:?J@: picb*JWw6)[vKy0v]</ @l}>'[THu!t~pE;0:9n_}I"Ww
=,HHHHHH:=<@6]+/`nlrhGt
jln"
xel'leHHHHHHHliap:~X!8;MWJWN4meCHHHHHHH$fnni/^UEoYb@/0+tM}%JWD\i)`#(/1zX1qYw(VW9ZvB2
ND$@$@$@$@$@$@@D(t
^F0neU"Uj%`8dT>r$jmkF34.#ZI*L4
."\{D1LX#RtH^IR
AJ@x{>U,*AQIuu;BjUY[
b@pRAtg _/RO3Yg 8p_t}_(Nuw^&=nVn~IW`L(;b
oLxL89CHx-
[a&QF6t
nAClP.'9""|l^~jjCa9o4>j

iX)Do9,
W)DHJ@K
gX0H*"QUM
*rw"bRAP
$@$@$@$@$@$@$p,Z4#HxS}os"Y4/l<YvITXzEz(OnDDq}z`So`$"]4k_?3/KV7<
Zz7C3{#qFonAX7*\1Kfj#ogMc.bC$rs zm-Dq7}WMm->@Di1iRL2Kv- o\spx,
<(bZ\$<ib}q8iMukH .*E \;qB|!bg{(dTJL9uEu-}xxsm"4w/1- 8:\XL7w@H>,O q$v]Bl[^%OB!mPS/8l!|zb_=zv
FW""J4OqqFEEqaQk5[$#@_lk.F]NTCXZ,#FDu{g!
@8K_e_DIny[c62%_&Q#@[#%hIyb/q#BRj}Z*x`KbP"llKqC4a
wq(n"')h:u[@fmX t"m3+8*y/V!"ZhOAS.AXB?-Th`#
`4"J#GbfDUE}f`.vuiN-%u1Lm*HV)Wx.blR-CPoxz""Hka
-D$@$@$@$@$@$@@Pb7Tx
Wt FV4td'K-
5UQ,n~+gQ/BAYumS~kBXyHSfUQL*r14^.)k""7]=#J#9HHHHHHH3pB<<ByjAB]8/=,EX)tdP[hu?NfO}Pf_X`4v
BDpTum$gSq%SKbZeKGIRF|pJbet$DCL5F& D:T(U1jj'&U.^7O~
G{
nY,yq(dtcK MjiR"AV7o,X#<WOiDzl,'yOn,tiOBGXV$ZKRyHHHHHHH
:BF5Msc4+*CL5U%B_qx/40BYu@V^_&}*'XLjk'<CL@BcW `:A5{]^mc#j"K$K[& ,:DD(^eEb
^=a
YMZbN<"~=G[YE YNTS$rDbM+hKBW*(HRkt1>pUH jKF<jdaw 9 {*X^{d}R%j>.~s9x~
(`o`|bQ^|-
n![X<psI'^c23+4HVK(%VqSbfo/*HDp@}Dm%NZb?qrIK2)yb^/w#X|5Ho<b,bHHHHHHZ!juW\&VGZPKI[>udx\
6rXA1]m!_/tAFr *.d_

Kds8,XXxgF=_Rf)qQFPP2Bz{qM6YT?.Vq
lb1]:u.^]Ba+| o[|' B,- @&\%ABUVAWNRV~krFc`Wd!,`$<Lm8F{fb=b,#W$p4@UM-
*Z!cZCs!bL\1@w8qe31;d\/C!Lr1$Oz^>XoFFffhTK0*o~\9
eu?QA)Q=:4QKC'{IHHHHHH8ZJ
pekVnx>*G1sH H[&Vw'/47lFx05$_I%ljKc
_Y6[V
qaDsi0bA
-v)DhmP!"O 2e{H< OdMomiV`}).0*ZH
"J{JjP"Qmn^Wg.lKcZ0 v233fmKQkRV
\T8_/LKILh)+tZ4okU\KI~axv8JzdJ?uZKtkcU1DhjcW(Fh.X[$@$@$@$@$@$@$p vWW@'LvJ
BpOCdNlV>>s+j8}o`|~yU8iKO
<HHHHHH@p'[[)5R8_]RHHHHHHH p.mRtZQ'=Q;wG @Ep,tYX)Rk>f
Xiw 3@E]Co-YPJY)DBLvBXWS,HHHHHHHhh!fEJH`feiSWo^.z"1 iKmtSom^V
/*3KHZ{2!t9<HHHHHHN3m-t Rh
(,)QVVt<j'E=4C T+5Q=(O;2% Am4w>\`.zD-FVX\ hGm-"-@Wy8q?a" 8 MPjD$@$@$@$@$@$@$@L
Df$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$@LEf$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$@
LEf$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$@LEf$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$
@LEf?bJ$@$@$@$@$@$n$D@8M!
8GGGC|:}<xzjO$@$@$@$@$@$@`X@'www8R8O&{@$@$@$@$@$pVPY= "g{dHHHHHH#`?c
3C$@$@$@$@$@$@$pfhqfGHHHHHH8-S Aq 8P8N);D$@$@$@$@$@$@ggyd/HHHHHHH#@;
OI7zQ5?,QW8f v!jE:D`~d@r _n"j.b!$@$@$@$@$@$@$@?-Drso/;/G28h2_/}QHHHHHHH#1A>x C5kDZt
1";.3=okcEf}=& @.nos,,kq +OX^^9v{Fvi9VL\iibIHHHHHH#H{O.ac_d[q17%(gyE|}Ypq}F{:@IpP
4Gm@$@$@$@$@$@$@$sh_BV <9+7/N`Sl o\4j :|x6UAAg8aXh\(^0?IMcxEf+FEW6a" 8Ot2by11M\Bzv
6ncI=u8\9n(55jRKS

>xpQ?#J7O|Wt,rl}c;G$]UK:l]%]6u1llc@IDATf4*Lyf#Z/$@$@$@$@$@$@$ph_BLc
m:=VW't9"vfk7a`.!lF;awpqAfb<;Q#s#]1bQB>,@
?WWawfj
bky"*lCyH)<(cU1>5Q}E
Wj4WOR8W,eTvbHHHHHHHhwk!P,jV2Ar#X<X>p3t
[R&"
~_^VK%V_Yi(J29
t3|,{i?%X9M;A:J%/~Z<qr<BX~eip~/@MUx[qxJ{_bY|&\:7{6o>KF& 8
iujJ$,|}ptlO$E=j}P/.*tZ:D{')YQ6oOr@#cD 8O7NoE*.X7t5W][7?"\ml:|],Kw[<,Mb!kbA+nq?B@d>}\,0;<<zTb"
3@[Z;`kabdH3_U\=bX<N;d-$CXW
"U5(>."UNpzio'
"H,OOyNyj$-~7n(akP"8R/5?k+1pxh:<:F0;%sG(RkOe)~/r}(TP_~qZX4
C(J--6%T{G!4P_.+v86@0VNZ*]T34X0m`tDrNsE
"L~>h%PNp,
~{?
rlJ 2wGq
9t >a,.>z=Kj`+ja7eSSF\)X"E$@$@$@$@$@$p4 tizaxsZ\2j*j)!AbZ4AqcMWN@aIU8?iE-(^dYInX,
4$cogO~ZY2u2Q'K'Vx(/GYi>^o]hZ}Y,J<Q'un7P+Zr(.&#3N$<HHHHHHN+ZL|.K>u<t(O_v)V`_2Luo,ZR^7DDp
Q!+'8~Np|n~569y$h a{HAxp0r2 .N&g[`t+Z:
[5k$1Y2?5%T+;6%E
e^!QIiw ,HHHHHH:/-"S_,g_ v'R]$B\H^sG|;3:B|"~|AqCqhc)~J,ZgO09%A8,BWcz2xo'>onRMkWE[]X/7
 q8? [#ByE@A4"q6HHHHHHHl"k)),]s!>Of q*&

b3 6w 5B.i:o8y4zEkw
I`{XDHuHkQBkC5g{:;=S."
#!Jb\RK#Jc)0o[2UxPDD@fF6ru`ky%.}iH/APp=j'6"g~="P
iCIVD||<:FP"\WojS+@/+J<bfohBO|>ytZg]U(WMjdly.,4<|{aq_$]WLsl-{IHHHHH-=Pr,i:7!'81}",EH%^-
ONm@SAAE%<*(dvf{kP~$ 3@JGbsGh <%,Y+.E%(*/ar' "6I] B2RfkWWD$@$@$@$@$@$@$@GxVu8
L$@$@$@$@$@$@$@$p"$f! 8(ZzHHHHHHHP8HB$@$@$@$@$@$@$p oHHH
/b^7"SV__os!pLz4xbj;k;3Ag3ggIH4`u-r39-slS@EJxyyA$tTdlT#::vicI'MzNN3N@"8k ==8
t\]]m&555&ct<{&K'U+www7lCGB@||<n\tE;~Jms'~w}{r#,/t"o>tM'###ie`d\q9s&y 3c\
HgIeHHL'74sG}SN5>;c&:Q;v,v]'b6l0Mi!!!JG8s{w>*|'fcxrhMkI/_,[S
o?I@=}lb$L4]v?<YGkUC"33*`(V;{($@8_rq; t":
0|zl&:1e,_jPlL:BBB1>hj%|pZbx%{Jwthggdwc_*BK]T@=5ssAvo
RvQ?u@vC/F
#E$p'n2Q|mN7:a:t0j(FPi1cPW9A'~'/ME"OM}Y+[^pI=Bt1l3O{iSy\}k'7nMvU2j)WK'^S?%ZF$`O_L$@$@@+Z
Rzisz>jy
7Kz{=.TxxqNMM9s\17l-yOu8YMouqyj6:m4|2Oomwo 220H?1~xLRA6oltk
O?4nF*1 FEkdHH]'-rMz|Nq
?
c.tB>33_^`#|Z,<h|k1
ue766clk&]~z5YzZ
V$4?t_'[D^omRTTd
4Ykm;yMmKr=^jkcsr_0`yiS9o>>y1lVI6conj~=Vvf;MRtN}KIakiGt$Zvhm|5n)ml:lqe'-K_VV/\T_{-
WhmcI0pt~bMz}h~=Fh^-9#& N@E'8 l Dxn2<v9&/|=9xv&vsQ]wew
nt]z#|#ZN|tb&zLKh>};va'Soyb3r}asK5:5ex{{7Oj*1=ZVQEKgt}MM,*
B]ayu:
,+h)yaa(ce&c1m NKVk.@erP$&&w193kLb++iY{1/`iIikU\KT%%%XbY$??[nbESk=_

FnWkzn=Zu[fkzmu=vKumS}~k`cGJm2P&}5cV]ZukYXUZZj&_
`:lAIWMak~6UkjmS&=]1o&B9R'dfwSw_rzt@Cu)WOmMnmng"|(Zts CMn?
^ttun@L4M\EZ]o45FN/^l&'7oqoxZlpj,<\{@7:?kB- [o~\~&Wou+:z|Q-_-F~mSNd |kl7%].3h[m)
7`krk:gm['0jXTi}eHQ~5_uB&$nmt"{)NXTm:k~/NnZGEy)'vj<'|S} KKTWcS]hr.
`cF-O[>]uUZO]U@xFy2Q5@E^3
]6imsW/<fcD'O=nsN~a8kV^K:M[cN
kyK*=5kVD_+)jUKo=/=^y1yTrS@tR3o^Wb}]3n{9#4%=F-xTlcAWm?8?|a+Suli__ImK<^]UNRrtR7|nnk t
-:i`#HH#5
m<VKs631.t@wVng]z v!?5&Fc1?3154I4j7PPAA{}>w|<wa]3w3w3gf@+\FwF#y?Q<:tR1 *<!GF^tQYIyB
NP(b/.a+VDk5g?\(+Ol:x1l@<"<8F|2<QQGPXQdir^5-M=:}E~aQdA+g6q0n)'6F& VBIy8['|rJB
X0<2&W_}zhPbL|az7J ')'
zCv^RQE2JHw)rq#NxGQ'(ZU~)1xqm(c`q
CfAk|~F0V)^>s.6b??B0(d`ou?rc]F<c~{&6p"'RO@<N n;u8'V!93|&%T(-au?Xik1cs|+Iea<:>l<CQ2UFP\2K?A;S
HYZ#(#L`4k=t_{HysNQFQfU \KxKg2xk>@\]tEiD8\S.aqQ6q7GyLO
qH<pRC}L2%)|be 2UNY 2Q.x2rbBNQ4BN7IDEb!U<^Nwz<@a@`
A"

0 x?pQ)iysG
kaiI2c
q%D*_~H1\7(v`#,(/mY9hJrb,q0k(Ku$MEvZ0ki
r{B`O{B)LBf0$O~B7s4$(Frr-P.mJ}rFjro|,'~s_~Q;'pssN G@-4Y).*sf9r,NEe%yV|#ty:n2ZN((htq2:AGtD1
~Fh1zU8t* //hXpc9E(6of$WkeEe2sE9)v[4o@kc`e]P`RRCSX!{t32
2r!=`DCN.D:8r=e[5awUNLNaA2s-u1?du|5
AQZI'y7q04bhA.K/)`x&q?dB~m+SvP@o|1.=z"-y7?Zo'~o!p'sCP_BwL[~6d<ta`&b5
pFyB|-vy m8L+85yg'nj{N 87:.gmZUnZ=MChI_5uWni
t8tb`^|M=((lBwF]K5NEQiFA8E(V>g#6G)QQ8~I|JW,w8b0B)Aa.0PXP@B[@NHPT(?(<`=F)oyJ<$VN)!
 rb`,bEDB[!>-|r
+(a|6",Q
m,5c$FB:fs9:Z0&93 naJ<eIM^5y5o]q=uh?[=p9@71`x e2KZS2a&"vS,L!/7x0~b3uEh ~:K;]$N |u*5I4t,Vu_\V
9U'gpN.n!:vU=TVb=#rRQbG($8E:(W('QRBt1rAFC2tzciwtP
bg`J
eBy,pr8N>| ;y`ta4:K^xxp/jO!3HZ<'0PGQp;>s>Vj92!vx)PWzu;C:
Uh"OeJ,(7 wdF6daOf:<k 0-2;gm)\H5%0IY(x e8`mFKo0p
FZdnX6LM&>%gd1<tBA||{Mx>IyxpNP2sZ6fkV9QiSf}M"w'/El/kmO<]%6MiLK"o7QzP2f}@(W:,o -D
>+bGCT-<\OcPBP`btQ`<w\qfF66p#V) p~H?b\~V|!)1XTNJ;rbX=K8S>4fSLgayQ8=c4*AdS
_^Lb==u-+yp-cv!7_0ace+c\3"/!?i)x@UU8ESO^+c-x0Hpq,NxD0va(qgcl^@9YS6<)oJ<^:[jhae+%Qj78v.rN
8'P[[z@6Fh1k-2%dcyis5Q]YnCNGw:U:\u$Wp<#=:tUQSE5]#e(tCuhBP
COWn^{,YXk`".[d`O(8QPph#(0|)<I:P9bGv,p
#|0.`aPmh.: ~*@!O]ruqFTC{qmMno90Y>^axT+s{7K<~[cF
gCkjGlL[+o1N +^1Ppw{Dp3I3;*2nmU9'8Zf]B 8 8'PCE}pxV]31'DuTW<'pF11R CZ:xtVR;R\k"4Hg
s>Od`ACG,PWF/#n
^Pb)';QG)UqczN/D}=?/},'6O~%B[M9v;fC1`|yb'5H)!l-}Yio P@(yXOC]b6lEnU':W]yj:u /QV^;Y2r_^D-
iit<='@'5z6bb}Yn*6O)9nul|,RbA=$u):$Ab'{nyZ\rx_7u2+E h.>r-;:a*46
W^UE'.qypmekyK2baFDq>e=
>q#VXD.K.$_Y{A`Kf@QKxv6S_]uQ?y-FNLeyj;ub|dY/PL`1MEQ
P6XxqBf"l[uN\2c_UGFKYkx$^;SPx>/;fK8TqEwE4`
/
<W)-a1

}(/PuVs:uw'j@-JJ]K=v*OnX}GZF}
?i^-^g?;6wNjXi<CWc8=TwBam*EGtO5W 9'8rZu^_8+>:iSzc=>
NR[0c?t1]C N^!5YU{l3C`?wx\+*y
G}4RQXe3z'lANrr|8(Wau1zmc~C.0r[>o:8n!?"L=4";'[(s0~Yc22
>}7quDHVUH'H~Vu_q<ed=>ce:L`z'LM(k<L/_O_FY='G`^+~
]kUm3,QV!O8{P}:- vX3q{z]c7r4QU/-E=-s+]lKwT7!PjFG;#;\0eg&{/T9d8<tT`/]3:i mK<>=2p(u=
c^^QbdeC9hseNTbyqMFD_GJ[amJYF6lPU]W`#'F>O2qG}M<)+J(.Sn=2T`u`x_j`[#Ex_o*']uXC=D"l%)v7bS7_
kr``T6.[mAB@k8/8'P'TCv*NO}8WM;um1*nJ;~BW0m:=[GDZj5iedW\b/{H$k:k;:"CfM?'WSzF:4Q
-od;$6\:t8FqN a%6%Azxzoit^?)YuFzoGev1A;FDJ5rF0NL:|<< J)*Kz,fI!~WY9)'Sb-N>(_/{%{Nz
q'6] %SNw}TCyc$^;y
u{m(r]%655p2:|5Z/y\lrq@3!1YO6w8Ozrqrx?+b\`=>}Dl>~qETNa"4opp[|y!A}.cheh=1[y#3pX3!qyypNn95-
L/O_KyF
]ou>L=^9W?8]}[6-H_]~lKw'o8O]e)7_kxJMYnFuiaB5~n=;c**r=):QSO"h{M*{N 8L;n[#;uv6m/)jnd4L 9tlQoGys
~9sf4.#0FpmF`~_E
vL{" t,z-J
J

((:|z6QF{h-12BH17.LamX(\;n(JLe\+9E9k(O?-krQu*@Nd*gu^rRv`zd
Qh^_uUQ<ckeInV@=Q)~h; m3uvbV?F-d` -^P?#3O5<hgo>jJ@DFnT]xLi.5o'cc"xGeI^y(
q1ZP':33~\sM$wh6'O1bD$#r^x A{F{`wd'
ua{E]n5\\koFdKJDf@0Z/%t~j{Z\%sgjjn'Q#A.3>aX'l{6L$gM_HxS.#O?Zo|@{<$eZ^M26-
ej"Mzj:&hI#u9j}3=rP{MSvN n%)NeVh:B~"%E?q4 G)`I:\R7
yPsN2/:=PQ62A`'F$k
*PSNh>
&J3gv!i5+G~>~C-'OX2RYC^j4(+YCu!'(igx!'QfGNEi<J3e@0`aaKFY8KMMb`m6JY)0(Ei1
Lh_GPMcF$\'aPcmf&|d(=Hz
]^r2{GUm0<Q7xcT^zEoH6G[z~8.Ia\/k(_ 7XNm
wd%r!0es;?QWsQG\v
< nf!n65Esttb(0ImfQHdjy;~itxg.G{
Yx
\wj8S
Nvzlu2LJ/yw.;M)=*5#ES-}t,17Z|N?9'D[EQyEE85e<^(SN:tQ
/# /JJeaG_"9#(30sQPn`QE <"O |,U=(w}wp1QpjQ'rB)VXYS;p%PF^Pbo1eEq8["yNc#?@=Q[(q@gve<_~yd-o
ahL[5o]N$m
BhOq~!
i1[8{b
#iC"
8(8{7j>l@(_`?N 9kZ?uP+5sVl3v0+`5m=D'N;_nalyjz~tM.==ny
!.#,!U6,kI:y_5msv[?>FP~sm=)'@tW~vtaurGgttp03u;A1QeFQcN<F<8AH;>}3S
0*]Ai,p|<4(+7 #y{nQQs<P;3i,(V3J!-\1L ^
wRSL^r1(0%
ySf$X@bubBcSSx~*w!F?<AH?@!]3ZxR`>;2j+O)La NS6)'VP^J=~gt=/i(/3%%K~|(Q"#/q8lsQ6hE:2OU_~
8#`m'<'nE:{'^0u)vnd.
9n75^X,MvNies|d
YMOWuCO,o~y~9+SI#yE'J\F/7_lTn'^1R~-70s#[3mKV3`kn[EvO=W0:z_i_[:!zC97:g^0:(Pyd;k Kz
{eir
23"u(kCHBbefG@IDAT%C FM^|*0)=VrVLG@Cz!ez#2eY8G?E%VZ4|78G}ak|U<iGH'uaT7lPM-Bk^:`N(_c-
|c0<i/<1PqqH++"?i>>|&Mq?G))"rzWt1PFGYr6/>kKTVG= /Km[GfdpQ. /~%sX#<5UH;8A^U'6|&HBqN

p[CYj'Zp^^-QwCNNk,]\3Evg7fL_J;p~/][`OE/ahDf*^n'-Y6k@:C#n\oOPbB}aMtgzuml]7H],WK/R^$E% V:' MI.:a
ty/N`tLf:AG^Aanz$BQU<#?/9/0$dq{ePQywBCY!i@D#Mxw%OPss}[J(Vc
iqmhw{:6(|oObv6X'w:
<47XpsSosVx'{`H?NV<qyU&iqpu@-*%^F}gSG`uI.1\v0y,teNe{dC*]u9dvV)-fXdD7}fd7X2mNe6
3$'d,GCm6x]xCTTfZx /#N 8{tByjC[YG`NG{qP<xtp(LrJ}@yTW!ISXFy>|W,<wo:P9s?e@{m{vNn!9/mUf7
6:ie0wTPK7lQ F5L[:k\Ic?[U[Ykigl@S"KCJ;L5sj]j%(-x 65mzfeeg)v,o7 lnfhX% @zV 8'pN
8wCjhadm*)=J++Lv{Dq=V3yS]8g.nym0+\>i.p<b@8hce7 F}SN 8'pN 8M hQstCXv;`FpUalj^6> MY:$VuL?pN
8'pNF'pN 8'pN xpN 8'pN 8'@#F:W%.pN 8'pN 8'7Zx;pN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN
8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN
8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 8/hQZ&KJHHP`_Z*q'pN@Bb*9!:K
1}$W&*Q-eTctj_I I._IpN $zYZ4=4'[)QPiIj*\TOhpN J-
T7U(+56^cMWyEtz7uEs=];2k%&$PLc;5C\|QmT\lC"oX9K6WJO:"'jL_Ri.=G#tm[[s0^]+16/ 8'<$dX`h=^
bOZ~jHZzPo?GP%_KHg>zkz-[Vh5?lF3&]J4pN@6wEl=v4c75zy\Nzt4k{]g;ez.pN 8@%e{z-zm$hz{juzulH.~Ai i
$$$FcYY)3H+!,2S|
q-F
--O^c3`=[|%&%G^YJj1*-vCAT({iIqEZ\o[HrNdNhX~#]~Uqk[Yby?E8OMHf-
rF(W{oy&[*f48Z&gPZRNumUV],Q4xU*;'@y=4+2XC^5=o
KV6unpN uuj]5yZrSf4G?n6mW:fpn([WgjA*JP}<
 XV_jI*Mn}rx&j"mMi6-l)6wjv}wSO9^YR~\5k(XM]
}>9WJw<(/D9kSn=FWWM9jLA6C@y>xEC7I,7kZ\3>9+Ktqxl5MsJ_2M0V)vd<%{qPe][1t"a:Z`;45J7C`Bq}U5G.
ZA
iH+a6Et$nsT/H-0 GhbZ}lyewI#GMvnKutzNVP/[C^]hu3omR}4y&|4K[?Y;7i%*OP!G!},O#U33N
8'P;*/j2m[XS{DzQ7kP53PlJKMl5i]aIHE?Y;#E@Q
(_Og(1HY@G&VmjKB1P);WoQZd[LrzUd
So)y; Q,3?{Ui
eJet?-0RYdoC^ZwhBSbSyLR
4@)3AG~<O{UV
5S#OMk`J--ew 5HOg/W4B`KTP\n~kzG3T` %h[ck$)U[[beS5g?k~<Ko=/8ccj^2V{YziJHf>y5lYP/>N%*D}l
</iUe6-[RVZlTLTSO<p^`:O+C/3MY}Dg.QDi9qr4U|Z]S6acJ-
k?TAiz?jZ4ozI4]/=w')Tc}pUyPnHzLc&#\6qg?,2d,^^S[6
oyK2s=wWN}bh]]Y:))IrU>kEtS>a7oA`zY:fyG`kpN D^l??-
OkKtCTncRvMMuX#VfjwdzB)g}CFl!WV(L+H2b.&{xh@\qhgf'|jn#Go&M~]N>4l+JdWDvEqLVSYVK2jfE

Q[)k},y
ToIS5G;3*KLSiZe)hkhUQKW{6ef96WML%d&]rMyyT#zK^1oR}giTMtD.$W?:kC4WhJ-
:r:cD?[cVWkt?j_ct43xJKO3yzof"yQwQU)7=E%i<bRP|[kfE{%J+[Wc-f7'=5k1;ECw`0@j6mS/O
4gE-i#S.Fs5`0CNY~~7urOl'g.\+(X8wfc|:IVO5PK:R{I(+[bD9B<i7K'/o$i6}l#pN 8C#Z-
l{EA@a6\#z\k={hslUIWwTqaASBHkI=7W]#gpBq64Co=#K#EzobfRg~]A_hInnB5VC:-
]FJ>VmViQ{odYyZE8D}Rh0e=7iggdEmM&h98;]V$]27Xe)jGjtcd}dCipziZTs,3AolSQfRS7K')S8Z{LgUd:?76-
%SwwCzA}TF(~TeTPU'./vO=4Q~~AajT3'[}La
P[c6(HN 8'Pjha5*Sl;DsjZF~a$[Wp,7E2CWxkLEo
~)7~9:'%9M>w
}ziH[7'Wfl`ZPYc>[Q[4jShy7nTbFFq]VL-R7W
eu>tOo4cFzzzLs|Mg=r^K;S5k7m5t9
FMb"pN ${'kMxorhH#gob+xBr{oy^_Bj}q rZnvbs0
}jMTX3A'X"<P^{h&|0Q}[#@{)6!#2:3:>DGc 9e\J{/9\H6b~2S(1;d~.f)9p<>:MleX=67HOyKxl.ED
[TVBciV(PvSci,C3"4t},gaZj@f`az
eSQNS*dmZ*TR;NS-Yi hd'49X>kTC-W&sN 8$PF5^)t18r}j:{MFxPbe#/]?FJt<[?_D3

e)GO3CTmjY5|}5Sqa+/s4IRr_:J[S~.I(4i5OMv`L]z)4ZXo|8E%V)NPzxUyW6.7u?ao.7LpN 8&L2cIiL.MUQ[3`#
f@D,D;[` +E;@PuAi*<MMoA^|$W8 QVb^-;:Sjj`0%?%#W/L
0 bd"v.}HW_:ukMHa)(zg*/O*klQE lDy"$o>ZjOTEMcV8Yy&_X+T!EHR5o
slW>nh@mrhO6xr&V7Td=Fq6rX0V%N 8'PCtlSFeamK}:\YwDzbq_'U*fQxAQ6ES42R5s2}lo\o<4i]XSW-
3kI|Gi^,T5{cvf`s3hY[5?Q<![^u:SAKFhy]:Gum_?{N+j9yh4g>]|uSk[e7ZT"pN
"xi4h@WM&Nk#xb$&PT>26g^y[l{BPftKEsGmZB3S;HkfMkUc5
fsfyV3x$/x&SEW=iD4/v
eI7,u;ux6CLMKE7vd<X *JZ=f4[Rno<PM;Sj`bl~yCy)h
ds:?;?cZrz:bi`zzn{ESi
5}x3yLp:z5\XSi'w+ZP5(8'j@{Zl`3HBQ:`eWfz=6
W%JXt-SIIJL'Lm[_*V
[PBVMU='
%YW|Nsh]m6dq(FyGo|dGM54]c5>vVVij@:=kX5G7<~.i%Vmuc|BHOtmU4>O 8'`LFMolmy-6x9%ElQEE&6
i}F!Wv#aS;~d`uc.y }^fNc eb\]r+wQ5'~QylmSVh]:oKACQ.j]OlKms#O]EClS=k LaG,.r//OC,Z
>Mm/]cp[fD}.<o^|ma})Ia{N29va!,{OQj3LD=v=(QgPWss%UfzKn^;mF{iU9+!TC~=}:-XP Wa6(8'j@-
&;HClkfXcXb2lOo\/$"2L>WhF-
g{DjL]]{;[pQvf&h//?~|n_VzemLPQvs4D[;}e.|T2Vim;|={i<}:,x'gPnr(TVzZ$J&znM?YpM-
xQyBN 8'v(33;)&g67s]ak.l,n dwmZeYmKe-4s)8)11LoE?eeg('}m]OQf7QFr>{bHj/(,)g_R%c'~UuQh<4@i)Gg
V:c^82=V\yT[Q$f&
dSOLFui KlpmJi]fX?4ohX8`y~paZ1/bfu(H,EEKM0C
d,I93PmmJs~j;<pl}1J]n'y.Q}"Zt?v$%Qyz
=N'hogEhbieSl~`ko$;&K][#6bVrjcniW5npN ,cg."#OFOiTmC=G=0*AdQxG#]lco"PPn
Oafn9^xe]*jZh/hok)~c"^3Z,XF=
"3udT{p"=~PLkX_E;GC5k:qPW{#~l"P5<jag'pC4$37nnJK,'$iVhn,u!ZV,k!+;'72z0$=Q+6nS7U)t[HmiCd*q'{Cr
`
o-wNrV{V3"ohCBYeeFFi8f%1*,<2T@V3D<,NZ?[m!7y"-S+-JcX9:
L="sXU"U?bQc1^%n48^lSI`w'5kylXWr~rx0Sefx^WWznt}h;ub.=iS4zzt(
hu;u)!bj[sz5:Nf1Gf`C2,:,MK5zu&eO_3Bd<Di[X/E2|%/2>dMW9S.o<RYbpN
{Wx>VcEysxQ:WY:{;81*2b%%{:WZF343^TYHKd1R*mUBypN 8/@-
Lrul"]gfk,D~Zmk?\v+*/|i{YF{v7^O*Ii=1c7]Iw+syQo{T\>"J*RY!<qomUj`gq,fE'(Yz?6BRjSKG_3uiz[0WW>Hc3
mFjy{Q8N=ct/6}W;Gdb{W5>4 8'pBHfmc$<Q'E O/9OymNp_5-jrd2CKmv=1lZV wxd0BSG `_k[ah
;dkk4O/4GZoIt4:2YDsk5Zy;;A$EL;wlQCmibA7efa&LZ.otqaB>]!+Mcg/:6y-
Acj4(.vK@Y],_`m|erN 8'"l[K_ps57ojh3\4O{.=[=sNP0I9CXe%"OZS3{x qp)@#(at<{>n!j]dc#
#2E/9`(rxN 8'pN |m -j|zj,qozwi:yvoOh*{{
H8U+>|!,v'qsXVJU4!?pN 8'pNB!A/<K/Om3lTsH7Jg;:QRkHO 8'pN 8'"PFXx
bkfl(At@=8'pN 8'pN @-b3)4pN 8'pN 8"pN 8'pN =n{u9'pN 8'pFpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN
8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN
8:I K*!LxpN 8'pN 88c(-J*JJ$$)~dfDvRMEf8{99X 8'pN 8'!PFD3@hB[4VJJn]1rzkny],\^HVJJL;5v%uN 8'pN 8'5k()g-
zQfee63w ^%JcU:5*BIfZ;.pN 8'pN a[V`Stm`A2U,b##drxj0)#A`R*yaEE%f1i-5(OR4EfG
9'pN 8'>8yv;^sm[uSx|G'u]uu\nj-Q-!Do[SPzj)8Pf-MhtI#QOPepN 8'pN a5nx#l-
?<ggf[iTh^
}:Vv-t>"NlphSCjx`^
u5y-lI|203ujyfarwX,46REs53ugz-Ht_;9vN 8'pN 8'5&PF37cCyIdeT"Mk%qD}jN_R%Gv~!@A;T[w7_>6Q
]XyF^mdejpZnfJvlTQ\md041%f(Qf'}ba}:WZ~f,CSqnV-kn]Yw|Xr ,<pN 8'pN 8#5k0E{&{8(dfj)8V}A+ RKJ7Rvh<)e
&j}]
KZd(^?S=a9bOx_/
P;(/N.im+m =BH'f/r1$MZJ|>~wof{o4:| lGd"*-V#b-:9I1iUIBQa#i|82pqHC
%aT"]vQX. 0&`L 0&+L (uy -aj(`[`<'#H/{!B!S

a!(2T.iEXXXRyW<m>]IJ)>\0zo`ZNmcLz[^us9m%!1U5>50a0:4
kRNV1=Pve&+`L 0&`L oEH=!#s{O[o1mEC>!<6"kuJ

rfD?|y[O7<IY[=q;FrU.X)p8&FXVF4oq^eNK)pq,K>w
33
Zd2L 0&`L 0&(ZLN4bQ!_kvg8a3
O.#=wXJIiHc"N'$,Yr$ SGN=vBD
>a[!LF,aiix$6U =6SQS?)UE>,8/gL 0&`L 0&(ZZv
&.?#za3
)iB#ao?[EVgO
!IE-PCK:H"gI
iV!#b#TH(B!dQs 0&`L 0&Lh/<G2}\aL%J8Mp.83V@" 2K8U,ArB
!,'S4!s&T3P~xzT*]8IOFF\ }|8^w]bNol}?{ 4+KcIfFB3-x^`L 0& vY'0Y`L 3^(ZON#L[4-6aXw:-
VxTkCdi4!Dq=pU::H2cC?~]vsDU[PCT7McB?wArch
>t
EHQiF7\SCF_1jHHH-4g;Hb.H-1a`L |1B_LrQdP!iQ~U 4Jp,<5@%7FZ+JYAd*Sh/fC ;i48BADb
%p0SVIPkP.VOHB /JC28#HR{-DBtJM^M-PZE.ip~801)igd"S\'KC'/z.{!]<B`M@;U$L 0&cM|<JNc.T*`kk+M&)-
vr`LxLL_mkWXX|}}QBF/&-V6#!$%&-HJ$0R[$0> Jd
t2#B@#Gdoqf4aBgJ#QefA?bWbR'H 0&@q'@IDAT(899[1B"**JVI$&`E@9!!!RcERA
ZE (":*i]{ hyx"_;H~!">b' ~dbXdb/`L 0&`oo/MEqie 0&@qvvv0J(dY@ZQQ| BL8`L& G]f-VM7w 0&dmbm,Bxi`L
0@*/`L { B35/&UT?b"g`!{3Vp`LB"C(~X
n `L 0&`Lpc`L 0&`L 0b&Bb3&`L 0&`#BqTL 0&`L 0&@1`E1`L 0&`L 0`Eq*&`L 0&`L sL 0&`L 0&@p8`L 0&`L
0&PXhQ9{&`L 0&`LpXhQ8n 0&`L 0&(f,(f=`L 0&`L 0&P8,(7N`L 0&`L 3e13&@>
z@/D|)B8`L :t:-232&^FBBi2* $#>>)ySCC"C! Y+qlX^v8x+0&@&-"p/id$JKG@Os)y3;~Bn fW8}:tmwo| Gj>
au,0b<"S+aQ:%&O\F-
^F
Taep%Tj.+fh(:uh+|ot3:Tdr%QfL{SmS_'xX:$XtH#Em`&ABij%`w0qT7)2q/d"hAC/+L&vAaGs*t]ML^a[K.L?~W`Q
2jiEp_R82(kZD3&4zZxFDD4MrvJ.BLL g3N~3bA`p.0}p`/D 5 fM%[")]vY=qY<|hyzg+_/UY!|xB0`,aKiXJV#{sl
M_4+^24+%F'F-{<d<Z5[@Oa?kW
+0$x2
lE&Y&@ R9G
KB!G4$ii*) h
88$-91*

Ke4HL+<_M`ozZxGOMIu@M
dJ-ZjsmPDFOMab'RC {jR SJt)-|VF d8Mp2q3(m!hL[V\=HH%P|jN=5EVjq*q/M>BB>Jdpp&..mC*9
)tuNw=aL^VVFU!#3p8}5;7 E|\|cjfNNL@ZK]64iHXD7,IC@:/1&4<k|4.UnV#2UZ4<
MkW ?i,b<6Y_cgcISOK0
S
iSqF4uBFt)Qp]V4k4^4o{Pn&c=-c446)9TbZRRaZiIHIUS*V/8>`/A g DNSj>{4ssF}*4;oM3O| 0&HOE}{tEfm0W&
:uVh w<HI.!ozB$/{wCfMy"mo~{O|ScYZFazh3VH
Zl-Zk<nF|oBum0|#WDe4k"T gmkBV;h<58~!I;~q^=;Ev!]K}Zc? "M6s.8GePv-_9SNun.kpV<+1h9Zlm:":16q|2'zvjf-
Zm;pN\\:<Iq_{]GgM2S{`>~.;1&u
:N%X&dPA){3$D=g`h_ODS/IJ?3'1LC|9do>DqF0nYD
+nsGLO7~%]@83'Qn>C
&V/A`!b;EU!w9!|yL74PSCuGKG1R{|w;}J$D<+izL=uBs171
L}
cQz9d2Bi!(:7@Cq;<

i|."QJlmi'c[{Y'DoVwChLu,tp`D gINg`kcS3)E&^U7*6O|warvEvmszm6$51`bj5V3dn 6vDI9.>Lx4K 4Frj*Hj
Yyix%V[w>lyVh2=#78{QkKTI~1z0\|UMs<}|Nl+T<

E,T
CTV16:&a8}l-o
-^#VBBBcTfh?:WB&5.@Tk
J~q}s55ebMZ2*6C[wg=s
xb2 4
{LGXuGW<yx1@@h&zvY#:vSF@#{#3&^+p_-'YP9cPLq$rch-*p8}8fK`NX:}\|z9T[Bc{e
\:up*92yyRU;Ya~pQ=Vv`X{:]BY#<
kQm\nEoP]@wL{0VfdWSiA_mWZwgI]*uA/^FZ?A&o[.[TFA)i4:nH'?&.RuePYl,\:iFEXlP#?r3on-
k!\G\,]NL(:xJTmqu5I2w]$ q$>fL (|AO/n
boQ#BSqV?M>1Vo')~rJ>M:7ATG<g[1qX)wmxsRX_ (iV[WKcF[EW rNT-]j`:ow%EDsi-
m`8r"k7VRc31Q?jZ+vKTWq^Aib'Iadc'X;9GYei,j4jEKqHITJ3R0`Xl6r&Sbh$sO?~.futo^*
sGbJ<9CGpPC~ u'`_
/ [}w^WwFk1cL <@m1O],_ 8h4FjM}#1z\LBBS1^pA?&!4i\
oO[G`s:!t?szVQ|3?J93?E3R9r86o<Gh#YmJ;_p%MH$GK}NmXPbH%s.`o_ C0}xNMP:A_!'/n$]&DZn?{Ne -
TO`iq"76;Cj3jM%LDsP!H(NP1wp<570`S%8u$PdFL lx7; #V`pRo(6rVL 0%@}|M1g$F~z$=pXXg-
PT9G'{5781.NRY5\4<'Q{7L@N=^"9IGh=JxFBEkwn=%Y
zBHM
lIuK'3@JWv*
0h@/*#>s-DI Oq77hZ#B|HchF#l}m+'& -kbeW9CX<wU~; x NTR04E3nHzmJsiKZ{2{&)m89GJPk|90& //
6q,[? }cKk CWl5}RtNERFY#:%{U^]YI>Ae E]rhMrfgIYn,d8|p6>sHCL5s%aDNM^Ot CvPed$$
adoKOaUfN/FvyKdqT d@xD 6Ki5
MIH"Pz;tjOt#
I2+7iAfi|fDvI$CC"R7@(:1LY!:<pzbhj)xEW7q @$6U"9<`K]hylZ!GJe(rH]:!
(C9:kH/sK kVc-rI>I4!01LE
dpi&AF,@8c(}iK)g'b;IB{C'!G^=Q--L1WFJl]+-<#:Ym^>X=|R G(udb~|cR43$/%N}@iIb=6Vea[CI56 >?}a>9b,
rjj'M#uO>$AT %#cL 4MY;o%O[a&<iXt`1&T^wc-8v:rr*9VblT+-uG`!ddyr{QI3 r}S~RH.
gO[0{H\M'm
yiZ H[OJwvi_;~s##5jVpQ~JL0Cjl k=qSiWHJ 4Dg#Z-7|qM@9*G!m3H`[q
kl4!j2w|)&Pr}Au"^RgwI\lo>$X3sukDWsxmX~c+I]'8AH.QF+}HTuy^N~q[s52&nbg
YRdJo92K&Mh74)LM+F%2gr#;{3 X=
AsriLs@tQXvHUG*K;U%) fh:F_PZ>p>a-p
+t_.v++R)/Nn`ZP?)/='vnsW$#z,iw?IU`Y&s%9W*(.6wc'7-;b@=}+U
:8LfJ[hL 0""PrM\3L~7e"E0gWm7{n44Ap|l<3rFjM5\ $poMvnX*teZ\K6j
*Zbq|2>JV!SJ[jPtaN#E
|"ChJh4q" H;$r 89<-o}]f0.a=zyc[+0/=dF+Q?!wNbC~lQN\(T (DOm3wHw(k66u?l=Gp/SaHglJeQCkU[i?a,
G0n'Uz^,~+AU`0E
!g9"54^^ 0S9Z8<"[9h45f1mn?F3Od5-`Gk
9PB.K~2u7Vbd7'QOPdje\s"%Mj|2Sm:EKpi>[6^SJ %PmVF 6_A."V{f}e<K0"2lGRtxfr-H<ag}%*IN!Nd140>Cw;-
5pYYCV||O`L"Z!go!Ihu`mK?(AB7scysb##'&|?>EFTn>9S\mB(AT=JWx,g'
0XsZe \5MaZ$ #R$:4gY!c*pZLa2%?@>:sAPnJ_?M-q/$Z5CFr+w28 >a?6o[R4d23 \AKJ+yij:q "i4GID4N?KUdQr
t,Tl>LI0t>Gwa+ENu4VLlQ21]#XPp!YI@-]HM _|{-z:[t)nFZa$Q
/c-H[MXDBr:xrFe?AG:*p+aJWa_hXZ1*
AB"Af5;?)dHZm
zm3
&-jg?pQ.5l#4&lc

{2w&%A
mJ<VvaI qWR<}!{XT~r.,ZYWqm!/w 16/>;dZo
stbZ
L;=qaFz)(2#_ 4^I&4Ydh&d*Dg%F=miJ48"?rv"P:I<z(S"<.AA2:>}A!@k/B:8)WT.7 %
9PShAn&RZhZi~+.%3ir3URN ?W#%CA2ez[/+,"KD`F&"ZCooLj!shGZ Pr<j&z
}R/4#SEkC[Bb=(NEgNb|6cASt53M/#1F{m[U$vI},yud":.mfh*|$YKELh_o@JCa%Ke(y0&P.x}m||t?M{7CZb4]hbfI
y<s[tmPX1GH=ceL`hT{wALRu!5'` >7p9(-l-L,Qt4 [ZyiM%GF]^M<~5K }
:<mp>DJRKDC3x^:7z
DMC"'OpS[L&9&45Z5cahSo"MY4.c?w<dqzn5+9N{'&Mt
U<o-< OaT7Qw0I_KK7
w=0kHg]u_#+"Cj}kT
2&^mMV*TF>yU<sLDQn
%!dBGND2~PTQ`!U:-P3um:.Y1dB1ehhtLcuGV`T%aU6TrO8x
@N]U:{CgH;F$
p7oY6!aW!o99_+jZFwKD^GM79qbW9b,D%Q>S'#Hkt%na!Xd2*Uh<}^(Q1`hfxsLWzE\3st$E(QS0~R*<d0+[\~9
&\h!L@KdIiX9~
K`A61iU\~
_Fa\jxMJAK~JSAt`bZ,O;TA{"o@tl5ikatV 0FaZg,\'NP]
]gaN4,'?mXj59"3CZmSYQ
{&vw>%nTD[|w\k|;=TO%Sl/FAa_|;#2mBt&4w1epn\;.W9Jwwf)kl]$[ERYKe,Z?';}=J*mo_?;|f8"$n9
eBD ++'B|
Qs#{a"P ,Y_oK
T%<;7cqmw|6;C2qgE[i,~&y7z
PBI9>4}I3UO=5Fl0(Ysm!
WU+(b_&@ pAmZq7D>1a4r7#~a8wz[0K
hl:y G/a9?4[qr2w 24rS{HA=e:t$dSI3&9d"
+rsf G^};7p5CNM
1G$M??7.AgL7=_H0b%k6c8GuwFZu"T}~XUXy&|.l4,)Mm3]Ca2,O#{Xy?=%MmaX0wcmE'A%V|QiKv/g?
ps/AP,7P:3[;SZ?xQ7obc0y F}_m\fV(L\f'$I0-
3/u:v3{`Xq/Kg}7|4(w9,q%q*Gvac/jZQiQwyoQ7>c/XC@QQQ(SuKdQ
#gZ_5PM|i+oTW\`o3L[i$n]%\
lWh8/y*B:yOHmhNKA9Q_kI0H
?v ~F%Ceeu~+s~/'gOrDDx(d%\J*[P`7UCS]\vR@'
+:"^2gvgUhmK(")8^YRI~)Bu5
mP[p^1AG*=Tm35= [_D0U.V
/Ntl2r/?&T2 _prrMv
*FMR8TPzFE;hh
U.vs1W'!0(dcCh`RI;n.RIm`miUN&yd~gFim-3Pdo8S^R^f}"$0i9#>2

*
U*"Eu2\KRGr>//"}<hw1[!62&F|D~i Mc%i$"619|MG1nLGxp ibF'gbl*%4g
mF~NlaN!*,;Kb1%iFg_4?YIiUroHIIAxx8O;eH___
e99'T/]b&Ff2+Lto UcaKK99UG.T;>b;HHAQD;n=uu4Wt17<i^w&%LVK5^L-
\Q2{^pq+,w0G9%dy'L&ePrYd}7q@l8o-]C:"W<<G\u1B de
+J84!J{}VRP'72X9!|ZZ4}^$X9}80&^^L3XfbrhVr21)zv2ZPQo^y2N(]/R*]&K)l(Q,r$Gmnk1
S3X^>T;ih92n$<}\]u.Ug/"q0I p`EEhbCg1SKIB<!aI&Mgq8JU_R:I`!eR#$o)BMhFz
6[7oyR8G)X4|Zm`l+1ju}uh= 0&G "= "lG#OCYlW\#&PVhA
by>}5K6GI+DlMR4(imEm3S?Kj[5AuoRkbIP!RHc8`FBq);'Gr(+m\:3&*PR?!I{*1ZXaSC-L

0&%"csI6:9\M&PtgM/#g4.11Y!H<=43-MK_Js6=%YE:K{[hKB*;w$hS,!gSlF=ka,(]2'kx} bUHji6OU|ST\ui
0&@mqadZ\NFq@!#iZTWN%ET2pya f,NCh/%4O!v5SA'ZUL#=:Q(p aGlr9E&@wlJ[SXHM
v`L#`L_@uhv5+c]/&F wrGFLK tMH6+8&8xt]VQ<G{60L 0&`L 0&""p]GdrmR!u
OPXn,aB\Mcac&`L 0&`L 3Bca&4)e$LN:s^i 0&`L 0&7gn aJ/K!%>`L 0&`L 7sBWxIp+`L 0&`L 0|`L 0&`L 0&`+wKBL
0&`L 0& B~`L 0&`L 0W-^bL 0&`L 0&Xh`L 0&`L 0&J`+y[RL j$%@aVTn!:h43%CeA-
gUb;g4h=zh]i^c:j)6h~|(TD,'sLo$KOJ}tE.=0{He]zm_O]

o1>'dW_2-.oMQR[\:~,yr?fNqV{0u%J`/J@ ?>{>4q|v\E&ba^Fo`H'E;1}H)h8w~l.-
EK/%}<ErYi?E\u|x5td|;8~Xh_=0{hy"UN_|@e
/%d.0<W%bqc_R{eLQn]$TRo/"%j+jML2
J
;wR<'>}_~X)yZ|(<]K$Gwa6mNI7!-c4J-sHN(tX<|NwD0vAO%LM'>P%Iub:% Ojln>U,m;2M
>|iyP _<M0/'Tn)o&A*U:f&2m21IhrTSB;]J%e(r
"wTMBt&Ps
tCIeZ }LaBK2"SSQg2{ #5QRT*
O@Q&&N*LV
G]pTTzz(8%d*DF&t<Rz%7Wh5UNY!Xp
K&2s"=PYPudR[<](bn+Eu3HO<EN?_wtHNLF1c-!=&0+O"[+-abwLoB#T
&#:,KL~}~3k"iqCAa)_]z`Kr)j59.b\W
*/yaMwJ Z/4U-gM#KmE~A{hR0Oc5=xqP0G&z9
z{Au)^<oJS]q5f%p?k!jQ,/LYg2:zo0^xep@inh#[}Vb%~M~FWHOaV9ttqgG&4H&/[DXyaib'zG>~:\?=0{$Sg=EjZ
Uq'^%}p'SFfs>SF6dfTxno!&TM !Nu.Cv}l7N\]c.HxUWC(-
6NM+J;6owsk'2={OUkMz4IIyKWCNZ8p?N[0g<wcxJm/U
#&MB*1A>u'?!48IS&e)ju^Uu;OY`H+q:%Cl,
MqbaaC'[Yygx6]PCEM3XI<G+]wg:1]!7.`s[(kogcqC{Kuj,vY-trzjb9dpXb
ECC,$S@IDATbIFM1exT)DqtplDD|=Y-N
[z
+p{9Sx5ks^oJO5WDRE\viZ:QY3av<@s[W>g
ptR;cbH@.0~h/XgM|.UJxvb'7e[`AfNk^5KbDL0`(2kb}+n(Yma=LU1Xi(:&A9p.F.]#D3`gGGu&`?`^
]n@~cA)]EFM-d1,]Qw&QU]iQvPl M7p8N-jHBak%J|NXEid=zQ P%&Nj%
c'4}3<=] WXd)nSM-s"k|44DV!*`.R?oFY])C;q;o9^z|?:PgM8y"1=r!1 df1~Z#Z8FTb
,PIKL?6&rD>uQL{=J iB,_`n `j6kq; &<{gw^w0P&{bA8Ol;}V_]7>D~d?&6lq>gPfB#VQXi7#`[
&O&KC\f}x4ht]mW;4=7Jz{D()qc&a2VGjKlUq%xKrIeut8K&OwCL6PCg{v/mdpAi'AM0"[6{7
)p)[MFlC}o71yG$4
V0zxkh\=~1N0[sfvV 0J5 6@lX
go=&9\]KeV;}G2-'E%C -
VnKs0Bob>R{N,Z#NE8E\&a`Z^Ppz6lT\*` h]/:xNC vT7Pz^YcI~_}4:WTp
^+~U`L*De*VE:uaiaj'zl5pmV/^b%HV|9>X 5Z P$>1Sfzq\RCMWN;`Z@ed|G4okL3oGmN69]g0dhtiR|97$y2M
nUbaLz?"6w?i6i"Qu>Z0#_>5kwAJAh,shEt?PnsSHPk
DV>e7#Ksv@$li9lW
P'WEQvAj1j0hT?ZU<H]ESe?O{_[g1a[3qq7u.<NySl_R`XAf,
v^=h~5DeAVLi(16fL+f=z<;)uOJ~XwM;}d5[4y$>['eL&J91nFcqh]wfnt6ul"u|))1|
H`AC (EJ@q+Z\[qM sw7W<w3ggnJA^}5>N1~
]0}liFaIl.^={dM3t
*Yc9=J\Q&H%aCi4%@.eUWEJi7+7][_#
f8$TFGge~^slM{ap93!"s};+R@<'"#9)IIHU)]\ a4,5
g|QAx&]9 S&OE~3%s(3w1s0v?f$nPYoac\uVf.>OIu$q<EQ1t|1n,Nz%|ykE}PvVL^?~/[z4AZ[Ld(BCQ"<>[~p
7KOWi6~9^E@OORYeFR

=Et!y14KG`9_M50C`H!kR//`a\)twT
1i|5Cq3)]&>RQ/c4>ABP+N]?EOyx?xQhn%h%`P
Ud.Jh*7oGQ07ThZdkg Z}f3C&^0/b70P4/W+h78Wg+=x|h,xu:}+/[9,b_aQVSg1i,IC9BabKu,|@? S(L\1JSc*]c#n};~
<P<}:.WS1pzjFPuE4xxRY;a0PCHS4EB'%!C]+14wPd&hS4CuqAvU|0Ekw[tkw+4\<M98x*Z`8s[^^1!#Gez
G),f^"#hY@/0/qb`g<aFN_Sz=la8~3,9v:NjFp\tP,;|4D
uy^
KR_/ljKkrzk{`G,l$#m
H_EF=<q#^Je4QkZ/AOi\4<jPh=C]uj(pjCHm:pZH
1'LK+=7a`S6A)<^
WhK
{0n
y|>!9>Tv&>VZBF/E5&(Uk!vWDgjn86k& b^g/B|U|h75S]J7cE~`WoV5HjFG-9tQ0N&pG]9[
@Vt#8yd#Ce\z.MDYS:n/B"8D7m|6gm]FUqxb_x[]LL@
Kap/iDcbE-:)"hTqWsQ<[8pd-
[vm4~Ub<TO|csg0;qvJmLMbV}l9[u6n[7yE?vfSzYuDo\>$XXsiQX0uTWx'xlomk;9il`G|t-`l~QzhDe!UK
hpk`8g MRibpWt1!#kQ$10,j4
vq3{vnnXt1Gk/wn'_@}GnBF ::y'vDa5^UMk<ymB)l0gruRhmrc0m3;9.cg?`R?+M+dF
w|]Ocw]iQf{V>=M>\Ce7{xPV3O^1KdfF4?LZ6Kx)]W|6szn9j(B%-t
[W^U5~)WV
Fv mXWzCS!\n$h Fn\F4n Umwt|Oqlg
=O!-Jb].W-.!^Qgf*u2@?!0:
d>\v:W%Xt61WqVn*HQtH,]hmLg1N=*MHW&~IWAS]Yat$\4l6FU*[r(9|&eiQ2:nJ(Wk{2!t&
[b%($} s1{c3qA b1*b3FASO
#uP4$ o<zfU:R}S]S\QC[:WlZnu>`hq~&6Y`vI6Xh&h(so3UU2g]kY}kx9eVm{v{Xy2vvTBL)wH,6kh[J5&m2~?
'1<xC:D.-<jBI6;@:%y?r#1-9IHR:QYaQ ,>$fa8GJvGO89st>-4N,nA+
oSQG>a(:^(tN~
qU,)aw!0~5@\`
pket75qyI9fU"B69/ `Kp7gMQcF
MnE%mnrn3Yd4R'1zeKF'=.\y!p,>%\p]2{~?-ZUqA 1X>KA_M7d]TJ)CZzuc\.jtBPT}6.O"sjY%|]+S,x5AJ!hz"7]Df
 D'dR/IH[g),juKU
hD+3'"NCxFI!|kWhj7X(s5nEOY(j6:N
Gg"(y^=Zp-<])IbP<fKr}493x0\
sBK=hvj_gF5ka #<:
?{Tw."##5C%P'xbJw%$:H

M-'QgM.~*ai<\<&xnK%l@p76}
ZUAxfu).BgN
6O1.Qqu$1&]XB~K2Zg?!GhVnf[h\TL
Es:wH'xu6u!f,C[AZf%o\5J,A
U{N <Z67[<a261Bry%uLL8F%ULi8q8h)AmGifsAQN4A5O)Z8=j>Oer(9H\$HA(]*-
2cwq1CvR<b]$^{|y+,_=r&%!a:9r@+e`3KsE]{C@]k18~/x,2/W91aeKa<$"<jK/)*Ex4Z=dImi1tFjsaHa!PTy5oTh-
c)~=PT) SeUml=u{hsXrD0FVU*::I$g%ic?Hw,vVS/J3'(=WmCI'Nfml[5$-4cT6
(IV"WZTtMTws:,`lCJV$(B_~:^'eQ/"mplVQU6c`

47ZV-J0^ 1g*~8z-[yjNePGOVA4`+/qc7&3ikmz}2CO2M`7#i$ighy4
JL8`g?F]JQ[b EYO\|\=q^1X>c4/laUBq:0;|(u
mY+
!* @))pKV$k0oQF#4{DSU

_LU%('b6R{j9$A
Rw44@m($[GKX/W+hQrSS
+5YjK.qRN=yS%GK6b
!(Onow05:::[jRBi=G+%vQ:t++WlmlYmt~T85\#?%",yT_;DLQk}N<oC!P%yTthj6jI!cKc]BAG]hc,fO=]1.;s5}04YL
Mp+CcXYi@,1qDlG+pr++THK~kmHw>@*q[xZHx{ob
q4BHU8h(cuE&JK"~4 Pl.0Q3'"G/'+SV;>G:t5@+5|
1l
s9#}XQgE1j4.GkT$cx/,4p?3~&SUV;!FH];>+OTB)
uT{eO|"7/]k0
j AY,Tp#A{VdvKt:OOS>22d
a
O|6U&Fk$edAI#CQLY
>|SdzLk.E09i8q(*{9hznc,gNk-8gnjcv'
C1ogUehOl`i[HUS .z36D[7'(?\~u-;Y[HLMJC^~dckh9`.~-e; "
.DH5YW=jV;# Y4m=!'^i|Kh?Q(GW!aheE2.b(BXPcN=6`ux17)1c`>AFD T.!uCPdk9X
1FkH:YDqc{k*sO^D<z4pd&O-'zx..(kx>]}Y%.SPe?0"<BC/5AM_:_ PP<N]htE{Bf:#@QRE2q7xImtWM-
44.*iwbiN6&]&}f=c)\ .;H5]OvB^|~PXhC 0zfqgQ!;vTXg
,}$ qQ>Ud{m,_<i%I3v<@~cd,T@~?aA>z/YfR"NSv?8Nyr"C/$m~-x:[U<FC'p%!P-t9Mh<XphW<|DR`!KW-
KXbxNO8Km8R kQP-}f-T ABkOp6<
0g*/ \O4]9R^a've'</IO*R~YOvol]vr6Z()_=S.P>/_Zd.
C"7UKGxE?\L-
_lEJ8TmHJcYvPBcQTSqrli!}b,Q"~7]bcpv}>Ew=65#%_K7zUSjD{7l2A?K>}pa3B ,
F AAg[Z-|^xpKSX\:_a|vMocpu7VORcuEqq,]9Qm0l.UVj)Tu/ "#"ml?z /BfA1x6_d2EAoVQ-!|qy^~a
+ErRZ8u6o'pgSZmjfSV[m3Uh{ \<t!&iLKrQzsn=<p8/bG C4n?|l=@DX(n\<WK8X'2*Uzwf}GGWNb-
*igX/bchKr,Bw!
p+u}X<;UtD`8o/c<Z!!&B|# <~Y0QIA+U|{>z(F'S|HwDCP~{Tv7"4Wsy7ibJKb^g.>?
Z_<ybX<~FC||1cs9[6WE6w_6k]w[1}v)2@HPNrk ~c"?=_:w1qJ)FR/6wAF)XqeDk,?=CxXnzI0q,<f`\p*.\xhT[7|q_g
&5SxV3 5A-!O?Dh2z
_!8<
GGA`x4kB;y"Z%_y/8y3m<wF.(u Qi=Z>TKFt
.H[Ab?tHK[ear!Or*itt p1>_x>qIsUR?GVSB-D]Bb:6PbV}Xu\@C
=(%9I0b4<}{Y,_ar?"a9ww=E`F&|}q!&]fNN(V14>?
7*uwAYFhs=?(wsnV8#-T?5aI%A-[CD_Y8w.|X)E?Quy3qk18u)En]$!k {G3tZ
-Y5T}blW}fqO!d
)E WfFFRb[q6\h623uhOk+bhoS($(.
f-Qu22LWUc
]SO%]hbYZ;6t261jh
z"#&<6c\uMBGaqlhK[mYAgjZnWivf
I6=qi7!Z+xF]y+,Ipw+N<*.kcb`|t448kb$f~z%yoiU5VZt#jt
lgw[23l[~f#[*LM^_Ci`q'W(Pk00}Tx(
m;|` fly6*lVX(P2SBESHhFoQ*,|w#5VX40lhGFm{]#
Np'].iT8_syZ]=T9s%kj
NP|5Wos`_`rBa[O\3Vm[BYU7W+grn?Dy}A(s*vUfJz
L+{vo|*VgPf~5F3`[\@~34*4e)~BhGtt!19s
3:Q3Z*/0hl9xUF6UmtH@RR
p4c>O+x&2|Ium0cNW&9{yod*1ok 2yB6l&CV],
CK
(MA|B-Z\n,xH.UujfI&^1~R>n6E)6d>Fn&ShvsC9YJu(#!/

f'M43}E5-a1wZ|&41bQKC0ZVo\!MO&I5e/]9B(H=\KtSZT4g-cdx06,:h-Vp{>)2,gz{B`0Ps$&RSK}G
GuKzg6zqalya,Q
6kA<vHhS|!'CaQqf1gdkJqS*[~<!PSMToTm-mx:+L;l\5?`(X|@xDU|d!
2X{R)R| xu/F`~2|d:8jTT2-P93&gG/
f&2QRajc&&cr{<=,]q3e6 _KF_[PDK|&B|OZ3K2d5[HP#
VcT"JBE4<x@jBqD2zCg01p"9E7J ig$*UJ
+[f$i)QN X0UEuE2U(JimlfI|E>4})&7& Cmk6=)V)hkM@ddC
i^hMVRnL$We;tH[ZXHJ4PX*`0fzs2Q^Oo3~(zyR*)(R$&/TZIf2D
!xV(H3#32*st{KKK.EQ2hAG1mFD
$&&I>!9< ML!"s
}3S[fBp|6yFCRc"PJ#"@8F`1a8DD8LpQF&TuA=4"39g,O^L%X<",,2SI`lX8-c:eYnT[~GEf`
F}61v1gRGDF8*vDTxo%G%\<IK=0E$e]4Ncx>C|f(62O>Al,LCYT6*uE4PE|"@?y\|dmk/|\BBZ9jOWE6pX
*Z="S@@}h>n;9V=NNNd
Pb
INDI8#
SanN3A!ivS~F:2r-@R*w&J2aD&e$SA Id7<%+1}dj8uD`$s7a^8(316vk^,x/C~Y9<
\ICSUe@$5c#OV\ka88$uaN%%"1L->~Vjkp/JKbM"]D5nNML!=r.~A}ELI1_:>/"}
{];[mVl]sTS[ZP)M"es=M"a\i8:,}{"PHXrNH$sO6f=54@Tc?^y#9KCtAWH3K}>EQ_1hN!S
_WSO~&CKgC5G}%%ya(<'&#>'Wcc'FWY2b\=o<rTTW6"$`usbSXgvtby|P|EJ`MC.0$:Cx9|8^]ODI#@oV^EQHV
kEpQ/.:kR:]ObE(NXrn^UBeXX-DNQJ*2r?ewX7'EO7[@d
BQ1c)fmc
zdCE1eI]2vx Xp60w:t**pO~\~"~3 ZDs1wo4 C]>56_Hp\=EiS&>@lY;`pkt ([L_7LEcSH62,oa+c./S@@J^GOaU W,
*50~ry)nE2]B('2@.U_N2>D =W%-~78O`!6>Cpk9>}PcCybPf`c) -hz;GB!&EP)%]/SmTvg2Ed
7/nmo}UY,w_tdllA+5uB!"M`hr9

Zo_1yFv:[-dtUl~~~C.eKp-+"GFh/lR|E
89e/~G\BQUA< GCTGPX@Q.?
bXN ?,Y?G\20G#?vW=[Z*(>dm[]?po,`$Su7p&?:QjG!w_# (5,% f(";/gggtkU&Z%\6`Og@hl.:q)QUaEGn?FxB$}3`
H>)_2d
) S@L2d
) S_G?C"D= Ng!
'gUE

1FjNj`K]AE>HK
i`5
HTDe<:N*:51Rgf`:]QQe5`Ea-P[,#YF jX-^>O@GL2d
) S@L2d
[Q/Z
}6 _."EOs60g7~0{aP!kR,kBi{h2DX$!r,Mao HG!l)k\6$KT$e\}4
AXdna!*BDZh|1dX,1c&QX) S@L2d
)
S@L>BhT|s7kH&A+dsGcw#ekr/u7t;ZwL},?mmQw:N21n@F+531lr)l?G8@IDATmwi2Z7c?%p5*mblQ"f=1;/#FW
H)BL>xV6q>E:AX^
2d
) S@L2d
) SEq.xI%}SocKk< hksS,I"fN_suZH@)h_nB@>UV_2* fD{BVZ!}4t>:>
X5=Li1gWws]Q}B,=;FB,~[;9t~+cHY`G\FL2d
)P

L_b1O&\l2)\})otQn6*eg!V{oI)fi^IH29CjEqm!==
9:@UhVo##aSnzNHB ZIs.-CiXsR0jq3T*ky<7/?8SWZ]* S7"02$S@L2d
)@=r%>Fg3S;<y`"Myc8.$x3xGJ2/^FQ,?jiwoiE{Fm1/X
/m[L(r7R(z^mZtpF>(!XHfH`@zc8)ySnBLZHF'"*:[
JW=VDX3!@2r67`fp 96hm3sM2%GdKQ(_Mb^$Ee;}0oa,=<F)S@L2d
)P0t
2RkiIxL#ttr\6/+aNsDHIJ`m V#MMim !K:.Y L-P;!,/~#vC:><gT,h`+l_
VXMG$d(G\|43,,#-*aO oaB
TbN,{zy9%g]9s4zm:Pl(4~I)6>HVmJw- !c4.gb4.YIVdd 4B
S#0CTx:,>onrJX0+OA0 w<9L2d
) SO_PTS7;V.Da? -RLJ(]=@
-d4-*!B@aUBnB]>(,9u-0j1)@ESGWa$tQJc}o/F,
ZaY[VSQPelK(?E*+#`t9C<zqI.!q2,WLy$-
oUUQj9(r?e?Uwk/Gp3RY]X$S<bzMF*3T"[G_L]|E}gd(_M>gbU"YVp~MF<<)#Y! &UjW}9{
-]
4&)PEXX(N1.2V&zTElPO][b|P2nScxg_DK:I/!TqA4\ym0fqX&.vA/X
60S;TOI:)X04ON1/) S@L2d
9s0~{60A.Eap
w?HFw.6&Ve0pp4U.S7O}20|(r,F%*!9kNX8{ZdD`e%},Zt
cR-#.33`j+LZ-|Vf=|UaXq<J[`WJL9>iw&69A=]Kwq9^s:@xG8oS`fmq7 ,%eJnpZ
{F
6x)\}SwB`KE G{kX{WIK{ux;b/ew4NL7&~&Jb'-P_{"reKaw
cOpvRe1
#;+k7`A(ki>#8t
)D1|p_$e?Pl{EhviLA/q:}Bs4<4{j=a
Xo`1^N*##P>tWuG6.]O"0%{0p`y#^=n|q5jTMUqFi&~O]B[F[O{PVC<8WF1S#_[Q>&uHLe2
TXU<cX;;+/Mx%;q#$GY`Apf`Rdw}BB#OLEbY8U+xO .['|~iCXt@|r
N.C~==O/C*((:!) S@L2d
YshOl;W=O`B1G
Vlx`,Y%k6~}xM
q
;is`V!Y5p%|:fIs4 _xly}MJC
2gIP*VvfU~n #fp2!KrxaFnBSPbYL$
Bzw3p
$=
/nV{-Hy!Z{#A.+'/N`}s5.xVl &&\A`W$)&
oNgo0cIlQPvs!qcF#<jv^vuY
@]7} lUlgNQ~^]y>1.[9G^=E:Nx6K+<8{Py>6XZg%Dq?}dKaOp2mQ*U+k5g5rn<tf*d 37zJb8p9-ka_,LinXj+GleW
w*Pb3!7cfCP9f-9"?t`XqvW=[uX$-p QYXKWA.)+$\E4K<Wxz
T,k#APx8^?KjcZF?"rsQcc8M4t.,\R&|"{rBoquruprdDFC]0E6$S@L2d
)P 3;vjmcIC3JWMNc'tE&%N{)wMV@@tda-p:7Q[pVWxdM,]/" v7O#QB#OB1V:\hSlAUZD
GxIk<==+f#4m=LKpoch+,taB"\ZnW+:T.7F]}8`a06RK@ol
{
XQsx]v4K8"[
WHWp:k-36U
QyMI_ByH,&p+J,4Qe/%:Z:U
KCQ(1vw@k ,>@5fz->n?;~9~IAY
\;{tn76[FukCz#5%=@}UqoWw_*u*{@IkMIn 43L+;;)lIX0mt5aLk\3{{)i)#uo*)gV,-irV\?$,5)|

(
R{vE*i6|0Se2E^k
}
$I)ff
*KDRUJ+_6XTyE)4 ~
r~BKzl[%@KU=18[
Z2d
) S@L9)
K@24M1TLTXPnC9U+K}CXeD]YaZZr<2u3Z+,|~YmQBb`/>|HBNqVX4xE@'&hZB)%c#C!U7>HQ%]0%>F?!t)
!NN(R#YcVEG#+$"A[b@<
uubO4Yw>!m-LU!ERRC?zG{Z$'!J$\
BUj]+%&MHK{{$#;u`ko"P:%Lm;t_8YU,up?=t A3ad@H/k*Qak.h<x 8CW5u~ Z7]c>2W-s"Hm#s
8Ug\dB'\9ma#<8o@mXX|^1^_JIkJAH1|mqO$aJ1u2d
) S@L<ojIK<g/5ME\6q-|,%KVtySd1Z<:2NFjm2
czQd% N6 KBP`qYQ3L@)'uvV\F8+X d(KQX'V5 %!
_l}tfKLC61KYKuZIGcPhG20@y}|Y
nP`&Ik
+B^\,TrU4RP[_:k}xz5B;S\;<Tojep5ptmGO
eJ9oMVwb8wG"WPd)]b~A}cJ
pNKRV,pUB6FjBQcDUdRT%g) S@L2d
)wL4M_`f@6<MdIF-FNTF,+*_824f,:i%Aw?M0q9)'lvD^!]qPP1GBD*$8TdkR
.6uj>DX(BFHP(I!g&!.:\[qw'.#]Q_IZHiR-)PEK&h{Nr~g=K(.as+|vb3g_JWrAw_|^t
hjj=;Za1)/?"{1s1.&s.~,ICzY=BNLQ|l)qS.^j+WbL]{gBj5){}*>vm=xHhYu*(-oe!H
t)uymz7WPyO?{.tm9_]*0nx][T:???h\7c'hz$#8{zd$Dy_7Qv~B$l({w)Mf%jzvU
=6a]kls~22h5j%%I}u[K7j{N6o^mo*7jKujJe7o+vh}6za$%dCImO/!'jKGhgnoQio)3UZcHG}=dfj
ve]w+`7AlG:C{O:lk*WF]hfd9/w>{dl.}[sWo=J0Jg:FwKP_o+l6:@g:Uf={@9^z$50^<N<B{SO,Fon;W/||pmK<
~z}>$ ~8iN~)^5xErW^:_w//}[7di
B7O@~a4H1WzU;oS}3>bWm2ki7mP0[p0teWFUvBvP~.y
<\vR
 q)fOn=+Rk$qD;Vp/5)y{##>i#7Bd.GwKp]6cJR67CGB3D#O'g/m8}e[9x::kX1u:lo#vHsmh|JY&ED'g?-
"m:aKNb[cN2hK ILj""z$&y}q<2~8$NtO:E9rtc[
nmu_\q4i@t?k)\o_qN]}UQq~~rAX]{mpm~/\m[J[/qN7odY/d?)"N-
toPLRSaSOgj~X6&^]frd#C-]@W=:GlVO~)$@`,d[]f?'MrQ:D>}UL?U6F@ ndum6_QQjTj9RQ?#V&(.+l
gmRJ}|"C/ZeE*HJIvqls?t]%Hqd?nE:w
lZOR+hLl2cUe]#NL6a4U52lNJ2J
VR3rF&)*!!a+%LFZMpj)%>W,R
\2[<#gn
{gIR-acKKJUZ$n6YiqeM@]X\9h
q{E*wju7^sTZ&tijj[i=H(U`+ zTn^WnP-fJoNBB^l"(q JM1jmxLAqblB;x>J
Ua\ZjnXGMdc|~+Z%g\UDdRRmvv{Q6+n[e'{P>|i=",6,o)`_`EC@:
(s@C]3u]JEjN* @ hu Zt2 t]-]JEjN* @ hu Zt2 t]-]JEjN* @ hu Zt2 t]-]JEjN* @ hu
"hmrprC=zB@@@)pujG~i--RUWJ/H@@@h/p`u1a~tM1Pm/U.>C(7h[n@wAB@@@
t~zX\~U(.&6?n_==+~tehTTR@CWFnKDX`=-%*E%@NhsB;qEtDc tnzX[vqr_wPq#`6\tjrCS_(cO:pe:mWD
~
YV@[^]? LhAOr'%'hE
on\MmmM_2Q}SDsYb&4Z?/g1~myyg/:~3u#5GxSdADCGU+]lVK3]T'<%.Vx}J*PU.(9T{W?0%rGT]['WlU\|"H>]
Fui#(22HQyz];z;_*(0 C}=r7YS-hZ~H;&?jw~N1um

x=zL.6XEW}#KNu]{UZ7=BrCf\}?*oa=,bu1?g{jYie[m.o3r2$8(kNS7;8{P
N@tQ#?BO~y"o, @eM~*tX]0qP*FwZosW&z9i{uZ}h"+}GTAw?j7 i|9ZWc:
>@3Ft8,X&$Co0LX=EJ['zc6{+nXo.RCGizM]-QLNe$i;v[`'/U-
Y_?T[%Un=0N=HUzq&x]]%RGqRm(R5Z=G>FNM0<gr?@w_o+d(3"$@
J*_U3|Leptzb{N4zh;v_8{k3kgye%PrtO="k0uJmQMuMaW!g/st
$k/^VN@jHXRS, Xt^Q=[AG)3(~T=;uw9zE/zy&
}d!JlozMau-K),* SXz>w^_^}PU))):0W'nib~4CzhN31&K .*Bq=H @@Ybo"p)vBB govoUV!ucpvTT3z)(Ev
w:)Stj]q-XAkDs=QE?}aNy?}U(h7?.i'X>u
z}:^Y_r,g>QJN9Km;>b|)uJJLPj|7%*"RR/jo>&+fhz4'[0Gz7v?h<A
_07Zu\63
u[!G p(!qmCZ:7hay_YN:O5nb/-ZJ]v`)9!NgwZY>hv>Du
+[a#Fh03J2wv9*{CjRyG~e-Sc=+lH_BBvEYsHpY[.9Kv6hL9ch3iy
Yzu&m)7
{&w~eY/*})>6;^mCN

t+H,Z !A Ac74?R g$v|8t 9-Egy=]`1l0}eQ-=_&|!x Vxq|7EOdfj6.F6
U[MPVUc\QaFXl-
z`U{m}),!f^wX[gS,bwAWHlCeUk/Ru+@@@8D:7h*mA-t;jd/--
W~Y9Qfx>jG/VhQMq,EFQvidW*mR6_EBDm8Hdj)JUmm+[TVjHz,j;-;'Rwg:laA\Yh(-
&RO<@@@@8Tw_gZ9,btq1nFEOY4`a
w
t;2=:>zE
!>rnM8LIk
"rmM@tz.\8l(=mEy4r}LM4\lJ
k6a>2\._M(fnV#{e2mC*c4b}8' 8p]l~]pw/16Gw>QunY6)Ll+mNE*w=x?UoZy
%Phe))6qyFz%gj#+a,z$nQm\EYo=&Ney[`k${y*x
z*&a*,/WVCV57&\Uz&J)Qut\X}~q@@@@KK
ts.uC"z0'w1G]_X2.:Ta'bcbl.:U!ueHB}$;n:U5/fA6i~jmL
!}3>RElP;wj(AMZQ]gV{ll/#UE?N/7fm~kyERl.w_SHSW7uU}#n/PQSW8Y$%J];w:Y}R"RD&U~c[uu4p^6S>@Y
s3>zsX@W>9_;MaG: N!i0v?}&++K:E"qO=+5:u(r-`Yil/90uhy~i-0svleimL;J+-
WY[}xA7eCMw.J[]KgUzW+{Cql\4\+4ipS;RNK:;i@@@@J&ao\>6 wxF.eBB BqWYA>vuYy{JX~ptt%^-PbJ[>I]
!^ M+zhdClp(
@@qT;a[~E|p0rnMQZBz%D)(tsaYo(!{wi~9vZ
I[n5grAw?6t]_$[Ww{(wn,W<Vb! pUev"/UJeKCP|-=p_8I`
"vCr}v}u WW69O7wVigy|Zo.;]=U4y HFksn]c
:w/-U^8U#+}Z8AW?7,_GpYkW!l"oi}~IjCCm.7jkUmP_Uc=0PM5d5%lVVr&ul*Tn_tNv5X0Z6
qkY2vWS^6t@FDyc`"M.i:N,^Qn?e-+Y>sn^g}1'+[w8KTXxm}srR78d
voSC{eTd!zuf }R>WS^]a0-\P.
zq
dJJl0
9H9}-`qzI7VHc5wX_Z5O'NU/l//<Jjf/MN>Z'LQKK.Fi#z;Dvkh8=d*j2Y'6EW"WcqH]9k$GVVF_ CcI:
@cy~1D}n2c"Tj+l)S(|W95~:
Zn[=2
P`;EFk:c0`3:=\37E-zlnFk}&F;nD
jSo8**EGn<~F{8cK?9uN[5kb(hK_=:]^3^Z`'mkh|zu5sprLwb~LYC2=1O?n]Ok>tLzORto4n=kvgk//^&
CUQj'vmiuy2]Z_?DNvDCUZ^8\UK>B9T@>:Wi^,}rJ&W/n{m]5skU: ~z{*m*-T
9|~~eoom+U^

}qKUwR%-5ap]7~"M_v}cuB:G+3>\7lY=ye Izp%?+o#JS
<6;(;.Gi=|]1&F][9V?;yZT2"Lzu4+O1q}pN4L#<wz_0E6d-_#V8c'6m+rtiw'K+ykktHL
P-"l%xvxSDi_h[&:AwQc(,W4u#Q%u(E3@_ %%w42!\%;3l+=~i:SYQz3,8__\uqUlhZK_9b>,-
J%;>n,PjdT2qF7cAj`zty2F*ZT~)Fo_> !@@x|acQ13ehS+,(]GM6bNh.Mt*;P }jruck-hasDP]vj|-vq?vEyq
i{uXNn)\xXekmY<ELTFj]E%&.i_=Z]NV*U6wU7F
j.AS@/vG7LxP}J5v`|xRyf<wjbp7SCn804}HVEYJU7l(M?T=~=t- hS
M8O S0NYv]*m!x&&u@IDATCvbe[i:i4*[%76&#XJ{OgZU#|zS3#M.xCWDwj-iwN=B%M9e
%):BXT/xN;l[A?~=x^rMj5cFPbFu6oF8 W{g;vt{U;~9}vxlz-]#j qc|jj]{9YUa_u
:cLE,[6ien-~O[-< >gUm2{KvhqCfBk5u5W1iMc{
lXE~Ro;SK35Q`S6O=mK|lJx};&`/G\I+"ir#X5OGm;8>VyHKh-Kh/^'>#GSi}kPckR}|@|-
6im#5zzS6],8#5bp_]81eUB/QBLRmHO6cea}dTel.9lJjKQ@/9 uZt& E
7G9cuMnk}N4
pu6OW3V=H%1I:PlMsWh8/A[w)0lz}.(,{*u@uK}I=~yO?rTg]ms 64mVBK6ha{Y{_]3FobnVab.<+O.&M$'Kw-
Lisq,QJl30@%y6NC-mm
W
ii@*zw
U6xp\p T=u Z7Vqw
Bz&6U&pBLgm4Loi}VmTca!$AW6H]R8t"lr{.[^Z_r'SA}+Owwf4?W.!izitR^<}6CE6'yC-
+rD?c.NK3%Gi+\uMj,&Lax#Vu)nH=;h'|R}ee5u$oQ.dOIgYv^*yymFs[U;a
Gy#'[k K[s& s6in#IXZZ?6=c;u\(Bl:_tRb=pvB8`\Y_xn1> vZo~=4|nEPry#.zN5haZ5I
w%wDhs!F]#_]<C/Cu]r'59Ni8PBdLXR:>91AQv.df_=]eUIjtf]ur!\IqzcWzoq6DT~]^N1M
;\G&t'Z|y~\("z4h *(v+^1Jh~oZ8z3zPw}?`>z[VqspD*|6Q]WPM>$",H7 @zU;l"pWaizIge/fcc]Sg}m)kklq)ghsIy:ub7s
#C)/<\F
nqvlj"y.x1|`?[v4M7e;9w*|NzLAY:vAdo0w!5 @dFIPw 6?DBsv
WmjbwTZJVh@RFdL2'nXQ;pv(_#[jA9*]/:4ds^n >ps0=sTZYk`w?.rM)LOScc }W*9Ekp~Zp;
bm8WlyHNn9YwEMo8z:R]; m(w?QWb7,z#WwxK;E@3h{&"|Z !cj[MP sagM1[MAw<m_nJ1ULi@vjUA
!$kCNzPT@@>Lu@@@LZ|@@@>L: |N7I'Mis@@@,pP.N[V2 @@@
Zt=*4o^DR@/>@
c`/ZXWQy
@@@$phhP%kGb`eZOH7g$I5u,)#$@x#\3Rp1y[]y|Lwcf| g!&Jx,`ELLRX]pv4uNKw_Yxv2
GuJWT>J Y
Qja+.xAB$hi?quO>~ko7Z;^n-=zyClQ.@#blQ2s~lph
<hC'(z@
MWF\t:4dyu!Y
eH^CzuX,]EZ?ngXUo`I&^P>5u3iI3>,'Vl=Nv6EB h7&w-V
$
UOUNqn=u/,p$hh{mHWOnUXY btvYUK%7F&|hh;cW.lKCzrSA{ZC 3;]u]U
>h}auT{DY!4*'gNm(t;swfMl]&'UQbtD4=)iNr^viuolc.X=6Ov]RbcZNS,JD-
vCzg|Vkejf89.VC/O'H=>v2]Pz(y7my?s{m42v}^<Cql6n{{omz>>k;`w
'l %ipkpN*mzcw?p-EGD(=1N qrz'Q?ua:J9Z28k@25i+A>wL?64$
NgU4Su+K5wNJkx&X>&d" +CgP~oRuIVwozwtv6j32`cf*NhKQ}
a3N
exq[^v3)sI}2TWZTY9zs[zwO)v*-K]sin*+*rzEDhjd+O8qym-Q2K'PU@ K!{~{EG[)^~YjckXavEn;W0 Ap#KI
6)glw;q;-cz,2-;_q6~Eg=']}wUhqVaRg+1%h>483I?8q^pov}'a*./WLB.]KtaJSu\nyqC+V
U~|mU-o<]SOGOZAYY[z|Wnu~v Y@To=GKs>*gNz+_jj$l~_G4j\
G]%Du#AMgI"-Sg?jS'E[&L QM%w#Y6A#!]v]szW4 `, Y [9uT@k4]x
tn*WtHnn1;[o{66v0g"#taT+MYJ(W>@^Xo?@kfkKu^I~@}V5%G
cys2};~`Sjklh]O/H]0{{KsO'sV&8{fY!W=sQ:5P0[zfs >HM_>nn>~NM[lh3ju;rpn>f~\v0e<M'kX.?=6WjkqeP}5!G

?)EEB@xE*UC(-(pKu-Sv4v}}Qk6)6T#$>#_jVdu}+V_QW~ulDM[Se;TjHaaTcF7p$&E)%Hj+wk}Y\KkJllnh
=hH9 S6$&1fVP
_-=25")B5:_mq}S#,=lwjK^eM_I
KV0EYFLGUTGjj[7[@k&j>_

t<abw]5U]Z]RoI va=w[Sq^o?y7rTc1zS&j$J?B6l'l6u{*'

PQ ej"}^1B?<HR
oIT ZnA/`c]/zWlbx;8M9knem~7n,xus>40()tYw~{vgVlU>>$\."cGw
KK.{FX[XCJN\1w +^ufs'z&/M*OwOE|&"7f{RE-
s8wxEj!oUNe":#]>nV_/RQM:%*>t;nffg*6urton=`w^5&xesk_Epo{eM~"5nU\{WCMhc}2w 80@|
%Y6N;_fW?<olZm-\/$utxrP^Ob.J}25q`1 QUy%^z"AhRd`=.<zia'Q%Y^X3@5[jG&+&[^V3+/W.*6}
z1V[whkhJXcC>4VYq5zq%^IT*\^Sj[l9UjhG Z-li'{7^]7_lm6cz^9,h-
1CTkg[8{R%kYn~&z@=>*15"H+*CrX\^%*wkcW%
(z>|~@oHGx}u\OM-]{&GZJW{zODZBv%ZnH{HgqDzQUlE?j(TmJnL-j
y:& =at.V Y@@@ -] OWuE(if{/p
ZS~i-Yn
<|fVRm-whVH
q;JRg:\cQT[B`on.f"!]eO@E5hM(Z}vL]7$Z.}k9]=-
V8MdTKj^Qn)@dpeuV:7Ha{JV]9H`R,8+lNg\x;;Xj/{x}6^6uuS_4&&~@Dv,u222eCJ%wlU4piw:
%***LEDD}&+bh"p=Uj+`Pw[zz&z*(585^;JT.;`juoy}v@zNM;l]Pj.{hgn[mgu|z:}\?%nf\7<ds446K
!m6
GknHKeMm\`zP,zxz&lhB[KO~G Xl;K]s 9%[qo.[CMEjuR?8LtMV?Fk:trm(K?^{i:n,?Q^9o 8pAW^^UR\ne
7!nz+}eVZlzG)F6pCSH "p`-i}+Z7{RVV8J@h,oJEA@"t9Pu ! p;?fwi>gNG}W6
My 'ty[IFB@ct{ZtF%yAz3}/ @0](wR IIIvGj_.h&"w}@+Vbt*:*t}iDZ|=>e"##RJuVb g/N\0rD
0}V-s'$@'&&]SuF\:$O[.ZJqI#Q@
ZcIvI 1=]N_.t Xt&B g.s\v%?7K=Ao}uF"hB'w;'t]y wT
`|gx@cAm:*E"! ptNC@@@6=-,Ni8
@Wh
ZJ?aQT@@MA0%Y+sSJwS/-M
@@@N7.^ce" @ Vf z-6 BE J" p 8# hf* '@k3J @ Z<
DudP$@@:.t-G4(U[6&FQ{ oh566!n'J @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8{w]UyqIN1PT$TL A8,ojm[nzvY)^,zEuu2(pdNNN"/=Ng?w# -@h!
Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -
@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! q_Y~%> p-R%&=!9IyA
KE' D
-,1UY:OWCJ]oU{UqW
.pB:M<SO]e^t}8Rm=di@@@OXL=}!U;$Kwz_Du"/c@@@@g>=_N(hn{U\)6U[D:kGM&jC_&@@@@CUH<k
u|\{m;Z4O]pa@h
Ru1g%"kA@pqX5@@@*-52,.=tp>M^nbT{S/\eCk_z;k;7@?9J-8fU~x}-U}=;{p%}|
)'sZVe|5Pct3s*~u7H~|9{TX^=[TVk^XtBmZ0q{$471nkoFP

5Xr6wl{;@@@/Z=|?}-x~,@XbEI7Y(K^O[TW[N{Zr\eOe+?GiuZ~aS]e
>]Sedfm/5k*+Ttq.?@|!JdO)C4zzW}=fTUc/m -|[HUMn7<R]$> a/BtUlBi~>];2+-
RYCTxW$v\}z]=>[{5uBmw]R:4m7/\9^v$~G
=s|Otqs`A@@@xE8k8]>f6dV
|Cw{=mvYj;x)*,,8 uW~97{f?-suwg5[Uk-;V?Xuwj4ogmas~[kstk Z&}Oh?^){WWhqz=ec @hhF!Mlea.{|v1k7x7Wqj
kEACcJ7ek-Y*..o^{S
q
;6Pc}7]{Hh;OUUQ?V
13G=jkP<Wio)qs`Lp^F]?wmZ @blZAO^l:dMg8*,$6RQf{Y&"5Una#=S3bU5u^&<[vpZU%/jGn3*v
+zDlhzgy' Z sZCu _{uon7jzheYF-7Y
3@Vj1hMmRW@aa->%<S(iFZF& q&N#joU+aW7;i\ot
M+ta'f[xNVL
 S k d TZ~7/ETiye}6XyU5zUzzgu~_,@K+(W)&*wvs@Cqys`!5.7yN:hp=K_]]WKtu/-
7^zk$b~ez'ngVo>;jxFW=Vl@@@@kpz4'^Nk&::1?T610{}i]Jx@\q.0Z>Q\V}{?/Zf(kTu
[6y]?SjJKTk;+5g0Aj
JClAmK33|,R}iWYU^x
zgGxOM.lC;U -1~aCYI>BX{d{G'?oR'tCz6,$YJS#M,Z&8"`D-B/0Xb`Hl
S&zf
'u+;u.^vC*6a~NKn)6hX*:K{Jl? E6?)-e @j-uC597u*Ix!IDATyU^}6cla >y"b-H
"Y.p)c*AlkdP!
#q,s'$*%igaEYRb%,,j~;z , D=M=sZWWog{zG{+6{CpszU}itI+s[>Q^4\Z^;}cn# T=t[QP^}Z*Zyoo[*'*52-u"
8O WQp5]V|M\@@@x
~yX@@@@*-"4@@@~J[i BSYx>{Fva: mC+< C_{6,h% D@AA-VZkHKL $PTTb%"p%j@Z{`5}_ -
BmY)))rSY4=CiZm>-*WnEe
?MCowMno<z~|<Mg=r{}'z7g+JEw p-O<nmmn{^Ejjndgg7^u55V]]"C`#tx{P%.Pmk
B>5nwby(*V~ X *
\<oC=D)x?bKJIENDB`
[[managing-licenses]]
== License Management

When you install {kib}, it generates a Basic license
with no expiration date. To start a 30-day trial,
view the status of your license, or install a new license
go to *Management* and click *License
Management*.

[role="screenshot"]
image::images/management-license.png[]

The 30-day trial enables you to try out the full set of
https://www.elastic.co/subscriptions[Platinum features],
including security, machine learning, alerting, graph capabilities,
and more. For example, the following pages are added to the side navigation:

* **Machine Learning**
* **Graph**

The **Management** page gains options related to configuring security, managing

{ls} pipelines, and using {watcher}.

At the end of the trial period, the Platinum features operate in a
{stack-ov}/license-expiration.html[degraded mode]. You can revert to a Basic
license, extend the trial, or purchase a subscription.

For a comparison of the Elastic license levels,
see https://www.elastic.co/subscriptions[the subscription page].

TIP: If {security} is enabled, before you can install a gold or platinum
license, you must configure Transport Layer Security (TLS) in {es}. See
{stack-ov}/encrypting-communications.html[Encrypting communications].
ELASTIC LICENSE AGREEMENT

PLEASE READ CAREFULLY THIS ELASTIC LICENSE AGREEMENT (THIS "AGREEMENT"), WHICH CONSTITUTES A LEGALLY BINDING AGREEMENT AND GOVERNS ALL OF YOUR USE OF ALL OF THE ELASTIC SOFTWARE WITH WHICH THIS AGREEMENT IS INCLUDED ("ELASTIC SOFTWARE") THAT IS PROVIDED IN OBJECT CODE FORMAT, AND, IN ACCORDANCE WITH SECTION 2 BELOW, CERTAIN OF THE ELASTIC SOFTWARE THAT IS PROVIDED IN SOURCE CODE FORMAT. BY INSTALLING OR USING ANY OF THE ELASTIC SOFTWARE GOVERNED BY THIS AGREEMENT, YOU ARE ASSENTING TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH SUCH TERMS AND CONDITIONS, YOU MAY NOT INSTALL OR USE THE ELASTIC SOFTWARE GOVERNED BY THIS AGREEMENT. IF YOU ARE INSTALLING OR USING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY.

Posted Date: April 20, 2018

This Agreement is entered into by and between Elasticsearch BV ("Elastic") and
You, or the legal entity on behalf of whom You are acting (as applicable,
"You").

1. OBJECT CODE END USER LICENSES, RESTRICTIONS AND THIRD PARTY OPEN SOURCE SOFTWARE

1.1 Object Code End User License. Subject to the terms and conditions of
Section 1.2 of this Agreement, Elastic hereby grants to You, AT NO CHARGE and
for so long as you are not in breach of any provision of this Agreement, a
License to the Basic Features and Functions of the Elastic Software.

1.2 Reservation of Rights; Restrictions. As between Elastic and You, Elastic
and its licensors own all right, title and interest in and to the Elastic
Software, and except as expressly set forth in Sections 1.1, and 2.1 of this
Agreement, no other license to the Elastic Software is granted to You under
this Agreement, by implication, estoppel or otherwise. You agree not to: (i)
reverse engineer or decompile, decrypt, disassemble or otherwise reduce any
Elastic Software provided to You in Object Code, or any portion thereof, to

Source Code, except and only to the extent any such restriction is prohibited by applicable law, (ii) except as expressly permitted in this Agreement, prepare derivative works from, modify, copy or use the Elastic Software Object Code or the Commercial Software Source Code in any manner; (iii) except as expressly permitted in Section 1.1 above, transfer, sell, rent, lease, distribute, sublicense, loan or otherwise transfer, Elastic Software Object Code, in whole or in part, to any third party; (iv) use Elastic Software Object Code for providing time-sharing services, any software-as-a-service, service bureau services or as part of an application services provider or other service offering (collectively, "SaaS Offering") where obtaining access to the Elastic Software or the features and functions of the Elastic Software is a primary reason or substantial motivation for users of the SaaS Offering to access and/or use the SaaS Offering ("Prohibited SaaS Offering"); (v) circumvent the limitations on use of Elastic Software provided to You in Object Code format that are imposed or preserved by any License Key, or (vi) alter or remove any Marks and Notices in the Elastic Software. If You have any question as to whether a specific SaaS Offering constitutes a Prohibited SaaS Offering, or are interested in obtaining Elastic's permission to engage in commercial or non-commercial distribution of the Elastic Software, please contact elastic_license@elastic.co.

1.3 Third Party Open Source Software. The Commercial Software may contain or be provided with third party open source libraries, components, utilities and other open source software (collectively, "Open Source Software"), which Open Source Software may have applicable license terms as identified on a website designated by Elastic. Notwithstanding anything to the contrary herein, use of the Open Source Software shall be subject to the license terms and conditions applicable to such Open Source Software, to the extent required by the applicable licensor (which terms shall not restrict the license rights granted to You hereunder, but may contain additional rights). To the extent any condition of this Agreement conflicts with any license to the Open Source Software, the Open Source Software license will govern with respect to such Open Source Software only. Elastic may also separately provide you with certain open source software that is licensed by Elastic. Your use of such Elastic open source software will not be governed by this Agreement, but by the applicable open source license terms.

2. COMMERCIAL SOFTWARE SOURCE CODE

2.1 Limited License. Subject to the terms and conditions of Section 2.2 of this Agreement, Elastic hereby grants to You, AT NO CHARGE and for so long as you are not in breach of any provision of this Agreement, a limited, non-exclusive, non-transferable, fully paid up royalty free right and license to the Commercial Software in Source Code format, without the right to grant or authorize sublicenses, to prepare Derivative Works of the Commercial Software, provided You (i) do not hack the licensing mechanism, or otherwise circumvent the intended limitations on the use of Elastic Software to enable features other than Basic Features and Functions or those features You are

entitled to as part of a Subscription, and (ii) use the resulting object code only for reasonable testing purposes.

2.2 Restrictions. Nothing in Section 2.1 grants You the right to (i) use the Commercial Software Source Code other than in accordance with Section 2.1 above, (ii) use a Derivative Work of the Commercial Software outside of a Non-production Environment, in any production capacity, on a temporary or permanent basis, or (iii) transfer, sell, rent, lease, distribute, sublicense, loan or otherwise make available the Commercial Software Source Code, in whole or in part, to any third party. Notwithstanding the foregoing, You may maintain a copy of the repository in which the Source Code of the Commercial Software resides and that copy may be publicly accessible, provided that you include this Agreement with Your copy of the repository.

3. TERMINATION

3.1 Termination. This Agreement will automatically terminate, whether or not You receive notice of such Termination from Elastic, if You breach any of its provisions.

3.2 Post Termination. Upon any termination of this Agreement, for any reason, You shall promptly cease the use of the Elastic Software in Object Code format and cease use of the Commercial Software in Source Code format. For the avoidance of doubt, termination of this Agreement will not affect Your right to use Elastic Software, in either Object Code or Source Code formats, made available under the Apache License Version 2.0.

3.3 Survival. Sections 1.2, 2.2. 3.3, 4 and 5 shall survive any termination or expiration of this Agreement.

4. DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITY

4.1 Disclaimer of Warranties. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, THE ELASTIC SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, AND ELASTIC AND ITS LICENSORS MAKE NO WARRANTIES WHETHER EXPRESSED, IMPLIED OR STATUTORY REGARDING OR RELATING TO THE ELASTIC SOFTWARE. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, ELASTIC AND ITS LICENSORS SPECIFICALLY DISCLAIM ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT WITH RESPECT TO THE ELASTIC SOFTWARE, AND WITH RESPECT TO THE USE OF THE FOREGOING. FURTHER, ELASTIC DOES NOT WARRANT RESULTS OF USE OR THAT THE ELASTIC SOFTWARE WILL BE ERROR FREE OR THAT THE USE OF THE ELASTIC SOFTWARE WILL BE UNINTERRUPTED.

4.2 Limitation of Liability. IN NO EVENT SHALL ELASTIC OR ITS LICENSORS BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY DIRECT OR INDIRECT DAMAGES, INCLUDING, WITHOUT LIMITATION, FOR ANY LOSS OF PROFITS, LOSS OF USE, BUSINESS INTERRUPTION, LOSS OF DATA, COST OF SUBSTITUTE GOODS OR SERVICES, OR FOR ANY SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND, IN CONNECTION WITH

OR ARISING OUT OF THE USE OR INABILITY TO USE THE ELASTIC SOFTWARE, OR THE PERFORMANCE OF OR FAILURE TO PERFORM THIS AGREEMENT, WHETHER ALLEGED AS A BREACH OF CONTRACT OR TORTIOUS CONDUCT, INCLUDING NEGLIGENCE, EVEN IF ELASTIC HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

5. MISCELLANEOUS

This Agreement completely and exclusively states the entire agreement of the parties regarding the subject matter herein, and it supersedes, and its terms govern, all prior proposals, agreements, or other communications between the parties, oral or written, regarding such subject matter. This Agreement may be modified by Elastic from time to time, and any such modifications will be effective upon the "Posted Date" set forth at the top of the modified Agreement. If any provision hereof is held unenforceable, this Agreement will continue without said provision and be interpreted to reflect the original intent of the parties. This Agreement and any non-contractual obligation arising out of or in connection with it, is governed exclusively by Dutch law. This Agreement shall not be governed by the 1980 UN Convention on Contracts for the International Sale of Goods. All disputes arising out of or in connection with this Agreement, including its existence and validity, shall be resolved by the courts with jurisdiction in Amsterdam, The Netherlands, except where mandatory law provides for the courts at another location in The Netherlands to have jurisdiction. The parties hereby irrevocably waive any and all claims and defenses either might otherwise have in any such action or proceeding in any of such courts based upon any alleged lack of personal jurisdiction, improper venue, forum non conveniens or any similar claim or defense. A breach or threatened breach, by You of Section 2 may cause irreparable harm for which damages at law may not provide adequate relief, and therefore Elastic shall be entitled to seek injunctive relief without being required to post a bond. You may not assign this Agreement (including by operation of law in connection with a merger or acquisition), in whole or in part to any third party without the prior written consent of Elastic, which may be withheld or granted by Elastic in its sole and absolute discretion. Any assignment in violation of the preceding sentence is void. Notices to Elastic may also be sent to legal@elastic.co.

6. DEFINITIONS

The following terms have the meanings ascribed:

6.1 "Affiliate" means, with respect to a party, any entity that controls, is controlled by, or which is under common control with, such party, where "control" means ownership of at least fifty percent (50%) of the outstanding voting shares of the entity, or the contractual right to establish policy for, and manage the operations of, the entity.

6.2 "Basic Features and Functions" means those features and functions of the Elastic Software that are eligible for use under a Basic license, as set forth

at https://www.elastic.co/subscriptions, as may be modified by Elastic from time to time.

6.3 "Commercial Software" means the Elastic Software Source Code in any file containing a header stating the contents are subject to the Elastic License or which is contained in the repository folder labeled "x-pack", unless a LICENSE file present in the directory subtree declares a different license.

6.4 "Derivative Work of the Commercial Software" means, for purposes of this Agreement, any modification(s) or enhancement(s) to the Commercial Software, which represent, as a whole, an original work of authorship.

6.5 "License" means a limited, non-exclusive, non-transferable, fully paid up, royalty free, right and license, without the right to grant or authorize sublicenses, solely for Your internal business operations to (i) install and use the applicable Features and Functions of the Elastic Software in Object Code, and (ii) permit Contractors and Your Affiliates to use the Elastic software as set forth in (i) above, provided that such use by Contractors must be solely for Your benefit and/or the benefit of Your Affiliates, and You shall be responsible for all acts and omissions of such Contractors and Affiliates in connection with their use of the Elastic software that are contrary to the terms and conditions of this Agreement.

6.6 "License Key" means a sequence of bytes, including but not limited to a JSON blob, that is used to enable certain features and functions of the Elastic Software.

6.7 "Marks and Notices" means all Elastic trademarks, trade names, logos and notices present on the Documentation as originally provided by Elastic.

6.8 "Non-production Environment" means an environment for development, testing or quality assurance, where software is not used for production purposes.

6.9 "Object Code" means any form resulting from mechanical transformation or translation of Source Code form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

6.10 "Source Code" means the preferred form of computer software for making modifications, including but not limited to software source code, documentation source, and configuration files.

6.11 "Subscription" means the right to receive Support Services and a License to the Commercial Software.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

```
  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
import * as Rx from 'rxjs';
import { catchError, mergeMap, map, switchMap, tap } from 'rxjs/operators';

export const RETRY_SCALE_DURATION = 100;
export const RETRY_DURATION_MAX = 10000;

const calculateDuration = i => {
 const duration = i * RETRY_SCALE_DURATION;
 if (duration > RETRY_DURATION_MAX) {
   return RETRY_DURATION_MAX;
 }

 return duration;
};

// we can't use a retryWhen here, because we want to propagate the red status and then retry
const propagateRedStatusAndScaleRetry = () => {
 let i = 0;
 return (err, caught) =>
   Rx.concat(
     Rx.of({
       state: 'red',
       message: err.message
     }),
     Rx.timer(calculateDuration(++i)).pipe(mergeMap(() => caught))
   );
};

export function watchStatusAndLicenseToInitialize(xpackMainPlugin, downstreamPlugin, initialize) {
 const xpackInfo = xpackMainPlugin.info;
 const xpackInfoFeature = xpackInfo.feature(downstreamPlugin.id);

 const upstreamStatus = xpackMainPlugin.status;
 const currentStatus$ = Rx
   .of({
     state: upstreamStatus.state,
     message: upstreamStatus.message,
   });
```

```
const newStatus$ = Rx
  .fromEvent(upstreamStatus, 'change', null, (previousState, previousMsg, state, message) => {
    return {
      state,
      message,
    };
  });
const status$ = Rx.merge(currentStatus$, newStatus$);

const currentLicense$ = Rx.of(xpackInfoFeature.getLicenseCheckResults());
const newLicense$ = Rx
  .fromEventPattern(xpackInfo.onLicenseInfoChange.bind(xpackInfo))
  .pipe(map(() => xpackInfoFeature.getLicenseCheckResults()));
const license$ = Rx.merge(currentLicense$, newLicense$);

Rx.combineLatest(status$, license$)
  .pipe(
    map(([status, license]) => ({ status, license })),
    switchMap(({ status, license }) => {
      if (status.state !== 'green') {
        return Rx.of({ state: status.state, message: status.message });
      }

      return Rx.defer(() => initialize(license))
        .pipe(
          map(() => ({
            state: 'green',
            message: 'Ready',
          })),
          catchError(propagateRedStatusAndScaleRetry())
        );
    }),
    tap(({ state, message }) => {
      downstreamPlugin.status[state](message);
    })
  )
  .subscribe();
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { EventEmitter } from 'events';
import { watchStatusAndLicenseToInitialize, RETRY_SCALE_DURATION, RETRY_DURATION_MAX } from
'./watch_status_and_license_to_initialize';
```

```
const createMockXpackMainPluginAndFeature = (featureId) => {
 const licenseChangeCallbacks = [];

 const mockFeature = {
   getLicenseCheckResults: jest.fn(),
   mock: {
     triggerLicenseChange: () => {
       for (const callback of licenseChangeCallbacks) {
         callback();
       }
     },
     setLicenseCheckResults: (value) => {
       mockFeature.getLicenseCheckResults.mockReturnValue(value);
     }
   }
 };

 const mockXpackMainPlugin = {
   info: {
     onLicenseInfoChange: (callback) => {
       licenseChangeCallbacks.push(callback);
     },
     feature: (id) => {
       if (id === featureId) {
         return mockFeature;
       }
       throw new Error('Unexpected feature');
     }
   },
   status: new EventEmitter(),
   mock: {
     setStatus: (state, message) => {
       mockXpackMainPlugin.status.state = state;
       mockXpackMainPlugin.status.message = message;
       mockXpackMainPlugin.status.emit('change', null, null, state, message);
     }
   }
 };

 return { mockXpackMainPlugin, mockFeature };
};

const createMockDownstreamPlugin = (id) => {
 const defaultImplementation = () => { throw new Error('Not implemented'); };
 return {
   id,
   status: {
     disabled: jest.fn().mockImplementation(defaultImplementation),
```

```
    yellow: jest.fn().mockImplementation(defaultImplementation),
    green: jest.fn().mockImplementation(defaultImplementation),
    red: jest.fn().mockImplementation(defaultImplementation),
  },
};
};

const advanceRetry = async (initializeCount) => {
 await Promise.resolve();
 let duration = initializeCount * RETRY_SCALE_DURATION;
 if (duration > RETRY_DURATION_MAX) {
   duration = RETRY_DURATION_MAX;
 }
 jest.advanceTimersByTime(duration);
};

['red', 'yellow', 'disabled'].forEach(state => {
 test(`mirrors ${state} immediately`, () => {
   const pluginId = 'foo-plugin';
   const message = `${state} is now the state`;
   const { mockXpackMainPlugin } = createMockXpackMainPluginAndFeature(pluginId);
   mockXpackMainPlugin.mock.setStatus(state, message);
   const downstreamPlugin = createMockDownstreamPlugin(pluginId);
   const initializeMock = jest.fn();
   downstreamPlugin.status[state].mockImplementation(() => { });

   watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);

   expect(initializeMock).not.toHaveBeenCalled();
   expect(downstreamPlugin.status[state]).toHaveBeenCalledTimes(1);
   expect(downstreamPlugin.status[state]).toHaveBeenCalledWith(message);
 });
});

test(`calls initialize and doesn't immediately set downstream status when the initial status is green`, () => {
 const pluginId = 'foo-plugin';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green', 'green is now the state');
 const licenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(licenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);
 const initializeMock = jest.fn().mockImplementation(() => new Promise(() => { }));

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);

 expect(initializeMock).toHaveBeenCalledTimes(1);
 expect(initializeMock).toHaveBeenCalledWith(licenseCheckResults);
 expect(downstreamPlugin.status.green).toHaveBeenCalledTimes(0);
```

```
});

test(`sets downstream plugin's status to green when initialize resolves`, (done) => {
 const pluginId = 'foo-plugin';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green', 'green is now the state');
 const licenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(licenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);
 const initializeMock = jest.fn().mockImplementation(() => Promise.resolve());

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);

 expect(initializeMock).toHaveBeenCalledTimes(1);
 expect(initializeMock).toHaveBeenCalledWith(licenseCheckResults);
 downstreamPlugin.status.green.mockImplementation(actualMessage => {
   expect(actualMessage).toBe('Ready');
   done();
 });
});

test(`sets downstream plugin's status to red when initialize initially rejects, and continually polls initialize`, (done)
=> {
 jest.useFakeTimers();

 const pluginId = 'foo-plugin';
 const errorMessage = 'the error message';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green');
 const licenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(licenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);

 let isRed = false;
 let initializeCount = 0;
 const initializeMock = jest.fn().mockImplementation(() => {
   ++initializeCount;

   // on the second retry, ensure we already set the status to red
   if (initializeCount === 2) {
     expect(isRed).toBe(true);
   }

   // this should theoretically continue indefinitely, but we only have so long to run the tests
   if (initializeCount === 100) {
     done();
   }
```

```
  // everytime this is called, we have to wait for a new promise to be resolved
  // allowing the Promise the we return below to run, and then advance the timers
  setImmediate(() => {
    advanceRetry(initializeCount);
  });
  return Promise.reject(new Error(errorMessage));
 });

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);

 expect(initializeMock).toHaveBeenCalledTimes(1);
 expect(initializeMock).toHaveBeenCalledWith(licenseCheckResults);
 downstreamPlugin.status.red.mockImplementation(message => {
  isRed = true;
  expect(message).toBe(errorMessage);
 });
});

test(`sets downstream plugin's status to green when initialize resolves after rejecting 10 times`, (done) => {
 jest.useFakeTimers();

 const pluginId = 'foo-plugin';
 const errorMessage = 'the error message';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green');
 const licenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(licenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);

 let initializeCount = 0;
 const initializeMock = jest.fn().mockImplementation(() => {
  ++initializeCount;

  // everytime this is called, we have to wait for a new promise to be resolved
  // allowing the Promise the we return below to run, and then advance the timers
  setImmediate(() => {
    advanceRetry(initializeCount);
  });

  if (initializeCount >= 10) {
    return Promise.resolve();
  }

  return Promise.reject(new Error(errorMessage));
 });

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);
```

```
 expect(initializeMock).toHaveBeenCalledTimes(1);
 expect(initializeMock).toHaveBeenCalledWith(licenseCheckResults);
 downstreamPlugin.status.red.mockImplementation(message => {
   expect(initializeCount).toBeLessThan(10);
   expect(message).toBe(errorMessage);
 });
 downstreamPlugin.status.green.mockImplementation(message => {
   expect(initializeCount).toBe(10);
   expect(message).toBe('Ready');
   done();
 });
});

test(`calls initialize twice when it gets a new license and the status is green`, (done) => {
 const pluginId = 'foo-plugin';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green');
 const firstLicenseCheckResults = Symbol();
 const secondLicenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(firstLicenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);
 const initializeMock = jest.fn().mockImplementation(() => Promise.resolve());

 let count = 0;
 downstreamPlugin.status.green.mockImplementation(message => {
   expect(message).toBe('Ready');
   ++count;
   if (count === 1) {
     mockFeature.mock.setLicenseCheckResults(secondLicenseCheckResults);
     mockFeature.mock.triggerLicenseChange();
   }
   if (count === 2) {
     expect(initializeMock).toHaveBeenCalledWith(firstLicenseCheckResults);
     expect(initializeMock).toHaveBeenCalledWith(secondLicenseCheckResults);
     expect(initializeMock).toHaveBeenCalledTimes(2);
     done();
   }
 });

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);
});

test(`doesn't call initialize twice when it gets a new license when the status isn't green`, (done) => {
 const pluginId = 'foo-plugin';
 const redMessage = 'the red message';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green');
 const firstLicenseCheckResults = Symbol();
```

```
 const secondLicenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(firstLicenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);
 const initializeMock = jest.fn().mockImplementation(() => Promise.resolve());

 downstreamPlugin.status.green.mockImplementation(message => {
   expect(message).toBe('Ready');
   mockXpackMainPlugin.mock.setStatus('red', redMessage);
   mockFeature.mock.setLicenseCheckResults(secondLicenseCheckResults);
   mockFeature.mock.triggerLicenseChange();
 });

 downstreamPlugin.status.red.mockImplementation(message => {
   expect(message).toBe(redMessage);
   expect(initializeMock).toHaveBeenCalledTimes(1);
   expect(initializeMock).toHaveBeenCalledWith(firstLicenseCheckResults);
   done();
 });

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);
});

test(`calls initialize twice when the status changes to green twice`, (done) => {
 const pluginId = 'foo-plugin';
 const { mockXpackMainPlugin, mockFeature } = createMockXpackMainPluginAndFeature(pluginId);
 mockXpackMainPlugin.mock.setStatus('green');
 const licenseCheckResults = Symbol();
 mockFeature.mock.setLicenseCheckResults(licenseCheckResults);
 const downstreamPlugin = createMockDownstreamPlugin(pluginId);
 const initializeMock = jest.fn().mockImplementation(() => Promise.resolve());

 let count = 0;
 downstreamPlugin.status.green.mockImplementation(message => {
   expect(message).toBe('Ready');
   ++count;
   if (count === 1) {
     mockXpackMainPlugin.mock.setStatus('green');
   }
   if (count === 2) {
     expect(initializeMock).toHaveBeenCalledWith(licenseCheckResults);
     expect(initializeMock).toHaveBeenCalledTimes(2);
     done();
   }
 });

 watchStatusAndLicenseToInitialize(mockXpackMainPlugin, downstreamPlugin, initializeMock);
});
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
import * as React from 'react';
import { NoDataLayout } from '../components/layouts/no_data';

export const InvalidLicensePage: React.SFC<any> = () => (
 <NoDataLayout title="Expired license" actionSection={[]}>
   <p>
     Your current license is expired. Enrolled Beats will continue to work, but you need a valid
     license to access the Beats Management UI.
   </p>
 </NoDataLayout>
);
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import { checkLicenseFactory } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 let checkLicense;

 beforeEach(() => {
   mockLicenseInfo = {};
   checkLicense = checkLicenseFactory({
     getAll: () => [{
       id: 'test',
       name: 'Test Export Type',
       jobType: 'testJobType'
     }]
   });
 });

 describe('license information is not available', () => {
   beforeEach(() => mockLicenseInfo.isAvailable = () => false);

   it('should set management.showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).management.showLinks).to.be(true);
   });
```

```
  it('should set test.showLinks to true', () => {
    expect(checkLicense(mockLicenseInfo).test.showLinks).to.be(true);
  });

  it('should set management.enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).management.enableLinks).to.be(false);
  });

  it('should set test.enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).test.enableLinks).to.be(false);
  });

  it ('should set management.jobTypes to undefined', () => {
    expect(checkLicense(mockLicenseInfo).management.jobTypes).to.be(undefined);
  });
});

describe('license information is available', () => {
  beforeEach(() => {
    mockLicenseInfo.isAvailable = () => true;
    set(mockLicenseInfo, 'license.getType', () => 'basic');
  });

  describe('& license is trial, standard, gold, platinum', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

    describe('& license is active', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

      it ('should set management.showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).management.showLinks).to.be(true);
      });

      it ('should set test.showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).test.showLinks).to.be(true);
      });

      it ('should set management.enableLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).management.enableLinks).to.be(true);
      });

      it ('should set test.enableLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).test.enableLinks).to.be(true);
      });

      it ('should set management.jobTypes to contain testJobType', () => {
        expect(checkLicense(mockLicenseInfo).management.jobTypes).to.contain('testJobType');
```

```
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it ('should set management.showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).management.showLinks).to.be(true);
    });

    it ('should set test.showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).test.showLinks).to.be(true);
    });

    it ('should set management.enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).management.enableLinks).to.be(false);
    });

    it ('should set test.enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).test.enableLinks).to.be(false);
    });

    it ('should set management.jobTypes to undefined', () => {
      expect(checkLicense(mockLicenseInfo).management.jobTypes).to.be(undefined);
    });
  });
});

describe('& license is basic', () => {
  beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => false));

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it ('should set management.showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).management.showLinks).to.be(false);
    });

    it ('should set test.showLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).test.showLinks).to.be(false);
    });

    it('should set management.jobTypes to an empty array', () => {
      expect(checkLicense(mockLicenseInfo).management.jobTypes).to.be.an(Array);
      expect(checkLicense(mockLicenseInfo).management.jobTypes).to.have.length(0);
    });
  });
```

```
    describe('& license is expired', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

      it ('should set management.showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).management.showLinks).to.be(true);
      });

      it ('should set test.showLinks to false', () => {
        expect(checkLicense(mockLicenseInfo).test.showLinks).to.be(false);
      });

      it ('should set management.jobTypes to undefined', () => {
        expect(checkLicense(mockLicenseInfo).management.jobTypes).to.be(undefined);
      });
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

const messages = {
 getUnavailable: () => {
   return 'You cannot use Reporting because license information is not available at this time.';
 },
 getExpired: (license) => {
   return `You cannot use Reporting because your ${license.getType()} license has expired.`;
 }
};

const makeManagementFeature = (exportTypes) => {
 return {
   id: 'management',
   checkLicense: (license) => {
     if (!license) {
       return {
         showLinks: true,
         enableLinks: false,
         message: messages.getUnavailable(),
       };
     }

     if (!license.isActive()) {
       return {
         showLinks: true,
```

```
        enableLinks: false,
        message: messages.getExpired(license),
      };
    }

    const validJobTypes = exportTypes
      .filter(exportType => license.isOneOf(exportType.validLicenses))
      .map(exportType => exportType.jobType);

    return {
      showLinks: validJobTypes.length > 0,
      enableLinks: validJobTypes.length > 0,
      jobTypes: validJobTypes
    };
  }
 };
};

const makeExportTypeFeature = (exportType) => {
 return {
  id: exportType.id,
  checkLicense: (license) => {
    if (!license) {
      return {
        showLinks: true,
        enableLinks: false,
        message: messages.getUnavailable(),
      };
    }

    if (!license.isOneOf(exportType.validLicenses)) {
      return {
        showLinks: false,
        enableLinks: false,
        message: `Your ${license.getType()} license does not support ${exportType.name} Reporting. Please upgrade
your license.`
      };
    }

    if (!license.isActive()) {
      return {
        showLinks: true,
        enableLinks: false,
        message: messages.getExpired(license),
      };
    }

    return {
```

```
      showLinks: true,
      enableLinks: true,
    };
  }
 };
};

export function checkLicenseFactory(exportTypesRegistry) {
 return function checkLicense(xpackLicenseInfo) {
   const license = xpackLicenseInfo === null || !xpackLicenseInfo.isAvailable() ? null : xpackLicenseInfo.license;

   const exportTypes = Array.from(exportTypesRegistry.getAll());
   const reportingFeatures = [...exportTypes.map(makeExportTypeFeature),
makeManagementFeature(exportTypes)];

   return reportingFeatures.reduce((result, feature) => {
     result[feature.id] = feature.checkLicense(license);
     return result;
   }, {});
 };
}
```

This Font Software is licensed under the SIL Open Font License,
Version 1.1.

This license is copied below, and is also available with a FAQ at:
http://scripts.sil.org/OFL

SIL OPEN FONT LICENSE Version 1.1 - 26 February 2007

PREAMBLE
The goals of the Open Font License (OFL) are to stimulate worldwide
development of collaborative font projects, to support the font
creation efforts of academic and linguistic communities, and to
provide a free and open framework in which fonts may be shared and
improved in partnership with others.

The OFL allows the licensed fonts to be used, studied, modified and
redistributed freely as long as they are not sold by themselves. The
fonts, including any derivative works, can be bundled, embedded,
redistributed and/or sold with any software provided that any reserved
names are not used by derivative works. The fonts and derivatives,
however, cannot be released under any other type of license. The
requirement for fonts to remain under this license does not apply to
any document created using the fonts or their derivatives.

DEFINITIONS

"Font Software" refers to the set of files released by the Copyright Holder(s) under this license and clearly marked as such. This may include source files, build scripts and documentation.

"Reserved Font Name" refers to any names specified as such after the copyright statement(s).

"Original Version" refers to the collection of Font Software components as distributed by the Copyright Holder(s).

"Modified Version" refers to any derivative made by adding to, deleting, or substituting -- in part or in whole -- any of the components of the Original Version, by changing formats or by porting the Font Software to a new environment.

"Author" refers to any designer, engineer, programmer, technical writer or other person who contributed to the Font Software.

PERMISSION & CONDITIONS
Permission is hereby granted, free of charge, to any person obtaining a copy of the Font Software, to use, study, copy, merge, embed, modify, redistribute, and sell modified and unmodified copies of the Font Software, subject to the following conditions:

1) Neither the Font Software nor any of its individual components, in Original or Modified Versions, may be sold by itself.

2) Original or Modified Versions of the Font Software may be bundled, redistributed and/or sold with any software, provided that each copy contains the above copyright notice and this license. These can be included either as stand-alone text files, human-readable headers or in the appropriate machine-readable metadata fields within text or binary files as long as those fields can be easily viewed by the user.

3) No Modified Version of the Font Software may use the Reserved Font Name(s) unless explicit written permission is granted by the corresponding Copyright Holder. This restriction only applies to the primary font name as presented to the users.

4) The name(s) of the Copyright Holder(s) or the Author(s) of the Font Software shall not be used to promote, endorse or advertise any Modified Version, except to acknowledge the contribution(s) of the Copyright Holder(s) and the Author(s) or with their explicit written permission.

5) The Font Software, modified or unmodified, in part or in whole, must be distributed entirely under this license, and must not be distributed under any other license. The requirement for fonts to

remain under this license does not apply to any document created using
the Font Software.

TERMINATION
This license becomes null and void if any of the above conditions are
not met.

DISCLAIMER
THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

```
{
 "xpack.license.post": {
   "data_autocomplete_rules": {
     "licenses": []
   }
 }
}
{
 "xpack.license.post_start_trial": {
   "url_params": {
     "type": "\"trial\"",
     "acknowledge": "__flag__"
   },
   "methods": [
     "POST"
   ],
   "patterns": [
     "_xpack/license/start_trial"
   ],
   "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
{
 "xpack.license.get": {
   "url_params": {
     "local": "__flag__"
   },
   "methods": [
     "GET"
   ],
   "patterns": [
```

```
    "_xpack/license"
  ],
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
{
 "xpack.license.delete": {
  "methods": [
    "DELETE"
  ],
  "patterns": [
    "_xpack/license"
  ],
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
{
 "xpack.license.get_basic_status": {
  "methods": [
    "GET"
  ],
  "patterns": [
    "_xpack/license/basic_status"
  ],
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
{
 "xpack.license.post": {
  "url_params": {
    "acknowledge": "__flag__"
  },
  "methods": [
    "PUT",
    "POST"
  ],
  "patterns": [
    "_xpack/license"
  ],
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
{
 "xpack.license.post_start_basic": {
  "url_params": {
    "acknowledge": "__flag__"
  },
  "methods": [
```

```
    "POST"
  ],
  "patterns": [
    "_xpack/license/start_basic"
  ],
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
{
 "xpack.license.get_trial_status": {
  "methods": [
    "GET"
  ],
  "patterns": [
    "_xpack/license/trial_status"
  ],
  "documentation": "https://www.elastic.co/guide/en/x-pack/current/license-management.html"
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { startBasic } from '../../../lib/start_basic';
import { wrapEsError } from '../../../lib/wrap_es_error';

export function registerStartBasicRoute(server) {
 const xpackInfo = server.plugins.xpack_main.info;
 server.route({
  path: '/api/license/start_basic',
  method: 'POST',
  handler: (request, reply) => {
    return startBasic(request, xpackInfo)
      .then(reply, e => reply(wrapEsError(e)));
  }
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { canStartTrial, startTrial } from '../../../lib/start_trial';
import { wrapEsError } from '../../../lib/wrap_es_error';
```

```
export function registerStartTrialRoutes(server) {
 const xpackInfo = server.plugins.xpack_main.info;
 server.route({
   path: '/api/license/start_trial',
   method: 'GET',
   handler: (request, reply) => {
     return canStartTrial(request)
       .then(reply, e => reply(wrapEsError(e)));
   }
 });
 server.route({
   path: '/api/license/start_trial',
   method: 'POST',
   handler: (request, reply) => {
     return startTrial(request, xpackInfo)
       .then(reply, e => reply(wrapEsError(e)));
   }
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { putLicense } from '../../../lib/license';
import { wrapEsError } from '../../../lib/wrap_es_error';

export function registerLicenseRoute(server) {
 const xpackInfo = server.plugins.xpack_main.info;
 server.route({
   path: '/api/license',
   method: 'PUT',
   handler: (request, reply) => {
     return putLicense(request, xpackInfo)
       .then(reply, e => reply(wrapEsError(e)));
   }
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { registerLicenseRoute } from './register_license_route';
export { registerStartBasicRoute } from './register_start_basic_route';
export { registerStartTrialRoutes } from './register_start_trial_routes';
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

const getStartBasicPath = (acknowledge) => `/_xpack/license/start_basic${ acknowledge ? '?acknowledge=true' :
"}`;


export async function startBasic(req, xpackInfo) {
 const { acknowledge } = req.query;
 const { callWithRequest } = req.server.plugins.elasticsearch.getCluster('admin');
 const options = {
   method: 'POST',
   path: getStartBasicPath(acknowledge)
 };
 try {
   const response = await callWithRequest(req, 'transport.request', options);
   /*eslint camelcase: 0*/
   const { basic_was_started } = response;
   if (basic_was_started) {
     await xpackInfo.refreshNow();
   }
   return response;
 } catch (error) {
   return error.body;
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export async function canStartTrial(req) {
 const { callWithRequest } = req.server.plugins.elasticsearch.getCluster('admin');
 const options = {
   method: 'GET',
   path: '/_xpack/license/trial_status'
 };
 try {
   const response = await callWithRequest(req, 'transport.request', options);
   const { eligible_to_start_trial } = response;
   return eligible_to_start_trial;
 } catch (error) {
   return error.body;
 }
```

```
}
export async function startTrial(req, xpackInfo) {
 const { callWithRequest } = req.server.plugins.elasticsearch.getCluster('admin');
 const options = {
   method: 'POST',
   path: '/_xpack/license/start_trial?acknowledge=true'
 };
 try {
   /*eslint camelcase: 0*/
   const response = await callWithRequest(req, 'transport.request', options);
   const { trial_was_started } = response;
   if (trial_was_started) {
     await xpackInfo.refreshNow();
   }
   return response;
 } catch (error) {
   return error.body;
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import Boom from 'boom';

/**
* Wraps ES errors into a Boom error response and returns it
* This also handles the permissions issue gracefully
*
* @param err Object ES error
* @return Object Boom error response
*/
export function wrapEsError(err) {
 const statusCode = err.statusCode;
 if (statusCode === 403) {
   return Boom.forbidden('Insufficient user permissions for adding a license.');
 }
 return Boom.boomify(err, err.statusCode);
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

const getLicensePath = (acknowledge) => `/_xpack/license${ acknowledge ? '?acknowledge=true' : ''}`;
```

```
export async function putLicense(req, xpackInfo) {
 const { acknowledge } = req.query;
 const { callWithRequest } = req.server.plugins.elasticsearch.getCluster('admin');
 const options = {
   method: 'POST',
   path: getLicensePath(acknowledge),
   body: req.payload
 };
 try {
   const response = await callWithRequest(req, 'transport.request', options);
   const { acknowledged, license_status: licenseStatus } = response;
   if (acknowledged && licenseStatus === 'valid') {
     await xpackInfo.refreshNow();
   }
   return response;
 } catch (error) {
   return error.body;
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { LicenseStatus } from '../public/sections/license_dashboard/license_status';
import { createMockLicense, getComponent } from './util';

describe('LicenseStatus component', () => {
 test('should display normally when license is active', () => {
   const rendered = getComponent(
     {
       license: createMockLicense('gold')
     },
     LicenseStatus
   );
   expect(rendered.html()).toMatchSnapshot();
 });
 test('should display display warning is expired', () => {
   const rendered = getComponent(
     {
       license: createMockLicense('platinum', 0)
     },
     LicenseStatus
   );
   expect(rendered.html()).toMatchSnapshot();
 });
```

```
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { RequestTrialExtension } from '../public/sections/license_dashboard/request_trial_extension';
import { createMockLicense, getComponent } from './util';

describe('RequestTrialExtension component', () => {
 test('should not display when license is active and trial has not been used', () => {
   const rendered = getComponent(
     {
       trialStatus: {
         canStartTrial: true
       },
       license: createMockLicense('trial')
     },
     RequestTrialExtension
   );
   const html = rendered.html();
   expect(html).toBeNull();
 });
 test('should display when license is active and trial has been used', () => {
   const rendered = getComponent(
     {
       trialStatus: {
         canStartTrial: false
       },
       license: createMockLicense('trial')
     },
     RequestTrialExtension
   );
   const html = rendered.html();
   expect(html).not.toBeNull();
   expect(html).toMatchSnapshot();
 });
 test('should not display when license is not active and trial has not been used', () => {
   const rendered = getComponent(
     {
       trialStatus: {
         canStartTrial: true
       },
       license: createMockLicense('trial', 0)
     },
     RequestTrialExtension
   );
```

```
  const html = rendered.html();
  expect(html).toBeNull();
 });
 test('should display when license is not active and trial has been used', () => {
  const rendered = getComponent(
    {
      trialStatus: {
        canStartTrial: false
      },
      license: createMockLicense('trial', 0)
    },
    RequestTrialExtension
  );
  const html = rendered.html();
  expect(html).not.toBeNull();
  expect(html).toMatchSnapshot();
 });
 test('should display when platinum license is not active and trial has been used', () => {
  const rendered = getComponent(
    {
      trialStatus: {
        canStartTrial: false
      },
      license: createMockLicense('platinum', 0)
    },
    RequestTrialExtension
  );
  const html = rendered.html();
  expect(html).not.toBeNull();
  expect(html).toMatchSnapshot();
 });
 test('should not display when platinum license is active and trial has been used', () => {
  const rendered = getComponent(
    {
      trialStatus: {
        canStartTrial: false
      },
      license: createMockLicense('platinum')
    },
    RequestTrialExtension
  );
  const html = rendered.html();
  expect(html).toBeNull();
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
```

```
* you may not use this file except in compliance with the Elastic License.
*/

import { mount } from 'enzyme';
import React from 'react';
import { Provider } from 'react-redux';
import { uploadLicense } from '../public/store/actions/upload_license';
import { licenseManagementStore } from '../public/store/store';
import { UploadLicense } from '../public/sections/upload_license';
import { BASE_PATH } from '../common/constants';
import {
 UPLOAD_LICENSE_EXPIRED,
 UPLOAD_LICENSE_REQUIRES_ACK,
 UPLOAD_LICENSE_SUCCESS,
 UPLOAD_LICENSE_TLS_NOT_ENABLED,
 UPLOAD_LICENSE_INVALID,
} from './api_responses';

import sinon from 'sinon';
window.location.reload = () => {};
let server = null;
let store = null;
let component = null;
const services = {
 kbnUrl: {
   change: jest.fn()
 },
 autoLogout: () => {},
 xPackInfo: {
   refresh: jest.fn(),
   get: () => {
     return { license: { type: 'basic' } };
   }
 }
};

describe('UploadLicense', () => {
 beforeEach(() => {
   server = sinon.fakeServer.create();
   server.respondImmediately = true;
   store = licenseManagementStore({}, services);
   component = (
     <Provider store={store}>
       <UploadLicense />
     </Provider>
   );
 });
 afterEach(() => {
```

```
  server.restore();
  services.xPackInfo.refresh.mockReset();
  services.kbnUrl.change.mockReset();
});
it('should display an error when submitting invalid JSON', async () => {
  const rendered = mount(component);
  store.dispatch(uploadLicense('INVALID', 'trial'));
  rendered.update();
  expect(rendered).toMatchSnapshot();
});
it('should display an error when ES says license is invalid', async () => {
  const rendered = mount(component);
  const invalidLicense = JSON.stringify({ license: { type: 'basic' } });
  server.respond(UPLOAD_LICENSE_INVALID);
  await uploadLicense(invalidLicense)(store.dispatch, null, services);
  rendered.update();
  expect(rendered).toMatchSnapshot();
});
it('should display an error when ES says license is expired', async () => {
  const rendered = mount(component);
  const invalidLicense = JSON.stringify({ license: { type: 'basic' } });
  server.respond(UPLOAD_LICENSE_EXPIRED);
  await uploadLicense(invalidLicense)(store.dispatch, null, services);
  rendered.update();
  expect(rendered).toMatchSnapshot();
});
it('should display a modal when license requires acknowledgement', async () => {
  const unacknowledgedLicense = JSON.stringify({
    license: { type: 'basic' }
  });
  server.respond(UPLOAD_LICENSE_REQUIRES_ACK);
  await uploadLicense(unacknowledgedLicense, 'trial')(
    store.dispatch,
    null,
    services
  );
  const rendered = mount(component);
  expect(rendered).toMatchSnapshot();
});
it('should refresh xpack info and navigate to BASE_PATH when ES accepts new license', async () => {
  const validLicense = JSON.stringify({ license: { type: 'basic' } });
  server.respond(UPLOAD_LICENSE_SUCCESS);
  await uploadLicense(validLicense)(store.dispatch, null, services);
  expect(services.xPackInfo.refresh).toHaveBeenCalled();
  expect(services.kbnUrl.change).toHaveBeenCalledWith(BASE_PATH);
});
it('should display error when ES returns error', async () => {
  const rendered = mount(component);
```

```
  const license = JSON.stringify({ license: { type: 'basic' } });
  server.respond(UPLOAD_LICENSE_TLS_NOT_ENABLED);
  await uploadLicense(license)(store.dispatch, null, services);
  rendered.update();
  expect(rendered).toMatchSnapshot();
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { RevertToBasic } from '../public/sections/license_dashboard/revert_to_basic';
import { createMockLicense, getComponent } from './util';

describe('RevertToBasic component', () => {
 test('should display when trial is active', () => {
  const rendered = getComponent(
   {
    license: createMockLicense('trial')
   },
   RevertToBasic
  );
  expect(rendered.html()).toMatchSnapshot();
 });
 test('should display when license is expired', () => {
  const rendered = getComponent(
   {
    license: createMockLicense('platinum', 0)
   },
   RevertToBasic
  );
  expect(rendered.html()).toMatchSnapshot();
 });
 test('should display when license is about to expire', () => {
  // ten days from now
  const imminentExpirationTime = new Date().getTime() + (10 * 24 * 60 * 60 * 1000);
  const rendered = getComponent(
   {
    license: createMockLicense('platinum', imminentExpirationTime)
   },
   RevertToBasic
  );
  expect(rendered.html()).toMatchSnapshot();
 });
 test('should not display for active basic license', () => {
  const rendered = getComponent(
```

```
    {
      license: createMockLicense('basic')
    },
    RevertToBasic
  );
  expect(rendered.html()).toBeNull();
});
test('should not display for active gold license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('gold')
    },
    RevertToBasic
  );
  expect(rendered.html()).toBeNull();
});
test('should not display for active platinum license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('platinum')
    },
    RevertToBasic
  );
  expect(rendered.html()).toBeNull();
});
});
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`LicenseStatus component should display display warning is expired 1`] = `"<div><div
class=\\"euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceAround euiFlexGroup--
directionRow euiFlexGroup--responsive\\"><div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><div
class=\\"euiFlexGroup euiFlexGroup--gutterSmall euiFlexGroup--alignItemsCenter euiFlexGroup--directionRow
euiFlexGroup--responsive\\"><div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><svg class=\\"euiIcon
euiIcon--medium euiIcon--danger\\" focusable=\\"false\\" xmlns=\\"http://www.w3.org/2000/svg\\" width=\\"16\\"
height=\\"16\\" viewBox=\\"0 0 16 16\\"><g fill-rule=\\"evenodd\\"><path d=\\"M7.5 2.236L1.618 14h11.764L7.5
2.236zm.894-.447l5.882 11.764A1 1 0 0 1 13.382 15H1.618a1 1 0 0 1-.894-1.447L6.606 1.789a1 1 0 0 1 1.788
0z\\"></path><path d=\\"M7 6h1v5H7zM7 12h1v1H7z\\"></path></g></svg></div><div class=\\"euiFlexItem
euiFlexItem--flexGrowZero\\"><h2 data-test-subj=\\"licenseText\\" class=\\"euiTitle euiTitle--large\\">Your
Platinum license has expired</h2></div></div></div></div><div class=\\"euiFlexGroup euiFlexGroup--
gutterLarge euiFlexGroup--justifyContentSpaceAround euiFlexGroup--directionRow euiFlexGroup--
responsive\\"><div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><span data-test-
subj=\\"licenseSubText\\"><div class=\\"euiText\\"><div class=\\"euiTextColor euiTextColor--subdued\\">Your
license expired on <strong></strong></div></div></span></div></div><div class=\\"euiSpacer euiSpacer--
l\\"></div></div>"`;

exports[`LicenseStatus component should display normally when license is active 1`] = `"<div><div
class=\\"euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceAround euiFlexGroup--
directionRow euiFlexGroup--responsive\\"><div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><div
```

class=\\"euiFlexGroup euiFlexGroup--gutterSmall euiFlexGroup--alignItemsCenter euiFlexGroup--directionRow euiFlexGroup--responsive\\"><div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><svg class=\\"euiIcon euiIcon--medium euiIcon--success\\" focusable=\\"false\\" width=\\"16\\" height=\\"16\\" viewBox=\\"0 0 16 16\\" xmlns=\\"http://www.w3.org/2000/svg\\"><path d=\\"M8 16A8 8 0 1 1 8 0a8 8 0 0 1 0 16zm-1.125-5a.377.377 0 0 0 .266-.11l5.25-5.25a.376.376 0 0 0-.531-.53l-4.985 4.985L4.141 7.36a.376.376 0 0 0-.531.53l3 3c.073.073.17.11.265.11z\\" fill-rule=\\"evenodd\\"></path></svg> <div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><h2 data-test-subj=\\"licenseText\\" class=\\"euiTitle euiTitle--large\\">Your Gold license is active</h2> <div class=\\"euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceAround euiFlexGroup--directionRow euiFlexGroup--responsive\\"><div class=\\"euiFlexItem euiFlexItem--flexGrowZero\\"><div class=\\"euiText\\"><div class=\\"euiTextColor euiTextColor--subdued\\">Your license will expire on October 12, 2099 7:00 PM EST <div class=\\"euiSpacer euiSpacer--l\\"> "`;
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`RequestTrialExtension component should display when license is active and trial has been used 1`] = `"<div class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\">Extend your trial<div class=\\"euiText euiText--small euiCard__description\\"><p>If youd like to continuing using security, machine learning, and our other awesome Platinum features, request an extension now.</p></div>Extend trial</div></div>"`;

exports[`RequestTrialExtension component should display when license is not active and trial has been used 1`] = `"<div class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\">Extend your trial<div class=\\"euiText euiText--small euiCard__description\\"><p>If youd like to continuing using security, machine learning, and our other awesome Platinum features, request an extension now.</p></div>Extend trial</div></div>"`;

exports[`RequestTrialExtension component should display when platinum license is not active and trial has been used 1`] = `"<div class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\">Extend your trial<div class=\\"euiText euiText--small euiCard__description\\"><p>If youd like to continuing using security, machine learning, and our other awesome Platinum features, request an extension now.</p></div>Extend trial</div></div>"`;
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`AddLicense component when license is active should display correct verbiage 1`] = `"<div class=\\"euiCard euiCard--centerAligned\\">Update your license<div class=\\"euiText euiText--small euiCard__description\\"><p>If you already have a new license, upload it now.</p></div>Update license</div>"`;

exports[`AddLicense component when license is expired should display with correct verbiage 1`] = `"<div class=\\"euiCard euiCard--centerAligned\\">Update your license<div class=\\"euiText euiText--small euiCard__description\\"><p>If you already have a new license, upload it now.</p></div>Update license</div>"`;
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`StartTrial component when trial is allowed display for basic license 1`] = `"<div class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\">Start a 30-day trial<div class=\\"euiText euiText--small euiCard__description\\"><p>Experience what security, machine learning, and all our other Platinum features have to offer.</p></div><button class=\\"euiButton euiButton--primary\\" type=\\"button\\" data-test-subj=\\"startTrialButton\\">Start trial</button></div></div>"`;

exports[`StartTrial component when trial is allowed should display for expired platinum license 1`] = `"<div class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\">Start a 30-day trial<div class=\\"euiText euiText--small euiCard__description\\"><p>Experience what security, machine learning, and all our other Platinum features have to offer.</p></div><button class=\\"euiButton euiButton--primary\\" type=\\"button\\" data-test-subj=\\"startTrialButton\\">Start trial</button></div></div>"`;

exports[`StartTrial component when trial is allowed should display for gold license 1`] = `"<div class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\"><span class=\\"euiCard__content\\"><span class=\\"euiTitle euiTitle--medium euiCard__title\\">Start a 30-day trial<div class=\\"euiText euiText--small euiCard__description\\"> Experience what security, machine learning, and all our other Platinum features have to offer. <span class=\\"euiCard__footer\\"><button class=\\"euiButton

```
euiButton--primary\\" type=\\"button\\" data-test-subj=\\"startTrialButton\\"><span
class=\\"euiButton__content\\"><span class=\\"euiButton__text\\">Start
trial</span></span></button></span></div></div>"`;
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`UploadLicense should display a modal when license requires acknowledgement 1`] = `
<Provider
 store={
   Object {
     "dispatch": [Function],
     "getState": [Function],
     "replaceReducer": [Function],
     "subscribe": [Function],
     Symbol(observable): [Function],
   }
 }
>
 <Connect(UploadLicense)>
   <UploadLicense
     addUploadErrorMessage={[Function]}
     applying={false}
     currentLicenseType=""
     errorMessage=""
     isInvalid={false}
     messages={
       Array [
         "Some functionality will be lost if you replace your
       TRIAL license with a
       BASIC license.  Review the list of features below.",
         Array [
           "Watcher will be disabled",
         ],
       ]
     }
     needsAcknowledgement={true}
     uploadLicense={[Function]}
     uploadLicenseStatus={[Function]}
   >
     <EuiTitle
       className="eui-textCenter"
       size="l"
     >
       <h1
         className="euiTitle euiTitle--large eui-textCenter"
       >
         Upload your license
       </h1>
     </EuiTitle>
```

```
    <EuiSpacer
      size="l"
    >
      <div
        className="euiSpacer euiSpacer--l"
      />
    </EuiSpacer>
    <EuiPageContent
      horizontalPosition="center"
      panelPaddingSize="l"
      verticalPosition="center"
    >
      <EuiPanel
        className="euiPageContent euiPageContent--verticalCenter euiPageContent--horizontalCenter"
        grow={true}
        hasShadow={false}
        paddingSize="l"
      >
        <div
          className="euiPanel euiPanel--paddingLarge euiPageContent euiPageContent--verticalCenter
euiPageContent--horizontalCenter"
        >
          <EuiPageContentBody>
            <div
              className="euiPageContentBody"
            >
              <EuiOverlayMask>
                <EuiConfirmModal
                  buttonColor="primary"
                  cancelButtonText="Cancel"
                  confirmButtonText="Confirm"
                  onCancel={[Function]}
                  onConfirm={[Function]}
                  title="Confirm License Upload"
                >
                  <EuiModal
                    className="euiModal--confirmation"
                    maxWidth={true}
                    onClose={[Function]}
                  >
                    <Component
                      focusTrapOptions={
                        Object {
                          "fallbackFocus": [Function],
                          "initialFocus": undefined,
                        }
                      }
                    >
```

```
<div>
  <div
    className="euiModal euiModal--maxWidth-default euiModal--confirmation"
    onKeyDown={[Function]}
    tabIndex={0}
  >
    <EuiButtonIcon
      aria-label="Closes this modal window"
      className="euiModal__closeIcon"
      color="text"
      iconSize="m"
      iconType="cross"
      onClick={[Function]}
      type="button"
    >
      <button
        aria-label="Closes this modal window"
        className="euiButtonIcon euiButtonIcon--text euiModal__closeIcon"
        onClick={[Function]}
        type="button"
      >
        <EuiIcon
          aria-hidden="true"
          className="euiButtonIcon__icon"
          size="m"
          type="cross"
        >
          <cross
            aria-hidden="true"
            className="euiIcon euiIcon--medium euiButtonIcon__icon"
            focusable="false"
            height="16"
            style={
              Object {
                "fill": undefined,
              }
            }
            viewBox="0 0 16 16"
            width="16"
            xmlns="http://www.w3.org/2000/svg"
            xmlnsXlink="http://www.w3.org/1999/xlink"
          >
            <svg
              aria-hidden="true"
              className="euiIcon euiIcon--medium euiButtonIcon__icon"
              focusable="false"
              height="16"
              style={
```

```
                    Object {
                      "fill": undefined,
                    }
                  }
                  viewBox="0 0 16 16"
                  width="16"
                  xmlns="http://www.w3.org/2000/svg"
                  xmlnsXlink="http://www.w3.org/1999/xlink"
                >
                  <defs>
                    <path
                      d="M7.293 8l-4.147 4.146a.5.5 0 0 0 .708.708L8 8.707l4.146 4.147a.5.5 0 0 0 .708-
.708L8.707 8l4.147-4.146a.5.5 0 0 0-.708-.708L8 7.293 3.854 3.146a.5.5 0 1 0-.708.708L7.293 8z"
                      id="cross-a"
                    />
                  </defs>
                  <use
                    fillRule="nonzero"
                    xlinkHref="#cross-a"
                  />
                </svg>
              </cross>
            </EuiIcon>
          </button>
        </EuiButtonIcon>
        <div
          className="euiModal__flex"
        >
          <EuiModalHeader>
            <div
              className="euiModalHeader"
            >
              <EuiModalHeaderTitle
                data-test-subj="confirmModalTitleText"
              >
                <div
                  className="euiModalHeader__title"
                  data-test-subj="confirmModalTitleText"
                >
                  Confirm License Upload
                </div>
              </EuiModalHeaderTitle>
            </div>
          </EuiModalHeader>
          <EuiModalBody>
            <div
              className="euiModalBody"
            >
```

```
                <EuiText
                  data-test-subj="confirmModalBodyText"
                  grow={true}
                >
                  <div
                    className="euiText"
                    data-test-subj="confirmModalBodyText"
                  >
                    <div>
                      <EuiText
                        grow={true}
                      >
                        <div
                          className="euiText"
                        >
                          Some functionality will be lost if you replace your
TRIAL license with a
BASIC license.  Review the list of features below.
                        </div>
                      </EuiText>
                      <EuiText
                        grow={true}
                      >
                        <div
                          className="euiText"
                        >
                          <ul>
                            <li
                              key="Watcher will be disabled"
                            >
                              Watcher will be disabled
                            </li>
                          </ul>
                        </div>
                      </EuiText>
                    </div>
                  </div>
                </EuiText>
              </div>
            </EuiModalBody>
            <EuiModalFooter>
              <div
                className="euiModalFooter"
              >
                <EuiButtonEmpty
                  buttonRef={[Function]}
                  color="primary"
                  data-test-subj="confirmModalCancelButton"
```

```
    iconSide="left"
    onClick={[Function]}
    type="button"
  >
    <button
      className="euiButtonEmpty euiButtonEmpty--primary"
      data-test-subj="confirmModalCancelButton"
      onClick={[Function]}
      type="button"
    >
      <span
        className="euiButtonEmpty__content"
      >
        <span
          className="euiButtonEmpty__text"
        >
          Cancel
        </span>
      </span>
    </button>
  </EuiButtonEmpty>
  <EuiButton
    buttonRef={[Function]}
    color="primary"
    data-test-subj="confirmModalConfirmButton"
    fill={true}
    iconSide="left"
    onClick={[Function]}
    type="button"
  >
    <button
      className="euiButton euiButton--primary euiButton--fill"
      data-test-subj="confirmModalConfirmButton"
      onClick={[Function]}
      type="button"
    >
      <span
        className="euiButton__content"
      >
        <span
          className="euiButton__text"
        >
          Confirm
        </span>
      </span>
    </button>
  </EuiButton>
</div>
```

```
            </EuiModalFooter>
          </div>
        </div>
      </div>
    </Component>
  </EuiModal>
</EuiConfirmModal>
</EuiOverlayMask>
<EuiText
  grow={true}
>
  <div
    className="euiText"
  >
    <p>
      Your license key is a JSON file with a signature attached.
    </p>
    <p>
      Uploading a license will replace your current

      <strong />
       license.
    </p>
  </div>
</EuiText>
<EuiSpacer
  size="l"
>
  <div
    className="euiSpacer euiSpacer--l"
  />
</EuiSpacer>
<EuiForm
  error={null}
  isInvalid={false}
>
  <div
    className="euiForm"
  >
    <EuiText
      grow={true}
    >
      <div
        className="euiText"
      >
        <EuiFilePicker
          compressed={false}
          id="licenseFile"
```

```
  initialPromptText="Select or drag your license file"
  onChange={[Function]}
>
  <div
    className="euiFilePicker"
  >
    <div
      className="euiFilePicker__wrap"
    >
      <input
        className="euiFilePicker__input"
        id="licenseFile"
        onChange={[Function]}
        onDragLeave={[Function]}
        onDragOver={[Function]}
        onDrop={[Function]}
        type="file"
      />
      <div
        className="euiFilePicker__prompt"
      >
        <EuiIcon
          aria-hidden="true"
          className="euiFilePicker__icon"
          size="l"
          type="importAction"
        >
          <importAction
            aria-hidden="true"
            className="euiIcon euiIcon--large euiFilePicker__icon"
            focusable="false"
            height="16"
            style={
              Object {
                "fill": undefined,
              }
            }
            viewBox="0 0 16 16"
            width="16"
            xmlns="http://www.w3.org/2000/svg"
          >
            <svg
              aria-hidden="true"
              className="euiIcon euiIcon--large euiFilePicker__icon"
              focusable="false"
              height="16"
              style={
                Object {
```

```
                "fill": undefined,
              }
            }
            viewBox="0 0 16 16"
            width="16"
            xmlns="http://www.w3.org/2000/svg"
          >
            <path
              d="M9 10.114l1.85-1.943a.52.52 0 0 1 .77 0c.214.228.214.6 0 .829l-1.95 2.05a1.552 1.552
0 0 1-2.31 0L5.41 9a.617.617 0 0 1 0-.829.52.52 0 0 1 .77 0L8 10.082V1.556C8 1.249 8.224 1 8.5
1s.5.249.5.556v8.558zM4.18 6a.993.993 0 0 0-.972.804l-1.189 6A.995.995 0 0 0 2.991 14h11.018a1 1 0 0 0 .972-
1.196l-1.19-6a.993.993 0 0 0-.97-.804H4.18zM6 5v1h5V5h1.825c.946 0 1.76.673 1.946 1.608l1.19 6A2 2 0 0 1
14.016 15H2.984a1.992 1.992 0 0 1-1.945-2.392l1.19-6C2.414 5.673 3.229 5 4.174 5H6z"
            />
          </svg>
        </importAction>
      </EuiIcon>
      <div
        className="euiFilePicker__promptText"
      >
        Select or drag your license file
      </div>
    </div>
  </div>
</div>
</EuiFilePicker>
</div>
</EuiText>
<EuiSpacer
  size="m"
>
  <div
    className="euiSpacer euiSpacer--m"
  />
</EuiSpacer>
<TelemetryOptIn />
<EuiSpacer
  size="m"
>
  <div
    className="euiSpacer euiSpacer--m"
  />
</EuiSpacer>
<EuiFlexGroup
  alignItems="stretch"
  component="div"
  direction="row"
  gutterSize="l"
```

```
            justifyContent="spaceBetween"
            responsive={true}
            wrap={false}
           >
            <div
             className="euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceBetween
euiFlexGroup--directionRow euiFlexGroup--responsive"
            >
             <EuiFlexItem
              component="div"
              grow={false}
             >
              <div
               className="euiFlexItem euiFlexItem--flexGrowZero"
              >
               <EuiButtonEmpty
                color="primary"
                href="#/management/elasticsearch/license_management/home"
                iconSide="left"
                type="button"
               >
                <a
                 className="euiButtonEmpty euiButtonEmpty--primary"
                 href="#/management/elasticsearch/license_management/home"
                >
                 <span
                  className="euiButtonEmpty__content"
                 >
                  <span
                   className="euiButtonEmpty__text"
                  >
                   Cancel
                  </span>
                 </span>
                </a>
               </EuiButtonEmpty>
              </div>
             </EuiFlexItem>
             <EuiFlexItem
              component="div"
              grow={false}
             >
              <div
               className="euiFlexItem euiFlexItem--flexGrowZero"
              >
               <EuiButton
                color="primary"
                data-test-subj="uploadLicenseButton"
```

```
                    fill={true}
                    iconSide="left"
                    isLoading={false}
                    onClick={[Function]}
                    type="button"
                  >
                    <button
                      className="euiButton euiButton--primary euiButton--fill"
                      data-test-subj="uploadLicenseButton"
                      onClick={[Function]}
                      type="button"
                    >
                      <span
                        className="euiButton__content"
                      >
                        <span
                          className="euiButton__text"
                        >
                          Upload
                        </span>
                      </span>
                    </button>
                  </EuiButton>
                </div>
              </EuiFlexItem>
            </div>
          </EuiFlexGroup>
        </div>
      </EuiForm>
    </div>
  </EuiPageContentBody>
 </div>
</EuiPanel>
</EuiPageContent>
</UploadLicense>
</Connect(UploadLicense)>
</Provider>
`;

exports[`UploadLicense should display an error when ES says license is expired 1`] = `
<Provider
 store={
  Object {
   "dispatch": [Function],
   "getState": [Function],
   "replaceReducer": [Function],
   "subscribe": [Function],
   Symbol(observable): [Function],
```

```
    }
  }
>
 <Connect(UploadLicense)>
   <UploadLicense
     addUploadErrorMessage={[Function]}
     applying={false}
     currentLicenseType=""
     errorMessage="The supplied license has expired."
     isInvalid={false}
     needsAcknowledgement={false}
     uploadLicense={[Function]}
     uploadLicenseStatus={[Function]}
   >
     <EuiTitle
       className="eui-textCenter"
       size="l"
     >
       <h1
         className="euiTitle euiTitle--large eui-textCenter"
       >
         Upload your license
       </h1>
     </EuiTitle>
     <EuiSpacer
       size="l"
     >
       <div
         className="euiSpacer euiSpacer--l"
       />
     </EuiSpacer>
     <EuiPageContent
       horizontalPosition="center"
       panelPaddingSize="l"
       verticalPosition="center"
     >
       <EuiPanel
         className="euiPageContent euiPageContent--verticalCenter euiPageContent--horizontalCenter"
         grow={true}
         hasShadow={false}
         paddingSize="l"
       >
         <div
           className="euiPanel euiPanel--paddingLarge euiPageContent euiPageContent--verticalCenter
euiPageContent--horizontalCenter"
         >
           <EuiPageContentBody>
             <div
```

```
  className="euiPageContentBody"
>
  <EuiText
    grow={true}
  >
    <div
      className="euiText"
    >
      <p>
        Your license key is a JSON file with a signature attached.
      </p>
      <p>
        Uploading a license will replace your current

        <strong />
         license.
      </p>
    </div>
  </EuiText>
  <EuiSpacer
    size="l"
  >
    <div
      className="euiSpacer euiSpacer--l"
    />
  </EuiSpacer>
  <EuiForm
    error={
      Array [
        "The supplied license has expired.",
      ]
    }
    isInvalid={true}
  >
    <div
      className="euiForm"
    >
      <EuiCallOut
        className="euiForm__errors"
        color="danger"
        size="m"
        title="Please address the errors in your form."
      >
        <div
          className="euiCallOut euiCallOut--danger euiForm__errors"
        >
          <div
            className="euiCallOutHeader"
```

```
    >
      <span
        className="euiCallOutHeader__title"
      >
        Please address the errors in your form.
      </span>
    </div>
    <EuiText
      grow={true}
      size="s"
    >
      <div
        className="euiText euiText--small"
      >
        <ul>
          <li
            className="euiForm__error"
            key="The supplied license has expired."
          >
            The supplied license has expired.
          </li>
        </ul>
      </div>
    </EuiText>
  </div>
</EuiCallOut>
<EuiText
  grow={true}
>
  <div
    className="euiText"
  >
    <EuiFilePicker
      compressed={false}
      id="licenseFile"
      initialPromptText="Select or drag your license file"
      onChange={[Function]}
    >
      <div
        className="euiFilePicker"
      >
        <div
          className="euiFilePicker__wrap"
        >
          <input
            className="euiFilePicker__input"
            id="licenseFile"
            onChange={[Function]}
```

```
                onDragLeave={[Function]}
                onDragOver={[Function]}
                onDrop={[Function]}
                type="file"
              />
              <div
                className="euiFilePicker__prompt"
              >
                <EuiIcon
                  aria-hidden="true"
                  className="euiFilePicker__icon"
                  size="l"
                  type="importAction"
                >
                  <importAction
                    aria-hidden="true"
                    className="euiIcon euiIcon--large euiFilePicker__icon"
                    focusable="false"
                    height="16"
                    style={
                      Object {
                        "fill": undefined,
                      }
                    }
                    viewBox="0 0 16 16"
                    width="16"
                    xmlns="http://www.w3.org/2000/svg"
                  >
                    <svg
                      aria-hidden="true"
                      className="euiIcon euiIcon--large euiFilePicker__icon"
                      focusable="false"
                      height="16"
                      style={
                        Object {
                          "fill": undefined,
                        }
                      }
                      viewBox="0 0 16 16"
                      width="16"
                      xmlns="http://www.w3.org/2000/svg"
                    >
                      <path
                        d="M9 10.114l1.85-1.943a.52.52 0 0 1 .77 0c.214.228.214.6 0 .829l-1.95 2.05a1.552 1.552
0 0 1-2.31 0L5.41 9a.617.617 0 0 1 0-.829.52.52 0 0 1 .77 0L8 10.082V1.556C8 1.249 8.224 1 8.5
1s.5.249.5.556v8.558zM4.18 6a.993.993 0 0 0-.972.804l-1.189 6A.995.995 0 0 0 2.991 14h11.018a1 1 0 0 0 .972-
1.196l-1.19-6a.993.993 0 0 0-.97-.804H4.18zM6 5v1h5V5h1.825c.946 0 1.76.673 1.946 1.608l1.19 6A2 2 0 0 1
14.016 15H2.984a1.992 1.992 0 0 1-1.945-2.392l1.19-6C2.414 5.673 3.229 5 4.174 5H6z"
```

```
                    />
                  </svg>
                </importAction>
              </EuiIcon>
              <div
                className="euiFilePicker__promptText"
              >
                Select or drag your license file
              </div>
            </div>
          </div>
        </div>
      </EuiFilePicker>
    </div>
  </EuiText>
  <EuiSpacer
    size="m"
  >
    <div
      className="euiSpacer euiSpacer--m"
    />
  </EuiSpacer>
  <TelemetryOptIn />
  <EuiSpacer
    size="m"
  >
    <div
      className="euiSpacer euiSpacer--m"
    />
  </EuiSpacer>
  <EuiFlexGroup
    alignItems="stretch"
    component="div"
    direction="row"
    gutterSize="l"
    justifyContent="spaceBetween"
    responsive={true}
    wrap={false}
  >
    <div
      className="euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceBetween
euiFlexGroup--directionRow euiFlexGroup--responsive"
    >
      <EuiFlexItem
        component="div"
        grow={false}
      >
        <div
```

```
    className="euiFlexItem euiFlexItem--flexGrowZero"
  >
    <EuiButtonEmpty
      color="primary"
      href="#/management/elasticsearch/license_management/home"
      iconSide="left"
      type="button"
    >
      <a
        className="euiButtonEmpty euiButtonEmpty--primary"
        href="#/management/elasticsearch/license_management/home"
      >
        <span
          className="euiButtonEmpty__content"
        >
          <span
            className="euiButtonEmpty__text"
          >
            Cancel
          </span>
        </span>
      </a>
    </EuiButtonEmpty>
  </div>
</EuiFlexItem>
<EuiFlexItem
  component="div"
  grow={false}
>
  <div
    className="euiFlexItem euiFlexItem--flexGrowZero"
  >
    <EuiButton
      color="primary"
      data-test-subj="uploadLicenseButton"
      fill={true}
      iconSide="left"
      isLoading={false}
      onClick={[Function]}
      type="button"
    >
      <button
        className="euiButton euiButton--primary euiButton--fill"
        data-test-subj="uploadLicenseButton"
        onClick={[Function]}
        type="button"
      >
        <span
```

```
                    className="euiButton__content"
                  >
                    <span
                      className="euiButton__text"
                    >
                      Upload
                    </span>
                  </span>
                </button>
              </EuiButton>
            </div>
          </EuiFlexItem>
        </div>
      </EuiFlexGroup>
    </div>
  </EuiForm>
</div>
</EuiPageContentBody>
</div>
</EuiPanel>
</EuiPageContent>
</UploadLicense>
</Connect(UploadLicense)>
</Provider>
`;

exports[`UploadLicense should display an error when ES says license is invalid 1`] = `
<Provider
 store={
  Object {
   "dispatch": [Function],
   "getState": [Function],
   "replaceReducer": [Function],
   "subscribe": [Function],
   Symbol(observable): [Function],
  }
 }
>
 <Connect(UploadLicense)>
  <UploadLicense
   addUploadErrorMessage={[Function]}
   applying={false}
   currentLicenseType=""
   errorMessage="The supplied license is not valid for this product."
   isInvalid={false}
   needsAcknowledgement={false}
   uploadLicense={[Function]}
   uploadLicenseStatus={[Function]}
```

```
  >
    <EuiTitle
      className="eui-textCenter"
      size="l"
    >
      <h1
        className="euiTitle euiTitle--large eui-textCenter"
      >
        Upload your license
      </h1>
    </EuiTitle>
    <EuiSpacer
      size="l"
    >
      <div
        className="euiSpacer euiSpacer--l"
      />
    </EuiSpacer>
    <EuiPageContent
      horizontalPosition="center"
      panelPaddingSize="l"
      verticalPosition="center"
    >
      <EuiPanel
        className="euiPageContent euiPageContent--verticalCenter euiPageContent--horizontalCenter"
        grow={true}
        hasShadow={false}
        paddingSize="l"
      >
        <div
          className="euiPanel euiPanel--paddingLarge euiPageContent euiPageContent--verticalCenter
euiPageContent--horizontalCenter"
        >
          <EuiPageContentBody>
            <div
              className="euiPageContentBody"
            >
              <EuiText
                grow={true}
              >
                <div
                  className="euiText"
                >
                  <p>
                    Your license key is a JSON file with a signature attached.
                  </p>
                  <p>
                    Uploading a license will replace your current
```

```
      <strong />
       license.
    </p>
  </div>
</EuiText>
<EuiSpacer
  size="l"
>
  <div
    className="euiSpacer euiSpacer--l"
  />
</EuiSpacer>
<EuiForm
  error={
    Array [
      "The supplied license is not valid for this product.",
    ]
  }
  isInvalid={true}
>
  <div
    className="euiForm"
  >
    <EuiCallOut
      className="euiForm__errors"
      color="danger"
      size="m"
      title="Please address the errors in your form."
    >
      <div
        className="euiCallOut euiCallOut--danger euiForm__errors"
      >
        <div
          className="euiCallOutHeader"
        >
          <span
            className="euiCallOutHeader__title"
          >
            Please address the errors in your form.
          </span>
        </div>
        <EuiText
          grow={true}
          size="s"
        >
          <div
            className="euiText euiText--small"
```

```
      >
        <ul>
          <li
            className="euiForm__error"
            key="The supplied license is not valid for this product."
          >
            The supplied license is not valid for this product.
          </li>
        </ul>
      </div>
    </EuiText>
  </div>
</EuiCallOut>
<EuiText
  grow={true}
>
  <div
    className="euiText"
  >
    <EuiFilePicker
      compressed={false}
      id="licenseFile"
      initialPromptText="Select or drag your license file"
      onChange={[Function]}
    >
      <div
        className="euiFilePicker"
      >
        <div
          className="euiFilePicker__wrap"
        >
          <input
            className="euiFilePicker__input"
            id="licenseFile"
            onChange={[Function]}
            onDragLeave={[Function]}
            onDragOver={[Function]}
            onDrop={[Function]}
            type="file"
          />
          <div
            className="euiFilePicker__prompt"
          >
            <EuiIcon
              aria-hidden="true"
              className="euiFilePicker__icon"
              size="l"
              type="importAction"
```

```
                >
                  <importAction
                    aria-hidden="true"
                    className="euiIcon euiIcon--large euiFilePicker__icon"
                    focusable="false"
                    height="16"
                    style={
                      Object {
                        "fill": undefined,
                      }
                    }
                    viewBox="0 0 16 16"
                    width="16"
                    xmlns="http://www.w3.org/2000/svg"
                  >
                    <svg
                      aria-hidden="true"
                      className="euiIcon euiIcon--large euiFilePicker__icon"
                      focusable="false"
                      height="16"
                      style={
                        Object {
                          "fill": undefined,
                        }
                      }
                      viewBox="0 0 16 16"
                      width="16"
                      xmlns="http://www.w3.org/2000/svg"
                    >
                      <path
                        d="M9 10.114l1.85-1.943a.52.52 0 0 1 .77 0c.214.228.214.6 0 .829l-1.95 2.05a1.552 1.552
0 0 1-2.31 0L5.41 9a.617.617 0 0 1 0-.829.52.52 0 0 1 .77 0L8 10.082V1.556C8 1.249 8.224 1 8.5
1s.5.249.5.556v8.558zM4.18 6a.993.993 0 0 0-.972.804l-1.189 6A.995.995 0 0 0 2.991 14h11.018a1 1 0 0 0 .972-
1.196l-1.19-6a.993.993 0 0 0-.97-.804H4.18zM6 5v1h5V5h1.825c.946 0 1.76.673 1.946 1.608l1.19 6A2 2 0 0 1
14.016 15H2.984a1.992 1.992 0 0 1-1.945-2.392l1.19-6C2.414 5.673 3.229 5 4.174 5H6z"
                      />
                    </svg>
                  </importAction>
                </EuiIcon>
                <div
                  className="euiFilePicker__promptText"
                >
                  Select or drag your license file
                </div>
              </div>
            </div>
          </div>
        </EuiFilePicker>
```

```
          </div>
        </EuiText>
        <EuiSpacer
          size="m"
        >
          <div
            className="euiSpacer euiSpacer--m"
          />
        </EuiSpacer>
        <TelemetryOptIn />
        <EuiSpacer
          size="m"
        >
          <div
            className="euiSpacer euiSpacer--m"
          />
        </EuiSpacer>
        <EuiFlexGroup
          alignItems="stretch"
          component="div"
          direction="row"
          gutterSize="l"
          justifyContent="spaceBetween"
          responsive={true}
          wrap={false}
        >
          <div
            className="euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceBetween
euiFlexGroup--directionRow euiFlexGroup--responsive"
          >
            <EuiFlexItem
              component="div"
              grow={false}
            >
              <div
                className="euiFlexItem euiFlexItem--flexGrowZero"
              >
                <EuiButtonEmpty
                  color="primary"
                  href="#/management/elasticsearch/license_management/home"
                  iconSide="left"
                  type="button"
                >
                  <a
                    className="euiButtonEmpty euiButtonEmpty--primary"
                    href="#/management/elasticsearch/license_management/home"
                  >
                    <span
```

```
            className="euiButtonEmpty__content"
          >
            <span
              className="euiButtonEmpty__text"
            >
              Cancel
            </span>
          </span>
        </a>
      </EuiButtonEmpty>
    </div>
  </EuiFlexItem>
  <EuiFlexItem
    component="div"
    grow={false}
  >
    <div
      className="euiFlexItem euiFlexItem--flexGrowZero"
    >
      <EuiButton
        color="primary"
        data-test-subj="uploadLicenseButton"
        fill={true}
        iconSide="left"
        isLoading={false}
        onClick={[Function]}
        type="button"
      >
        <button
          className="euiButton euiButton--primary euiButton--fill"
          data-test-subj="uploadLicenseButton"
          onClick={[Function]}
          type="button"
        >
          <span
            className="euiButton__content"
          >
            <span
              className="euiButton__text"
            >
              Upload
            </span>
          </span>
        </button>
      </EuiButton>
    </div>
  </EuiFlexItem>
</div>
```

```
                </EuiFlexGroup>
              </div>
            </EuiForm>
          </div>
        </EuiPageContentBody>
      </div>
    </EuiPanel>
  </EuiPageContent>
 </UploadLicense>
</Connect(UploadLicense)>
</Provider>
`;

exports[`UploadLicense should display an error when submitting invalid JSON 1`] = `
<Provider
 store={
  Object {
   "dispatch": [Function],
   "getState": [Function],
   "replaceReducer": [Function],
   "subscribe": [Function],
   Symbol(observable): [Function],
  }
 }
>
 <Connect(UploadLicense)>
  <UploadLicense
   addUploadErrorMessage={[Function]}
   applying={false}
   currentLicenseType=""
   errorMessage="Error encountered uploading license: Check your license file."
   isInvalid={false}
   needsAcknowledgement={false}
   uploadLicense={[Function]}
   uploadLicenseStatus={[Function]}
  >
   <EuiTitle
    className="eui-textCenter"
    size="l"
   >
    <h1
     className="euiTitle euiTitle--large eui-textCenter"
    >
     Upload your license
    </h1>
   </EuiTitle>
   <EuiSpacer
    size="l"
```

```
    >
      <div
        className="euiSpacer euiSpacer--l"
      />
    </EuiSpacer>
    <EuiPageContent
      horizontalPosition="center"
      panelPaddingSize="l"
      verticalPosition="center"
    >
      <EuiPanel
        className="euiPageContent euiPageContent--verticalCenter euiPageContent--horizontalCenter"
        grow={true}
        hasShadow={false}
        paddingSize="l"
      >
        <div
          className="euiPanel euiPanel--paddingLarge euiPageContent euiPageContent--verticalCenter
euiPageContent--horizontalCenter"
        >
          <EuiPageContentBody>
            <div
              className="euiPageContentBody"
            >
              <EuiText
                grow={true}
              >
                <div
                  className="euiText"
                >
                  <p>
                    Your license key is a JSON file with a signature attached.
                  </p>
                  <p>
                    Uploading a license will replace your current

                    <strong />
                     license.
                  </p>
                </div>
              </EuiText>
              <EuiSpacer
                size="l"
              >
                <div
                  className="euiSpacer euiSpacer--l"
                />
              </EuiSpacer>
```

```
<EuiForm
  error={
    Array [
      "Error encountered uploading license: Check your license file.",
    ]
  }
  isInvalid={true}
>
  <div
    className="euiForm"
  >
    <EuiCallOut
      className="euiForm__errors"
      color="danger"
      size="m"
      title="Please address the errors in your form."
    >
      <div
        className="euiCallOut euiCallOut--danger euiForm__errors"
      >
        <div
          className="euiCallOutHeader"
        >
          <span
            className="euiCallOutHeader__title"
          >
            Please address the errors in your form.
          </span>
        </div>
        <EuiText
          grow={true}
          size="s"
        >
          <div
            className="euiText euiText--small"
          >
            <ul>
              <li
                className="euiForm__error"
                key="Error encountered uploading license: Check your license file."
              >
                Error encountered uploading license: Check your license file.
              </li>
            </ul>
          </div>
        </EuiText>
      </div>
    </EuiCallOut>
```

```
<EuiText
  grow={true}
>
  <div
    className="euiText"
  >
    <EuiFilePicker
      compressed={false}
      id="licenseFile"
      initialPromptText="Select or drag your license file"
      onChange={[Function]}
    >
      <div
        className="euiFilePicker"
      >
        <div
          className="euiFilePicker__wrap"
        >
          <input
            className="euiFilePicker__input"
            id="licenseFile"
            onChange={[Function]}
            onDragLeave={[Function]}
            onDragOver={[Function]}
            onDrop={[Function]}
            type="file"
          />
          <div
            className="euiFilePicker__prompt"
          >
            <EuiIcon
              aria-hidden="true"
              className="euiFilePicker__icon"
              size="l"
              type="importAction"
            >
              <importAction
                aria-hidden="true"
                className="euiIcon euiIcon--large euiFilePicker__icon"
                focusable="false"
                height="16"
                style={
                  Object {
                    "fill": undefined,
                  }
                }
                viewBox="0 0 16 16"
                width="16"
```

```
                xmlns="http://www.w3.org/2000/svg"
              >
                <svg
                  aria-hidden="true"
                  className="euiIcon euiIcon--large euiFilePicker__icon"
                  focusable="false"
                  height="16"
                  style={
                    Object {
                      "fill": undefined,
                    }
                  }
                  viewBox="0 0 16 16"
                  width="16"
                  xmlns="http://www.w3.org/2000/svg"
                >
                  <path
                    d="M9 10.114l1.85-1.943a.52.52 0 0 1 .77 0c.214.228.214.6 0 .829l-1.95 2.05a1.552 1.552
0 0 1-2.31 0L5.41 9a.617.617 0 0 1 0-.829.52.52 0 0 1 .77 0L8 10.082V1.556C8 1.249 8.224 1 8.5
1s.5.249.5.556v8.558zM4.18 6a.993.993 0 0 0-.972.804l-1.189 6A.995.995 0 0 0 2.991 14h11.018a1 1 0 0 0 .972-
1.196l-1.19-6a.993.993 0 0 0-.97-.804H4.18zM6 5v1h5V5h1.825c.946 0 1.76.673 1.946 1.608l1.19 6A2 2 0 0 1
14.016 15H2.984a1.992 1.992 0 0 1-1.945-2.392l1.19-6C2.414 5.673 3.229 5 4.174 5H6z"
                  />
                </svg>
              </importAction>
            </EuiIcon>
            <div
              className="euiFilePicker__promptText"
            >
              Select or drag your license file
            </div>
          </div>
        </div>
      </div>
    </EuiFilePicker>
  </div>
</EuiText>
<EuiSpacer
  size="m"
>
  <div
    className="euiSpacer euiSpacer--m"
  />
</EuiSpacer>
<TelemetryOptIn />
<EuiSpacer
  size="m"
>
```

```
            <div
              className="euiSpacer euiSpacer--m"
            />
          </EuiSpacer>
          <EuiFlexGroup
            alignItems="stretch"
            component="div"
            direction="row"
            gutterSize="l"
            justifyContent="spaceBetween"
            responsive={true}
            wrap={false}
          >
            <div
              className="euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceBetween
euiFlexGroup--directionRow euiFlexGroup--responsive"
            >
              <EuiFlexItem
                component="div"
                grow={false}
              >
                <div
                  className="euiFlexItem euiFlexItem--flexGrowZero"
                >
                  <EuiButtonEmpty
                    color="primary"
                    href="#/management/elasticsearch/license_management/home"
                    iconSide="left"
                    type="button"
                  >
                    <a
                      className="euiButtonEmpty euiButtonEmpty--primary"
                      href="#/management/elasticsearch/license_management/home"
                    >
                      <span
                        className="euiButtonEmpty__content"
                      >
                        <span
                          className="euiButtonEmpty__text"
                        >
                          Cancel
                        </span>
                      </span>
                    </a>
                  </EuiButtonEmpty>
                </div>
              </EuiFlexItem>
              <EuiFlexItem
```

```
                    component="div"
                    grow={false}
                  >
                    <div
                      className="euiFlexItem euiFlexItem--flexGrowZero"
                    >
                      <EuiButton
                        color="primary"
                        data-test-subj="uploadLicenseButton"
                        fill={true}
                        iconSide="left"
                        isLoading={false}
                        onClick={[Function]}
                        type="button"
                      >
                        <button
                          className="euiButton euiButton--primary euiButton--fill"
                          data-test-subj="uploadLicenseButton"
                          onClick={[Function]}
                          type="button"
                        >
                          <span
                            className="euiButton__content"
                          >
                            <span
                              className="euiButton__text"
                            >
                              Upload
                            </span>
                          </span>
                        </button>
                      </EuiButton>
                    </div>
                  </EuiFlexItem>
                </div>
              </EuiFlexGroup>
            </div>
          </EuiForm>
        </div>
      </EuiPageContentBody>
    </div>
  </EuiPanel>
</EuiPageContent>
</UploadLicense>
</Connect(UploadLicense)>
</Provider>
`;
```

```
exports[`UploadLicense should display error when ES returns error 1`] = `
<Provider
 store={
   Object {
     "dispatch": [Function],
     "getState": [Function],
     "replaceReducer": [Function],
     "subscribe": [Function],
     Symbol(observable): [Function],
   }
 }
>
 <Connect(UploadLicense)>
   <UploadLicense
     addUploadErrorMessage={[Function]}
     applying={false}
     currentLicenseType=""
     errorMessage="Error encountered uploading license: Can not upgrade to a production license unless TLS is
configured or security is disabled"
     isInvalid={false}
     needsAcknowledgement={false}
     uploadLicense={[Function]}
     uploadLicenseStatus={[Function]}
   >
     <EuiTitle
       className="eui-textCenter"
       size="l"
     >
       <h1
         className="euiTitle euiTitle--large eui-textCenter"
       >
         Upload your license
       </h1>
     </EuiTitle>
     <EuiSpacer
       size="l"
     >
       <div
         className="euiSpacer euiSpacer--l"
       />
     </EuiSpacer>
     <EuiPageContent
       horizontalPosition="center"
       panelPaddingSize="l"
       verticalPosition="center"
     >
       <EuiPanel
         className="euiPageContent euiPageContent--verticalCenter euiPageContent--horizontalCenter"
```

```
      grow={true}
      hasShadow={false}
      paddingSize="l"
    >
      <div
        className="euiPanel euiPanel--paddingLarge euiPageContent euiPageContent--verticalCenter
euiPageContent--horizontalCenter"
      >
        <EuiPageContentBody>
          <div
            className="euiPageContentBody"
          >
            <EuiText
              grow={true}
            >
              <div
                className="euiText"
              >
                <p>
                  Your license key is a JSON file with a signature attached.
                </p>
                <p>
                  Uploading a license will replace your current

                  <strong />
                   license.
                </p>
              </div>
            </EuiText>
            <EuiSpacer
              size="l"
            >
              <div
                className="euiSpacer euiSpacer--l"
              />
            </EuiSpacer>
            <EuiForm
              error={
                Array [
                  "Error encountered uploading license: Can not upgrade to a production license unless TLS is
configured or security is disabled",
                ]
              }
              isInvalid={true}
            >
              <div
                className="euiForm"
              >
```

```
<EuiCallOut
  className="euiForm__errors"
  color="danger"
  size="m"
  title="Please address the errors in your form."
>
  <div
    className="euiCallOut euiCallOut--danger euiForm__errors"
  >
    <div
      className="euiCallOutHeader"
    >
      <span
        className="euiCallOutHeader__title"
      >
        Please address the errors in your form.
      </span>
    </div>
    <EuiText
      grow={true}
      size="s"
    >
      <div
        className="euiText euiText--small"
      >
        <ul>
          <li
            className="euiForm__error"
            key="Error encountered uploading license: Can not upgrade to a production license unless TLS
is configured or security is disabled"
          >
            Error encountered uploading license: Can not upgrade to a production license unless TLS is
configured or security is disabled
          </li>
        </ul>
      </div>
    </EuiText>
  </div>
</EuiCallOut>
<EuiText
  grow={true}
>
  <div
    className="euiText"
  >
    <EuiFilePicker
      compressed={false}
      id="licenseFile"
```

```
  initialPromptText="Select or drag your license file"
  onChange={[Function]}
>
  <div
    className="euiFilePicker"
  >
    <div
      className="euiFilePicker__wrap"
    >
      <input
        className="euiFilePicker__input"
        id="licenseFile"
        onChange={[Function]}
        onDragLeave={[Function]}
        onDragOver={[Function]}
        onDrop={[Function]}
        type="file"
      />
      <div
        className="euiFilePicker__prompt"
      >
        <EuiIcon
          aria-hidden="true"
          className="euiFilePicker__icon"
          size="l"
          type="importAction"
        >
          <importAction
            aria-hidden="true"
            className="euiIcon euiIcon--large euiFilePicker__icon"
            focusable="false"
            height="16"
            style={
              Object {
                "fill": undefined,
              }
            }
            viewBox="0 0 16 16"
            width="16"
            xmlns="http://www.w3.org/2000/svg"
          >
            <svg
              aria-hidden="true"
              className="euiIcon euiIcon--large euiFilePicker__icon"
              focusable="false"
              height="16"
              style={
                Object {
```

```
                "fill": undefined,
              }
            }
            viewBox="0 0 16 16"
            width="16"
            xmlns="http://www.w3.org/2000/svg"
          >
            <path
              d="M9 10.114l1.85-1.943a.52.52 0 0 1 .77 0c.214.228.214.6 0 .829l-1.95 2.05a1.552 1.552
0 0 1-2.31 0L5.41 9a.617.617 0 0 1 0-.829.52.52 0 0 1 .77 0L8 10.082V1.556C8 1.249 8.224 1 8.5
1s.5.249.5.556v8.558zM4.18 6a.993.993 0 0 0-.972.804l-1.189 6A.995.995 0 0 0 2.991 14h11.018a1 1 0 0 0 .972-
1.196l-1.19-6a.993.993 0 0 0-.97-.804H4.18zM6 5v1h5V5h1.825c.946 0 1.76.673 1.946 1.608l1.19 6A2 2 0 0 1
14.016 15H2.984a1.992 1.992 0 0 1-1.945-2.392l1.19-6C2.414 5.673 3.229 5 4.174 5H6z"
            />
          </svg>
        </importAction>
      </EuiIcon>
      <div
        className="euiFilePicker__promptText"
      >
        Select or drag your license file
      </div>
    </div>
  </div>
</div>
</EuiFilePicker>
</div>
</EuiText>
<EuiSpacer
  size="m"
>
  <div
    className="euiSpacer euiSpacer--m"
  />
</EuiSpacer>
<TelemetryOptIn />
<EuiSpacer
  size="m"
>
  <div
    className="euiSpacer euiSpacer--m"
  />
</EuiSpacer>
<EuiFlexGroup
  alignItems="stretch"
  component="div"
  direction="row"
  gutterSize="l"
```

```
                justifyContent="spaceBetween"
                responsive={true}
                wrap={false}
               >
                <div
                  className="euiFlexGroup euiFlexGroup--gutterLarge euiFlexGroup--justifyContentSpaceBetween
euiFlexGroup--directionRow euiFlexGroup--responsive"
                >
                  <EuiFlexItem
                    component="div"
                    grow={false}
                  >
                    <div
                      className="euiFlexItem euiFlexItem--flexGrowZero"
                    >
                      <EuiButtonEmpty
                        color="primary"
                        href="#/management/elasticsearch/license_management/home"
                        iconSide="left"
                        type="button"
                      >
                        <a
                          className="euiButtonEmpty euiButtonEmpty--primary"
                          href="#/management/elasticsearch/license_management/home"
                        >
                          <span
                            className="euiButtonEmpty__content"
                          >
                            <span
                              className="euiButtonEmpty__text"
                            >
                              Cancel
                            </span>
                          </span>
                        </a>
                      </EuiButtonEmpty>
                    </div>
                  </EuiFlexItem>
                  <EuiFlexItem
                    component="div"
                    grow={false}
                  >
                    <div
                      className="euiFlexItem euiFlexItem--flexGrowZero"
                    >
                      <EuiButton
                        color="primary"
                        data-test-subj="uploadLicenseButton"
```

```
                    fill={true}
                    iconSide="left"
                    isLoading={false}
                    onClick={[Function]}
                    type="button"
                  >
                    <button
                      className="euiButton euiButton--primary euiButton--fill"
                      data-test-subj="uploadLicenseButton"
                      onClick={[Function]}
                      type="button"
                    >
                      <span
                        className="euiButton__content"
                      >
                        <span
                          className="euiButton__text"
                        >
                          Upload
                        </span>
                      </span>
                    </button>
                  </EuiButton>
                </div>
              </EuiFlexItem>
            </div>
          </EuiFlexGroup>
        </div>
      </EuiForm>
    </div>
  </EuiPageContentBody>
</div>
</EuiPanel>
</EuiPageContent>
</UploadLicense>
</Connect(UploadLicense)>
</Provider>
`;
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`RevertToBasic component should display when license is about to expire 1`] = `"<div
class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\"><span
class=\\"euiCard__content\\"><span class=\\"euiTitle euiTitle--medium euiCard__title\\">Revert to Basic
license</span><div class=\\"euiText euiText--small euiCard__description\\"><p><span>Youll revert to our free
features and lose access to security, machine learning and other <a class=\\"euiLink euiLink--primary\\"
href=\\"https://www.elastic.co/subscriptions/xpack\\" target=\\"_blank\\" rel=\\"noopener noreferrer\\">Platinum
features</a>.</span></p></div></span><span class=\\"euiCard__footer\\"><button class=\\"euiButton euiButton--
primary\\" type=\\"button\\" data-test-subj=\\"revertToBasicButton\\"><span class=\\"euiButton__content\\"><span
```

```
class=\\"euiButton__text\\">Revert to Basic</span></span></button></span></div></div>"`;

exports[`RevertToBasic component should display when license is expired 1`] = `"<div
class=\\"euiFlexItem\\"><div class=\\"euiCard euiCard--centerAligned\\"><span
class=\\"euiCard__content\\"><span class=\\"euiTitle euiTitle--medium euiCard__title\\">Revert to Basic
license</span><div class=\\"euiText euiText--small euiCard__description\\"><p><span>Youll revert to our free
features and lose access to security, machine learning and other <a class=\\"euiLink euiLink--primary\\"
href=\\"https://www.elastic.co/subscriptions/xpack\\" target=\\"_blank\\" rel=\\"noopener noreferrer\\">Platinum
features</a>.</span></p></div></span><span class=\\"euiCard__footer\\"><button class=\\"euiButton euiButton--
primary\\" type=\\"button\\" data-test-subj=\\"revertToBasicButton\\"><span class=\\"euiButton__content\\"><span
class=\\"euiButton__text\\">Revert to Basic</span></span></button></span></div></div>"`;

exports[`RevertToBasic component should display when trial is active 1`] = `"<div class=\\"euiFlexItem\\"><div
class=\\"euiCard euiCard--centerAligned\\"><span class=\\"euiCard__content\\"><span class=\\"euiTitle euiTitle--
medium euiCard__title\\">Revert to Basic license</span><div class=\\"euiText euiText--small
euiCard__description\\"><p><span>Youll revert to our free features and lose access to security, machine learning
and other <a class=\\"euiLink euiLink--primary\\" href=\\"https://www.elastic.co/subscriptions/xpack\\"
target=\\"_blank\\" rel=\\"noopener noreferrer\\">Platinum features</a>.</span></p></div></span><span
class=\\"euiCard__footer\\"><button class=\\"euiButton euiButton--primary\\" type=\\"button\\" data-test-
subj=\\"revertToBasicButton\\"><span class=\\"euiButton__content\\"><span class=\\"euiButton__text\\">Revert to
Basic</span></span></button></span></div></div>"`;
// Jest Snapshot v1, https://goo.gl/fbAQLP

exports[`TelemetryOptIn should display when telemetry not opted in 1`] = `
<TelemetryOptIn>
 <EuiText
  grow={true}
 >
  <div
   className="euiText"
  >
   <p>
    Gold and platinum customers: help support give you better service.
   </p>
  </div>
 </EuiText>
 <EuiSpacer
  size="s"
 >
  <div
   className="euiSpacer euiSpacer--s"
  />
 </EuiSpacer>
 <EuiCheckbox
  checked={false}
  compressed={false}
  disabled={false}
  id="isOptingInToTelemetry"
```

```
  indeterminate={false}
  label={
    <span>
      Send basic feature usage statistics to Elastic periodically.
      <EuiPopover
        anchorPosition="downCenter"
        button={
          <EuiLink
            color="primary"
            onClick={[Function]}
            type="button"
          >
            Read more
          </EuiLink>
        }
        className="eui-AlignBaseline"
        closePopover={[Function]}
        hasArrow={true}
        id="readMorePopover"
        isOpen={false}
        ownFocus={true}
        panelPaddingSize="m"
      >
        <EuiText
          className="licManagement__narrowText"
          grow={true}
        >
          <p>
            This feature periodically sends basic feature usage statistics. This information will not be shared outside of
Elastic. See an
            <EuiLink
              color="primary"
              onClick={[Function]}
              type="button"
            >
              example
            </EuiLink>

            or read our

            <EuiLink
              color="primary"
              href="https://www.elastic.co/legal/telemetry-privacy-statement"
              target="_blank"
              type="button"
            >
              telemetry privacy statement
            </EuiLink>
```

```
          . You can disable this feature any time.
        </p>
      </EuiText>
    </EuiPopover>
  </span>
}
onChange={[Function]}
>
  <div
    className="euiCheckbox"
  >
    <input
      checked={false}
      className="euiCheckbox__input"
      disabled={false}
      id="isOptingInToTelemetry"
      onChange={[Function]}
      type="checkbox"
    />
    <div
      className="euiCheckbox__square"
    />
    <label
      className="euiCheckbox__label"
      htmlFor="isOptingInToTelemetry"
    >
      <span>
        Send basic feature usage statistics to Elastic periodically.
        <EuiPopover
          anchorPosition="downCenter"
          button={
            <EuiLink
              color="primary"
              onClick={[Function]}
              type="button"
            >
              Read more
            </EuiLink>
          }
          className="eui-AlignBaseline"
          closePopover={[Function]}
          hasArrow={true}
          id="readMorePopover"
          isOpen={false}
          ownFocus={true}
          panelPaddingSize="m"
        >
          <EuiOutsideClickDetector
```

```
          onOutsideClick={[Function]}
        >
          <div
            className="euiPopover euiPopover--anchorDownCenter eui-AlignBaseline"
            id="readMorePopover"
            onClick={[Function]}
            onKeyDown={[Function]}
          >
            <div
              className="euiPopover__anchor"
            >
              <EuiLink
                color="primary"
                onClick={[Function]}
                type="button"
              >
                <button
                  className="euiLink euiLink--primary"
                  onClick={[Function]}
                  type="button"
                >
                  Read more
                </button>
              </EuiLink>
            </div>
          </div>
        </EuiOutsideClickDetector>
      </EuiPopover>
    </span>
  </label>
 </div>
</EuiCheckbox>
</TelemetryOptIn>
`;

exports[`TelemetryOptIn should not display when telemetry is opted in 1`] = `<TelemetryOptIn />`;
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
import React from 'react';
import { TelemetryOptIn } from '../public/components/telemetry_opt_in';
import { mount } from 'enzyme';
describe('TelemetryOptIn', () => {
 test('should display when telemetry not opted in', () => {
   const telemetry = require('../public/lib/telemetry');
   telemetry.showTelemetryOptIn = () => { return true; };
```

```
  const rendered = mount(<TelemetryOptIn />);
  expect(rendered).toMatchSnapshot();
 });
 test('should not display when telemetry is opted in', () => {
  const telemetry = require('../public/lib/telemetry');
  telemetry.showTelemetryOptIn = () => { return false; };
  const rendered = mount(<TelemetryOptIn />);
  expect(rendered).toMatchSnapshot();
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { StartTrial } from '../public/sections/license_dashboard/start_trial';
import { createMockLicense, getComponent } from './util';

describe('StartTrial component when trial is allowed', () => {
 test('display for basic license', () => {
  const rendered = getComponent(
   {
    license: createMockLicense('basic'),
    trialStatus: { canStartTrial: true }
   },
   StartTrial
  );
  expect(rendered.html()).toMatchSnapshot();
 });
 test('should display for gold license', () => {
  const rendered = getComponent(
   {
    license: createMockLicense('gold'),
    trialStatus: { canStartTrial: true }
   },
   StartTrial
  );
  expect(rendered.html()).toMatchSnapshot();
 });

 test('should not display for trial license', () => {
  const rendered = getComponent(
   {
    license: createMockLicense('trial'),
    trialStatus: { canStartTrial: true }
   },
   StartTrial
```

```
  );
  expect(rendered.html()).toBeNull();
});
test('should not display for active platinum license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('platinum'),
      trialStatus: { canStartTrial: true }
    },
    StartTrial
  );
  expect(rendered.html()).toBeNull();
});
test('should display for expired platinum license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('platinum', 0),
      trialStatus: { canStartTrial: true }
    },
    StartTrial
  );
  expect(rendered.html()).toMatchSnapshot();
});
});

describe('StartTrial component when trial is not available', () => {
test('should not display for basic license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('basic'),
      trialStatus: { canStartTrial: false }
    },
    StartTrial
  );
  expect(rendered.html()).toBeNull();
});
test('should not display for gold license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('gold'),
      trialStatus: { canStartTrial: false }
    },
    StartTrial
  );
  expect(rendered.html()).toBeNull();
});
test('should not display for platinum license', () => {
  const rendered = getComponent(
```

```
    {
      license: createMockLicense('platinum'),
      trialStatus: { canStartTrial: false }
    },
    StartTrial
  );
  expect(rendered.html()).toBeNull();
});

test('should not display for trial license', () => {
  const rendered = getComponent(
    {
      license: createMockLicense('gold'),
      trialStatus: { canStartTrial: false }
    },
    StartTrial
  );
  expect(rendered.html()).toBeNull();
});
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { Provider } from 'react-redux';
import { licenseManagementStore } from '../../public/store/store';
import React from 'react';
import { mount } from 'enzyme';

const highExpirationMillis = new Date('October 13, 2099 00:00:00Z').getTime();

export const createMockLicense = (
 type,
 expiryDateInMillis = highExpirationMillis
) => {
 return {
   type,
   expiryDateInMillis,
   isActive: new Date().getTime() < expiryDateInMillis
 };
};
export const getComponent = (initialState, Component) => {
 const store = licenseManagementStore(initialState);
 return mount(
   <Provider store={store}>
     <Component />
```

```
  </Provider>
 );
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export * from './util';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export const UPLOAD_LICENSE_EXPIRED = [
 200,
 { 'Content-Type': 'application/json' },
 '{"acknowledged": "true", "license_status": "expired"}'
];

/* eslint-disable max-len */
export const UPLOAD_LICENSE_REQUIRES_ACK = [
 200,
 { 'Content-Type': 'application/json' },
 `{
   "acknowledged":false,
   "license_status":"valid",
   "acknowledge":
     {
       "message": "This license update requires acknowledgement. To acknowledge the license, please read the
following messages and update the license again, this time with the \\"acknowledge=true\\" parameter:",
       "watcher":["Watcher will be disabled"]
     }
   }`
];
/* eslint-enable max-len */

export const UPLOAD_LICENSE_SUCCESS = [
 200,
 { 'Content-Type': 'application/json' },
 '{"acknowledged": "true", "license_status": "valid"}'
];

export const UPLOAD_LICENSE_INVALID = [
 200,
 { 'Content-Type': 'application/json' },
```

```
 '{"acknowledged": "true", "license_status": "invalid"}'
];

export const UPLOAD_LICENSE_TLS_NOT_ENABLED = [
 200,
 { 'Content-Type': 'application/json' },
 `{
    "error":
    {
      "root_cause":
        [{
          "type":"illegal_state_exception",
          "reason":"Can not upgrade to a production license unless TLS is configured or security is disabled"
        }],"type":"illegal_state_exception",
        "reason":"Can not upgrade to a production license unless TLS is configured or security is disabled"},
        "status":500}
  `
];
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export * from './upload_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { AddLicense } from '../public/sections/license_dashboard/add_license';
import { createMockLicense, getComponent } from './util';

describe('AddLicense component when license is active', () => {
 test('should display correct verbiage', () => {
   const rendered = getComponent(
     {
       license: createMockLicense('basic')
     },
     AddLicense
   );
   expect(rendered.html()).toMatchSnapshot();
 });
});

describe('AddLicense component when license is expired', () => {
 test('should display with correct verbiage', () => {
```

```
  const rendered = getComponent(
    {
      license: createMockLicense('platinum', 0)
    },
    AddLicense
  );
  expect(rendered.html()).toMatchSnapshot();
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export const PLUGIN = {
 ID: 'license_management'
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { PLUGIN } from './plugin';
export { BASE_PATH } from './base_path';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export const BASE_PATH = '/management/elasticsearch/license_management/';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';

export default (props) => <div>{props.children}</div>;
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
import { resolve } from 'path';
import { PLUGIN } from './common/constants';
import { registerLicenseRoute, registerStartTrialRoutes, registerStartBasicRoute } from
"./server/routes/api/license/";

export function licenseManagement(kibana)  {
 return new kibana.Plugin({
   id: PLUGIN.ID,
   publicDir: resolve(__dirname, 'public'),
   require: ['kibana', 'elasticsearch'],
   uiExports: {
     styleSheetPaths: `${__dirname}/public/index.scss`,
     managementSections: [
       'plugins/license_management',
     ]
   },
   init: (server) => {
     registerLicenseRoute(server);
     registerStartTrialRoutes(server);
     registerStartBasicRoute(server);
   }
 });
}
<kbn-management-app section="elasticsearch/license_management">
 <div id="licenseReactRoot"></div>
</kbn-management-app>
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';
import { render, unmountComponentAtNode } from 'react-dom';
import { Provider } from 'react-redux';
import { HashRouter } from 'react-router-dom';
import { setTelemetryOptInService, setTelemetryEnabled, setHttpClient, TelemetryOptInProvider } from
'./lib/telemetry';


import App from './app';
import { BASE_PATH } from "../common/constants/base_path";

import routes from 'ui/routes';
import { XPackInfoProvider as xpackInfoProvider } from 'plugins/xpack_main/services/xpack_info';

import template from './main.html';
import { licenseManagementStore } from './store';
```

```
const renderReact = (elem, store) => {
 render(
   <Provider store={store}>
     <HashRouter>
       <App />
     </HashRouter>
   </Provider>,
   elem
 );
};

/*
 This method handles the cleanup needed when route is scope is destroyed.  It also prevents Angular
 from destroying scope when route changes and both old route and new route are this same route.
*/
const manageAngularLifecycle = ($scope, $route, elem) => {
 const lastRoute = $route.current;
 const deregister = $scope.$on('$locationChangeSuccess', () => {
   const currentRoute = $route.current;
   // if templates are the same we are on the same route
   if (lastRoute.$$route.template === currentRoute.$$route.template) {
     // this prevents angular from destroying scope
     $route.current = lastRoute;
   }
 });
 $scope.$on('$destroy', () => {
   deregister && deregister();
   // manually unmount component when scope is destroyed
   elem && unmountComponentAtNode(elem);
 });
};
const initializeTelemetry = ($injector) => {
 const telemetryEnabled = $injector.get('telemetryEnabled');
 const Private = $injector.get('Private');
 const telemetryOptInProvider = Private(TelemetryOptInProvider);
 setTelemetryOptInService(telemetryOptInProvider);
 setTelemetryEnabled(telemetryEnabled);
 setHttpClient($injector.get('$http'));
};
routes
 .when(`${BASE_PATH}:view?`, {
   template: template,
   controllerAs: 'licenseManagement',
   controller: class LicenseManagementController {

     constructor($injector, $window, $rootScope, $scope, $route, kbnUrl) {
       initializeTelemetry($injector);
```

```
      let autoLogout = null;
      /* if security is disabled, there will be no autoLogout service,
       so just substitute noop function in that case */
      try {
        autoLogout = $injector.get('autoLogout');
      } catch (e) {
        autoLogout = () => {};
      }

      $scope.$$postDigest(() => {
        const elem = document.getElementById('licenseReactRoot');
        const xPackInfo = xpackInfoProvider($window, $injector, $injector.get('Private'));
        const initialState = { license: xPackInfo.get('license') };
        const kbnUrlWrapper = {
          change(url) {
            kbnUrl.change(url);
            $rootScope.$digest();
          }
        };
        const services = { autoLogout, xPackInfo, kbnUrl: kbnUrlWrapper };
        const store = licenseManagementStore(initialState, services);
        renderReact(elem, store);
        manageAngularLifecycle($scope, $route, elem);
      });
    }
  }
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React, { Fragment } from 'react';
import {
 EuiLink,
 EuiCheckbox,
 EuiSpacer,
 EuiText,
 EuiPopover
} from '@elastic/eui';
import { showTelemetryOptIn, getTelemetryFetcher, PRIVACY_STATEMENT_URL, OptInExampleFlyout } from
'../../lib/telemetry';

export class TelemetryOptIn extends React.Component {
 constructor() {
   super();
   this.state = {
```

```
    showMoreTelemetryInfo: false,
    isOptingInToTelemetry: false,
    showExample: false
  };
}
isOptingInToTelemetry = () => {
  return this.state.isOptingInToTelemetry;
}
closeReadMorePopover = () => {
  this.setState({ showMoreTelemetryInfo: false });
}
onClickReadMore = () => {
  const { showMoreTelemetryInfo } = this.state;
  this.setState({ showMoreTelemetryInfo: !showMoreTelemetryInfo });
}
onClickExample = () => {
  this.setState({ showExample: true });
  this.closeReadMorePopover();
}
onChangeOptIn = event => {
  const isOptingInToTelemetry = event.target.checked;
  this.setState({ isOptingInToTelemetry });
}
render() {
  const { showMoreTelemetryInfo, isOptingInToTelemetry, showExample } = this.state;
  const { isStartTrial } = this.props;

  let example = null;
  if (showExample) {
    example = (
      <OptInExampleFlyout
        onClose={() => this.setState({ showExample: false })}
        fetchTelemetry={getTelemetryFetcher}
      />
    );
  }

  let toCurrentCustomers;
  if (!isStartTrial) {
    toCurrentCustomers = (
      <Fragment>
        <EuiText>
          <p>Gold and platinum customers: help support give you better service.</p>
        </EuiText>
        <EuiSpacer  size="s"/>
      </Fragment>
    );
  }
```

```
const readMoreButton = (
  <EuiLink onClick={this.onClickReadMore}>
  Read more
  </EuiLink>
);

const popover = (
  <EuiPopover
    ownFocus
    id="readMorePopover"
    button={readMoreButton}
    isOpen={showMoreTelemetryInfo}
    closePopover={this.closeReadMorePopover}
    className="eui-AlignBaseline"
  >
    <EuiText className="licManagement__narrowText" >
      <p>
    This feature periodically sends basic feature usage statistics.
    This information will not be shared outside of Elastic.
    See an <EuiLink onClick={this.onClickExample}>example</EuiLink>
        {' '}
      or read our
        {' '}
        <EuiLink
          href={PRIVACY_STATEMENT_URL}
          target="_blank"
        >
      telemetry privacy statement
        </EuiLink>.
    You can disable this feature any time.
      </p>
    </EuiText>
  </EuiPopover>
);

return showTelemetryOptIn() ? (
  <Fragment>
    {example}
    {toCurrentCustomers}
    <EuiCheckbox
      label={<span>Send basic feature usage statistics to Elastic periodically. {popover}</span>}
      id="isOptingInToTelemetry"
      checked={isOptingInToTelemetry}
      onChange={this.onChangeOptIn}
    />
  </Fragment>
) : null;
```

```
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { TelemetryOptIn } from './telemetry_opt_in';
// EUI globals
@import 'ui/public/styles/styling_constants';

// License amnagement plugin styles

// Prefix all styles with "lic" to avoid conflicts.
// Examples
//   licChart
//   licChart__legend
//   licChart__legend--small
//   licChart__legend-isLoading

@import 'license_management';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';
import { BASE_PATH } from '../../../../common/constants';

import { EuiCard, EuiButton } from '@elastic/eui';

export const AddLicense = ({ uploadPath = `#${BASE_PATH}upload_license` }) => {
 return (
   <EuiCard
     title="Update your license"
     description="If you already have a new license, upload it now."
     footer={
       <EuiButton
         data-test-subj="updateLicenseButton"
         href={uploadPath}
       >
         Update license
       </EuiButton>
     }
   />
 );
```

```
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { AddLicense } from './add_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';

import { EuiFlexItem, EuiCard, EuiLink, EuiButton } from '@elastic/eui';

export const RequestTrialExtension = ({ shouldShowRequestTrialExtension }) => {
 if (!shouldShowRequestTrialExtension) {
   return null;
 }
 const description = (
   <span>
     If youd like to continuing using security, machine learning, and our
     other awesome{' '}
     <EuiLink
       href="https://www.elastic.co/subscriptions/xpack"
       target="_blank"
     >
       Platinum features
     </EuiLink>, request an extension now.
   </span>
 );
 return (
   <EuiFlexItem>
     <EuiCard
       title="Extend your trial"
       description={description}
       footer={
         <EuiButton
           data-test-subj="extendTrialButton"
           target="_blank"
           href="https://www.elastic.co/trialextension"
         >
           Extend trial
         </EuiButton>
       }
```

```
    />
  </EuiFlexItem>
 );
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { connect } from 'react-redux';

import { RequestTrialExtension as PresentationComponent } from './request_trial_extension';
import {
 shouldShowRequestTrialExtension
} from '../../../store/reducers/licenseManagement';


const mapStateToProps = state => {
 return {
  shouldShowRequestTrialExtension: shouldShowRequestTrialExtension(state),
 };
};

export const RequestTrialExtension = connect(mapStateToProps)(
 PresentationComponent
);
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { RequestTrialExtension } from './request_trial_extension.container';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { LicenseStatus as PresentationComponent } from './license_status';
import { connect } from 'react-redux';
import { getLicense, getExpirationDateFormatted, isExpired } from '../../../store/reducers/licenseManagement';

const mapStateToProps = (state) => {
 const { isActive, type } = getLicense(state);
 return {
  status: isActive ? 'Active' : 'Inactive',
```

```
  type,
  isExpired: isExpired(state),
  expiryDate: getExpirationDateFormatted(state)
 };
};

export const LicenseStatus = connect(mapStateToProps)(PresentationComponent);
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React, { Fragment } from 'react';

import {
 EuiIcon,
 EuiFlexGroup,
 EuiFlexItem,
 EuiText,
 EuiTitle,
 EuiSpacer,
} from '@elastic/eui';

export class LicenseStatus extends React.PureComponent {
 render() {
  const { isExpired, status, type, expiryDate } = this.props;
  const typeTitleCase = type.charAt(0).toUpperCase() + type.substr(1).toLowerCase();
  let icon;
  let title;
  let message;
  if (isExpired) {
   icon = <EuiIcon color="danger" type="alert" />;
   message = (
    <Fragment>
     Your license expired on <strong>{expiryDate}</strong>
    </Fragment>
   );
   title = `Your ${typeTitleCase} license has expired`;
  } else {
   icon = <EuiIcon color="success" type="checkInCircleFilled" />;
   message = expiryDate ? (
    <Fragment>
     Your license will expire on <strong>{expiryDate}</strong>
    </Fragment>
   ) : (
    <Fragment>Your license will never expire.</Fragment>
   );
```

```
    title = `Your ${typeTitleCase} license is ${status.toLowerCase()}`;
  }
  return (
    <div>
      <EuiFlexGroup justifyContent="spaceAround">
        <EuiFlexItem grow={false}>
          <EuiFlexGroup alignItems="center" gutterSize="s">
            <EuiFlexItem grow={false}>{icon}</EuiFlexItem>
            <EuiFlexItem grow={false}>
              <EuiTitle size="l">
                <h2 data-test-subj="licenseText">{title}</h2>
              </EuiTitle>
            </EuiFlexItem>
          </EuiFlexGroup>
        </EuiFlexItem>
      </EuiFlexGroup>
      <EuiFlexGroup justifyContent="spaceAround">
        <EuiFlexItem grow={false}>
          <span data-test-subj="licenseSubText">
            <EuiText color="subdued">{message}</EuiText>
          </span>
        </EuiFlexItem>
      </EuiFlexGroup>
      <EuiSpacer />
    </div>
  );
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { LicenseStatus } from './license_status.container';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';



import { LicenseStatus } from './license_status';
import { RevertToBasic } from './revert_to_basic';
import { StartTrial } from './start_trial';
import { AddLicense } from './add_license';
```

```
import { RequestTrialExtension } from './request_trial_extension';
import {
 EuiFlexGroup,
 EuiFlexItem,
 EuiSpacer
} from '@elastic/eui';

export const LicenseDashboard = () => {
 return (
   <div className="licManagement__contain">
     <EuiFlexGroup justifyContent="spaceAround">
       <LicenseStatus />
     </EuiFlexGroup>
     <EuiSpacer size="l" />
     <EuiFlexGroup justifyContent="spaceAround">
       <EuiFlexItem>
         <AddLicense />
       </EuiFlexItem>
       <StartTrial />
       <RequestTrialExtension/>
       <RevertToBasic/>
     </EuiFlexGroup>
   </div>
 );
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { LicenseDashboard } from './license_dashboard';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';

import {
 EuiFlexItem,
 EuiCard,
 EuiButton,
 EuiLink,
 EuiOverlayMask,
 EuiConfirmModal,
 EuiText
```

```
} from '@elastic/eui';

export class RevertToBasic extends React.PureComponent {
 cancel = () => {
  this.props.uploadLicenseStatus({});
 };

 acknowledgeModal() {
  const {
   needsAcknowledgement,
   messages: [firstLine, ...messages] = [],
   startBasicLicense,
   cancelStartBasicLicense,
   licenseType
  } = this.props;
  if (!needsAcknowledgement) {
   return null;
  }
  return (
   <EuiOverlayMask>
    <EuiConfirmModal
     title="Confirm Revert to Basic License"
     onCancel={cancelStartBasicLicense}
     onConfirm={() => startBasicLicense(licenseType, true)}
     cancelButtonText="Cancel"
     confirmButtonText="Confirm"
    >
     <div>
      <EuiText>{firstLine}</EuiText>
      <EuiText>
       <ul>
        {messages.map(message => <li key={message}>{message}</li>)}
       </ul>
      </EuiText>
     </div>
    </EuiConfirmModal>
   </EuiOverlayMask>
  );
 }
 render() {
  const {
   licenseType,
   shouldShowRevertToBasicLicense,
   startBasicLicense
  } = this.props;
  if (!shouldShowRevertToBasicLicense) {
   return null;
  }
```

```
  const description = (
    <span>
      Youll revert to our free features and lose access to security, machine
      learning and other{' '}
      <EuiLink
        href="https://www.elastic.co/subscriptions/xpack"
        target="_blank"
      >
        Platinum features
      </EuiLink>.
    </span>
  );

  return (
    <EuiFlexItem>
      {this.acknowledgeModal()}
      <EuiCard
        title="Revert to Basic license"
        description={description}
        footer={
          <EuiButton data-test-subj="revertToBasicButton" onClick={() => startBasicLicense(licenseType)}>
            Revert to Basic
          </EuiButton>
        }
      />
    </EuiFlexItem>
  );
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { connect } from 'react-redux';

import { RevertToBasic as PresentationComponent } from './revert_to_basic';
import {
 startBasicLicenseNeedsAcknowledgement,
 getLicenseType,
 shouldShowRevertToBasicLicense,
 getStartBasicMessages
} from '../../../store/reducers/licenseManagement';
import { startBasicLicense, cancelStartBasicLicense } from '../../../store/actions/start_basic';

const mapStateToProps = state => {
 return {
```

```
  shouldShowRevertToBasicLicense: shouldShowRevertToBasicLicense(state),
  licenseType: getLicenseType(state),
  needsAcknowledgement: startBasicLicenseNeedsAcknowledgement(state),
  messages: getStartBasicMessages(state)
 };
};

const mapDispatchToProps = {
 startBasicLicense,
 cancelStartBasicLicense
};

export const RevertToBasic = connect(mapStateToProps, mapDispatchToProps)(
 PresentationComponent
);
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { RevertToBasic } from './revert_to_basic.container';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';

import {
 EuiButtonEmpty,
 EuiButton,
 EuiFlexGroup,
 EuiFlexItem,
 EuiCard,
 EuiLink,
 EuiOverlayMask,
 EuiText,
 EuiModal,
 EuiModalFooter,
 EuiModalHeader,
 EuiModalBody,
 EuiModalHeaderTitle
} from '@elastic/eui';

import { ELASTIC_WEBSITE_URL, DOC_LINK_VERSION } from 'ui/documentation_links';
import { TelemetryOptIn } from '../../../components/telemetry_opt_in';
```

```
import { optInToTelemetry } from '../../../lib/telemetry';
const esBase = `${ELASTIC_WEBSITE_URL}guide/en/elasticsearch/reference/${DOC_LINK_VERSION}`;
const securityDocumentationLink = `${esBase}/security-settings.html`;


export class StartTrial extends React.PureComponent {
 constructor(props) {
   super(props);
   this.state = { showConfirmation: false };
 }
 componentWillMount() {
   this.props.loadTrialStatus();
 }
 startLicenseTrial = () => {
   const { startLicenseTrial } = this.props;
   if (this.telemetryOptIn.isOptingInToTelemetry()) {
     optInToTelemetry(true);
   }
   startLicenseTrial();
 }
 cancel = () => {
   this.setState({ showConfirmation: false });
 }
 acknowledgeModal() {
   const { showConfirmation } = this.state;
   if (!showConfirmation) {
     return null;
   }

   return (
     <EuiOverlayMask>
       <EuiModal
         className="licManagement__modal"
         onClose={this.cancel}
       >
         <EuiModalHeader>
           <EuiModalHeaderTitle data-test-subj="confirmModalTitleText">
           Start your free 30-day trial
           </EuiModalHeaderTitle>
         </EuiModalHeader>
         <EuiModalBody>
           <EuiText data-test-subj="confirmModalBodyText">
             <div>
               <EuiText>
                 <p>
               This trial is for the full set of{' '}
                   <EuiLink
                     href="https://www.elastic.co/subscriptions/xpack"
```

```
          target="_blank"
        >
      Platinum features
        </EuiLink> of the Elastic Stack. You'll get immediate access to:
      </p>
      <ul>
        <li>Machine learning</li>
        <li>Alerting</li>
        <li>Graph capabilities</li>
        <li>JDBC connectivity for SQL</li>
      </ul>
      <p>
    Security features, such as authentication (native, AD/LDAP, SAML,
    PKI), role-based access control, and auditing, require
    configuration. See the{' '}
        <EuiLink
          href={securityDocumentationLink}
          target="_blank"
        >
      documentation
        </EuiLink>{' '}
    for instructions.
      </p>
      <p>
    By starting this trial, you agree that it is subject to these{' '}
        <EuiLink
          href="https://elastic.co/legal/trial_license"
          target="_blank"
        >
      terms and conditions
        </EuiLink>.
      </p>
    </EuiText>
  </div>
 </EuiText>
</EuiModalBody>

<EuiModalFooter>
 <EuiFlexGroup justifyContent="spaceBetween" alignItems="center">
  <EuiFlexItem grow={false}>
   <TelemetryOptIn isStartTrial={true} ref={(ref) => {this.telemetryOptIn = ref; }}/>
  </EuiFlexItem>
  <EuiFlexItem grow={false} className="licManagement__ieFlex">
   <EuiFlexGroup responsive={false}>
    <EuiFlexItem grow={false} className="licManagement__ieFlex">
     <EuiButtonEmpty
      data-test-subj="confirmModalCancelButton"
      onClick={this.cancel}
```

```
                buttonRef={this.cancelRef}
              >
            Cancel
              </EuiButtonEmpty>
            </EuiFlexItem>
            <EuiFlexItem grow={false} className="licManagement__ieFlex">
              <EuiButton
                data-test-subj="confirmModalConfirmButton"
                onClick={this.startLicenseTrial}
                fill
                buttonRef={this.confirmRef}
                color="primary"
              >
          Start my trial
              </EuiButton>
            </EuiFlexItem>
          </EuiFlexGroup>
        </EuiFlexItem>
      </EuiFlexGroup>
    </EuiModalFooter>
  </EuiModal>
</EuiOverlayMask>
);
}

render() {
  const { shouldShowStartTrial } = this.props;
  if (!shouldShowStartTrial) {
    return null;
  }
  const description = (
    <span>
      Experience what security, machine learning, and all our other{' '}
      <EuiLink
        href="https://www.elastic.co/subscriptions/xpack"
        target="_blank"
      >
        Platinum features
      </EuiLink>{' '}
      have to offer.
    </span>
  );

  const footer = (
    <EuiButton data-test-subj="startTrialButton" onClick={() => this.setState({ showConfirmation: true })}>
      Start trial
    </EuiButton>
  );
```

```
  return (
    <EuiFlexItem>
      {this.acknowledgeModal()}
      <EuiCard
        title="Start a 30-day trial"
        description={description}
        footer={footer}
      />
    </EuiFlexItem>
  );
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { StartTrial } from './start_trial.container';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { connect } from 'react-redux';

import { StartTrial as PresentationComponent } from './start_trial';
import { loadTrialStatus, startLicenseTrial } from '../../../store/actions/start_trial';
import { shouldShowStartTrial } from '../../../store/reducers/licenseManagement';

const mapStateToProps = (state) => {
 return {
   shouldShowStartTrial: shouldShowStartTrial(state),
 };
};

const mapDispatchToProps = {
 loadTrialStatus,
 startLicenseTrial
};

export const StartTrial = connect(mapStateToProps, mapDispatchToProps)(PresentationComponent);
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
export { LicenseDashboard } from './license_dashboard';
export { UploadLicense } from './upload_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React, { Fragment } from 'react';
import { BASE_PATH } from '../../../common/constants';
import {
 EuiButton,
 EuiButtonEmpty,
 EuiFilePicker,
 EuiForm,
 EuiSpacer,
 EuiConfirmModal,
 EuiOverlayMask,
 EuiText,
 EuiTitle,
 EuiFlexGroup,
 EuiFlexItem,
 EuiPageContent,
 EuiPageContentBody,
} from '@elastic/eui';
import { TelemetryOptIn } from '../../components/telemetry_opt_in';
import { optInToTelemetry } from '../../lib/telemetry';
export class UploadLicense extends React.PureComponent {
 componentDidMount() {
   this.props.addUploadErrorMessage('');
 }
 send = acknowledge => {
   const file = this.file;
   const fr = new FileReader();
   fr.onload = ({ target: { result } }) => {
     if (this.telemetryOptIn.isOptingInToTelemetry()) {
       optInToTelemetry(true);
     }
     this.props.uploadLicense(result, this.props.currentLicenseType, acknowledge);
   };
   fr.readAsText(file);
 };

 cancel = () => {
   this.props.uploadLicenseStatus({});
 };
```

```
acknowledgeModal() {
  const { needsAcknowledgement, messages: [firstLine, ...messages] = [] } = this.props;
  if (!needsAcknowledgement) {
    return null;
  }
  return (
    <EuiOverlayMask>
      <EuiConfirmModal
        title="Confirm License Upload"
        onCancel={this.cancel}
        onConfirm={() => this.send(true)}
        cancelButtonText="Cancel"
        confirmButtonText="Confirm"
      >
        <div>
          <EuiText>{firstLine}</EuiText>
          <EuiText>
            <ul>
              {messages.map(message => (
                <li key={message}>{message}</li>
              ))}
            </ul>
          </EuiText>
        </div>
      </EuiConfirmModal>
    </EuiOverlayMask>
  );
}
errorMessage() {
  const { errorMessage } = this.props;
  if (!errorMessage) {
    return null;
  }
  return [errorMessage];
}
handleFile = ([file]) => {
  if (file) {
    this.props.addUploadErrorMessage('');
  }
  this.file = file;
};
submit = event => {
  event.preventDefault();
  if (this.file) {
    this.send();
  } else {
    this.props.addUploadErrorMessage('You must select a license file.');
  }
```

```
};
render() {
  const { currentLicenseType, applying } = this.props;
  return (
    <Fragment>
      <EuiTitle className="eui-textCenter" size="l">
        <h1>Upload your license</h1>
      </EuiTitle>

      <EuiSpacer />

      <EuiPageContent horizontalPosition="center" verticalPosition="center">
        <EuiPageContentBody>
          {this.acknowledgeModal()}

          <EuiText>
            <p>Your license key is a JSON file with a signature attached.</p>
            <p>
              Uploading a license will replace your current{' '}
              <strong>{currentLicenseType.toUpperCase()}</strong> license.
            </p>
          </EuiText>
          <EuiSpacer />
          <EuiForm isInvalid={!!this.errorMessage()} error={this.errorMessage()}>
            <EuiText>
              <EuiFilePicker
                id="licenseFile"
                initialPromptText="Select or drag your license file"
                onChange={this.handleFile}
              />
            </EuiText>
            <EuiSpacer size="m" />
            <TelemetryOptIn
              ref={ref => {
                this.telemetryOptIn = ref;
              }}
            />
            <EuiSpacer size="m" />
            <EuiFlexGroup justifyContent="spaceBetween">
              <EuiFlexItem grow={false}>
                <EuiButtonEmpty href={`#${BASE_PATH}home`}>Cancel</EuiButtonEmpty>
              </EuiFlexItem>
              <EuiFlexItem grow={false}>
                <EuiButton
                  data-test-subj="uploadLicenseButton"
                  fill
                  isLoading={applying}
                  onClick={this.submit}
```

```
            >
              {applying ? 'Uploading...' : 'Upload'}
            </EuiButton>
          </EuiFlexItem>
        </EuiFlexGroup>
      </EuiForm>
    </EuiPageContentBody>
  </EuiPageContent>
 </Fragment>
 );
}
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { UploadLicense } from './upload_license.container';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { connect } from 'react-redux';
import { uploadLicense, uploadLicenseStatus } from "../../store/actions/upload_license";
import { addUploadErrorMessage } from "../../store/actions/add_error_message";

import {
 getUploadErrorMessage,
 getLicenseType,
 isInvalid,
 isApplying,
 uploadNeedsAcknowledgement,
 uploadMessages
} from "../../store/reducers/licenseManagement";
import { UploadLicense as PresentationComponent } from './upload_license';


const mapStateToProps = (state) => {
 return {
  isInvalid: isInvalid(state),
  needsAcknowledgement: uploadNeedsAcknowledgement(state),
  messages: uploadMessages(state),
  errorMessage: getUploadErrorMessage(state),
  applying: isApplying(state),
  currentLicenseType: getLicenseType(state) || ''
```

```
 };
};
const mapDispatchToProps = {
 addUploadErrorMessage,
 uploadLicense,
 uploadLicenseStatus,
};

export const UploadLicense = connect(mapStateToProps, mapDispatchToProps)(PresentationComponent);
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';
import { LicenseDashboard, UploadLicense } from './sections/';
import { Switch, Route } from 'react-router-dom';
import { BASE_PATH } from '../common/constants';
import {
 EuiPage,
 EuiPageBody,
} from '@elastic/eui';

export default () => (
 <EuiPage restrictWidth>
   <EuiPageBody className="licManagement__pageBody">
     <Switch>
       <Route path={`${BASE_PATH}upload_license`} component={UploadLicense}/>
       <Route path={`${BASE_PATH}`} component={LicenseDashboard}/>
     </Switch>
   </EuiPageBody>
 </EuiPage>
);
.licFeature {
 flex-grow: 1;
 background: $euiColorLightestShade;
 min-height: 100vh;
}

.licManagement__pageBody {
 padding-top: $euiSizeL;
 min-height: auto;
}

// EUITODO: Fix modal width/max-width
.licManagement__modal {
 width: 70vw;
```

```
}

.licManagement__narrowText {
 width: 240px;
}

.licManagement__ieFlex {
 flex-shrink: 0;
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { management } from 'ui/management';
import { BASE_PATH } from '../common/constants';

management.getSection('elasticsearch').register('license_management', {
 visible: true,
 display: 'License Management',
 order: 4,
 url: `#${BASE_PATH}home`
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import './management_section';
import './register_route';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { createStore, applyMiddleware, compose } from 'redux';
import thunk from 'redux-thunk';

import { licenseManagement } from './reducers/';

export const licenseManagementStore = (initialState = {}, services = {}) => {
 const enhancers = [ applyMiddleware(thunk.withExtraArgument(services)) ];

 window.__REDUX_DEVTOOLS_EXTENSION__ &&
enhancers.push(window.__REDUX_DEVTOOLS_EXTENSION__());
```

```
 return createStore(
   licenseManagement,
   initialState,
   compose(...enhancers)
 );
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { createAction } from 'redux-actions';
import { startBasic } from '../../lib/es';
import { toastNotifications } from 'ui/notify';

export const startBasicLicenseStatus = createAction(
 'LICENSE_MANAGEMENT_START_BASIC_LICENSE_STATUS'
);

export const cancelStartBasicLicense = createAction(
 'LICENSE_MANAGEMENT_CANCEL_START_BASIC_LICENSE'
);

export const startBasicLicense = (currentLicenseType, ack) => async (
 dispatch,
 getState,
 { xPackInfo }
) => {
 /*eslint camelcase: 0*/
 const { acknowledged, basic_was_started, error_message, acknowledge } = await startBasic(ack);
 if (acknowledged) {
   if (basic_was_started) {
     await xPackInfo.refresh();
     // reload necessary to get left nav to refresh with proper links
     window.location.reload();
   } else {
     return toastNotifications.addDanger(error_message);
   }
 } else {
   //messages coming back in arrays
   const messages = Object.values(acknowledge).slice(1).map((item) => {
     return item[0];
   });
   const first = `Some functionality will be lost if you replace your
       ${currentLicenseType.toUpperCase()} license with a
       BASIC license.  Review the list of features below.`;
   dispatch(startBasicLicenseStatus({ acknowledge: true, messages: [ first, ...messages] }));
```

```
 }

};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { createAction } from 'redux-actions';
import { canStartTrial, startTrial } from '../../lib/es';
import { toastNotifications } from 'ui/notify';

export const trialStatusLoaded = createAction(
 'LICENSE_MANAGEMENT_TRIAL_STATUS_LOADED'
);

export const loadTrialStatus = () => async dispatch => {
 const trialOK = await canStartTrial();
 dispatch(trialStatusLoaded(trialOK));
};

export const startLicenseTrial = () => async (
 dispatch,
 getState,
 { xPackInfo }
) => {
 /*eslint camelcase: 0*/
 const { trial_was_started, error_message } = await startTrial();
 if (trial_was_started) {
   await xPackInfo.refresh();
   // reload necessary to get left nav to refresh with proper links
   window.location.reload();
 } else {
   return toastNotifications.addDanger(error_message);
 }
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { createAction } from 'redux-actions';

export const addUploadErrorMessage =
 createAction('LICENSE_MANAGEMENT_ADD_UPLOAD_ERROR_MESSAGE');
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { createAction } from 'redux-actions';
import { addLicense } from '../actions/add_license';
import { BASE_PATH } from '../../../common/constants/base_path';
import { putLicense } from '../../lib/es';
import { addUploadErrorMessage } from "./add_error_message";

export const uploadLicenseStatus = createAction('LICENSE_MANAGEMENT_UPLOAD_LICENSE_STATUS');

const genericUploadError = 'Error encountered uploading license:';

const dispatchFromResponse = async (response, dispatch, currentLicenseType, newLicenseType, { xPackInfo,
kbnUrl }) => {
 const { error, acknowledged, license_status: licenseStatus, acknowledge } = response;
 if (error) {
   dispatch(uploadLicenseStatus({}));
   dispatch(addUploadErrorMessage(`${genericUploadError} ${error.reason}`));
 } else if (acknowledged) {
   if (licenseStatus === 'invalid') {
     dispatch(uploadLicenseStatus({}));
     dispatch(addUploadErrorMessage('The supplied license is not valid for this product.'));
   } else if (licenseStatus === 'expired') {
     dispatch(uploadLicenseStatus({}));
     dispatch(addUploadErrorMessage('The supplied license has expired.'));
   } else {
     await xPackInfo.refresh();
     dispatch(addLicense(xPackInfo.get('license')));
     dispatch(uploadLicenseStatus({}));
     kbnUrl.change(BASE_PATH);
     // reload necessary to get left nav to refresh with proper links
     window.location.reload();
   }
 } else {
   // first message relates to command line interface, so remove it
   const messages = Object.values(acknowledge).slice(1);
   // messages can be in nested arrays
   const first = `Some functionality will be lost if you replace your
       ${currentLicenseType.toUpperCase()} license with a
       ${newLicenseType.toUpperCase()} license.  Review the list of features below.`;
   dispatch(uploadLicenseStatus({ acknowledge: true, messages: [ first, ...messages] }));
 }

};
```

```
export const uploadLicense = (licenseString, currentLicenseType, acknowledge) => async (dispatch, getState,
services) => {
 dispatch(uploadLicenseStatus({ applying: true }));
 let newLicenseType = null;
 try {
   ({ type: newLicenseType } = JSON.parse(licenseString).license);
 } catch (err) {
   dispatch(uploadLicenseStatus({}));
   return dispatch(addUploadErrorMessage(`${genericUploadError} Check your license file.`));
 }
 try {
   const response = await putLicense(licenseString, acknowledge);
   await dispatchFromResponse(response, dispatch, currentLicenseType, newLicenseType, services);
 } catch (err) {
   const message = (err.responseJSON && err.responseJSON.error.reason) ? err.responseJSON.error.reason :
'Unknown error.';
   dispatch(uploadLicenseStatus({}));
   dispatch(addUploadErrorMessage(`${genericUploadError} ${message}`));
 }
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { createAction } from 'redux-actions';

export const addLicense =
 createAction('LICENSE_MANAGEMENT_ADD_LICENSE');
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { uploadLicense } from './upload_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { handleActions } from 'redux-actions';

import { uploadLicenseStatus } from '../actions/upload_license';
```

```
export const uploadStatus = handleActions({
 [uploadLicenseStatus](state, { payload }) {
   return payload;
 }
}, {});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { handleActions } from 'redux-actions';

import { startBasicLicenseStatus, cancelStartBasicLicense } from '../actions/start_basic';

export const startBasicStatus = handleActions({
 [startBasicLicenseStatus](state, { payload }) {
   return payload;
 },
 [cancelStartBasicLicense]() {
   return {};
 }
}, {});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { handleActions } from 'redux-actions';

import { trialStatusLoaded } from '../actions/start_trial';

export const trialStatus = handleActions({
 [trialStatusLoaded](state, { payload }) {
   return {
     canStartTrial: payload
   };
 },
}, {});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { handleActions } from 'redux-actions';
```

```
import { addLicense } from '../actions/add_license';

export const license = handleActions({
 [addLicense](state, { payload }) {
   return payload;
 }
}, {});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { combineReducers } from 'redux';
import { license } from './license';
import { uploadStatus } from './upload_status';
import { startBasicStatus } from './start_basic_license_status';
import { uploadErrorMessage } from './upload_error_message';
import { trialStatus } from './trial_status';
import moment from 'moment-timezone';

export const WARNING_THRESHOLD_IN_DAYS = 25;

export const licenseManagement = combineReducers({
 license,
 uploadStatus,
 uploadErrorMessage,
 trialStatus,
 startBasicStatus
});

export const getLicense = state => {
 return state.license;
};

export const getLicenseType = state => {
 return getLicense(state).type;
};

export const getExpirationMillis = state => {
 return getLicense(state).expiryDateInMillis;
};

export const getExpirationDate = state => {
 //basic licenses do not expire
 if (getLicenseType(state) === 'basic') {
   return null;
 }
```

```
 const expirationMillis = getExpirationMillis(state);
 if (expirationMillis) {
   return moment.tz(getExpirationMillis(state), moment.tz.guess());
 } else {
   return null;
 }
};

export const getExpirationDateFormatted = state => {
 const expirationDate = getExpirationDate(state);
 return expirationDate ? expirationDate.format('LLL z') : null;
};

export const isExpired = state => {
 return new Date().getTime() > getExpirationMillis(state);
};

export const isImminentExpiration = state => {
 const now = new Date();
 const expirationDate = getExpirationDate(state);
 return (
   expirationDate &&
   expirationDate.isAfter(now) &&
   expirationDate.diff(now, 'days') <= WARNING_THRESHOLD_IN_DAYS
 );
};

export const shouldShowRevertToBasicLicense = state => {
 const { type } = getLicense(state);
 return type === 'trial' || isImminentExpiration(state) || isExpired(state);
};

export const uploadNeedsAcknowledgement = state => {
 return !!state.uploadStatus.acknowledge;
};

export const isApplying = state => {
 return !!state.uploadStatus.applying;
};

export const uploadMessages = state => {
 return state.uploadStatus.messages;
};

export const isInvalid = state => {
 return !!state.uploadStatus.invalid;
};
```

```
export const getUploadErrorMessage = state => {
 return state.uploadErrorMessage;
};

export const shouldShowStartTrial = state => {
 const licenseType = getLicenseType(state);
 return (
   state.trialStatus.canStartTrial &&
   licenseType !== 'trial' &&
   //don't show for platinum unless it is expired
   (licenseType !== 'platinum' || isExpired(state))
 );
};

export const shouldShowRequestTrialExtension = state => {
 if (state.trialStatus.canStartTrial) {
   return false;
 }
 const { type } = getLicense(state);
 return type !== 'platinum' || isExpired(state);
};

export const startTrialError = state => {
 return state.trialStatus.startTrialError;
};

export const startBasicLicenseNeedsAcknowledgement = state => {
 return !!state.startBasicStatus.acknowledge;
};

export const getStartBasicMessages = state => {
 return state.startBasicStatus.messages;
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { handleActions } from 'redux-actions';

import { addUploadErrorMessage } from '../actions/add_error_message';

export const uploadErrorMessage = handleActions({
 [addUploadErrorMessage](state, { payload }) {
   return payload;
 }
}, '');
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { licenseManagement } from './licenseManagement';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { licenseManagementStore } from './store';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import chrome from 'ui/chrome';

import $ from 'jquery';

export function putLicense(license, acknowledge) {
 const options = {
   url: `${chrome.addBasePath('/api/license')}${acknowledge ? '?acknowledge=true' : ''}`,
   data: license,
   contentType: 'application/json',
   cache: false,
   crossDomain: true,
   type: 'PUT',
 };

 return $.ajax(options);
}

export function startBasic(acknowledge) {
 const options = {
   url: `${chrome.addBasePath('/api/license/start_basic')}${acknowledge ? '?acknowledge=true' : ''}`,
   contentType: 'application/json',
   cache: false,
   crossDomain: true,
   type: 'POST',
 };

 return $.ajax(options);
}
```

```
export function startTrial() {
 const options = {
   url: chrome.addBasePath('/api/license/start_trial'),
   contentType: 'application/json',
   cache: false,
   crossDomain: true,
   type: 'POST',
 };

 return $.ajax(options);
}

export function canStartTrial() {
 const options = {
   url: chrome.addBasePath('/api/license/start_trial'),
   contentType: 'application/json',
   cache: false,
   crossDomain: true,
   type: 'GET',
 };

 return $.ajax(options);
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { fetchTelemetry } from '../../../xpack_main/public/hacks/fetch_telemetry';
export { PRIVACY_STATEMENT_URL } from '../../../xpack_main/common/constants';
export { TelemetryOptInProvider } from '../../../xpack_main/public/services/telemetry_opt_in';
export { OptInExampleFlyout } from '../../../xpack_main/public/components';

let telemetryEnabled;
let httpClient;
let telemetryOptInService;
export const setTelemetryEnabled = (isTelemetryEnabled) => {
 telemetryEnabled = isTelemetryEnabled;
};
export const setHttpClient = (anHttpClient) => {
 httpClient = anHttpClient;
};
export const setTelemetryOptInService = (aTelemetryOptInService) => {
 telemetryOptInService = aTelemetryOptInService;
};
export const optInToTelemetry = async (enableTelemetry) => {
```

```
 await telemetryOptInService.setOptIn(enableTelemetry);
};
export const showTelemetryOptIn = () => {
 return telemetryEnabled && !telemetryOptInService.getOptIn();
};
export const getTelemetryFetcher = () => {
 return fetchTelemetry(httpClient);
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is not available', () => {
   beforeEach(() => mockLicenseInfo.isAvailable = () => false);

   it('should set showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).showAppLink).to.be(true);
   });

   it('should set enableLinks to false', () => {
     expect(checkLicense(mockLicenseInfo).enableAppLink).to.be(false);
   });
 });

 describe('license information is available', () => {
   beforeEach(() => {
     mockLicenseInfo.isAvailable = () => true;
     set(mockLicenseInfo, 'license.getType', () => 'basic');
   });

   describe('& license is trial, standard, gold, platinum', () => {
     beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

     describe('& license is active', () => {
       beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

       it ('should set showLinks to true', () => {
```

```
      expect(checkLicense(mockLicenseInfo).showAppLink).to.be(true);
    });

    it ('should set enableLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).enableAppLink).to.be(true);
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it ('should set showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).showAppLink).to.be(true);
    });

    it ('should set enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).enableAppLink).to.be(false);
    });
  });
});

describe('& license is basic', () => {
  beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => false));

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it ('should set showLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).showAppLink).to.be(false);
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it ('should set showLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).showAppLink).to.be(false);
    });
  });
});
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
export function checkLicense(xpackLicenseInfo) {

 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     showAppLink: true,
     enableAppLink: false,
     message: 'Search Profiler is unavailable - license information is not available at this time.'
   };
 }

 const isLicenseActive = xpackLicenseInfo.license.isActive();
 let message;
 if (!isLicenseActive) {
   message = `Search Profiler is unavailable - license has expired.`;
 }

 if (xpackLicenseInfo.license.isOneOf([ 'trial', 'basic', 'standard', 'gold', 'platinum' ])) {
   return {
     showAppLink: true,
     enableAppLink: isLicenseActive,
     message
   };
 }

 message = `Search Profiler is unavailable for the current ${xpackLicenseInfo.license.getType()} license. Please
upgrade your license.`;
 return {
   showAppLink: false,
   enableAppLink: false,
   message
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { registerRefreshRoute } from './register_refresh_route';

export function registerLicenseRoutes(server) {
 registerRefreshRoute(server);
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
import { licensePreRoutingFactory } from'../../../lib/license_pre_routing_factory';

/*
In order for the client to have the most up-to-date snapshot of the current license,
it needs to make a round-trip to the kibana server. This refresh endpoint is provided
for when the client needs to check the license, but doesn't need to pull data from the
server for any reason, i.e., when adding a new watch.
*/
export function registerRefreshRoute(server) {
 const licensePreRouting = licensePreRoutingFactory(server);

 server.route({
   path: '/api/watcher/license/refresh',
   method: 'GET',
   handler: (request, reply) => {
     reply({ success: true });
   },
   config: {
     pre: [ licensePreRouting ]
   }
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { registerLicenseRoutes } from './register_license_routes';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { mirrorPluginStatus } from '../../../../../server/lib/mirror_plugin_status';
import { checkLicense } from '../check_license';
import { PLUGIN } from '../../../common/constants';

export function registerLicenseChecker(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const watcherPlugin = server.plugins.watcher;

 mirrorPluginStatus(xpackMainPlugin, watcherPlugin);
 xpackMainPlugin.status.once('green', () => {
   // Register a function that is called whenever the xpack info changes,
   // to re-compute the license check results for this plugin
```

```
  xpackMainPlugin.info.feature(PLUGIN.ID).registerLicenseCheckResultsGenerator(checkLicense);
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { registerLicenseChecker } from './register_license_checker';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { once } from 'lodash';
import { wrapCustomError } from '../error_wrappers';
import { PLUGIN } from '../../../common/constants';

export const licensePreRoutingFactory = once((server) => {
 const xpackMainPlugin = server.plugins.xpack_main;

 // License checking and enable/disable logic
 function licensePreRouting(request, reply) {
   const licenseCheckResults = xpackMainPlugin.info.feature(PLUGIN.ID).getLicenseCheckResults();
   if (!licenseCheckResults.isAvailable) {
     const error = new Error(licenseCheckResults.message);
     const statusCode = 403;
     const wrappedError = wrapCustomError(error, statusCode);
     reply(wrappedError);
   } else {
     reply();
   }
 }

 return licensePreRouting;
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { licensePreRoutingFactory } from '../license_pre_routing_factory';

describe('license_pre_routing_factory', () => {
```

```
describe('#reportingFeaturePreRoutingFactory', () => {
  let mockServer;
  let mockLicenseCheckResults;

  beforeEach(() => {
    mockServer = {
      plugins: {
        xpack_main: {
          info: {
            feature: () => ({
              getLicenseCheckResults: () => mockLicenseCheckResults
            })
          }
        }
      }
    };
  });

  it('only instantiates one instance per server', () => {
    const firstInstance = licensePreRoutingFactory(mockServer);
    const secondInstance = licensePreRoutingFactory(mockServer);

    expect(firstInstance).to.be(secondInstance);
  });

  describe('isAvailable is false', () => {
    beforeEach(() => {
      mockLicenseCheckResults = {
        isAvailable: false
      };
    });

    it ('replies with 403', (done) => {
      const licensePreRouting = licensePreRoutingFactory(mockServer);
      const stubRequest = {};
      licensePreRouting(stubRequest, (response) => {
        expect(response).to.be.an(Error);
        expect(response.isBoom).to.be(true);
        expect(response.output.statusCode).to.be(403);
        done();
      });
    });
  });

  describe('isAvailable is true', () => {
    beforeEach(() => {
      mockLicenseCheckResults = {
        isAvailable: true
```

```
      };
    });

    it ('replies with nothing', (done) => {
      const licensePreRouting = licensePreRoutingFactory(mockServer);
      const stubRequest = {};
      licensePreRouting(stubRequest, (response) => {
        expect(response).to.be(undefined);
        done();
      });
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { licensePreRoutingFactory } from './license_pre_routing_factory';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is undefined', () => {
   beforeEach(() => mockLicenseInfo = undefined);

   it('should set isAvailable to false', () => {
     expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
   });

   it('should set showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
   });

   it('should set enableLinks to false', () => {
```

```
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('license information is not available', () => {
  beforeEach(() => mockLicenseInfo.isAvailable = () => false);

  it('should set isAvailable to false', () => {
    expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
  });

  it('should set showLinks to true', () => {
    expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
  });

  it('should set enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('license information is available', () => {
  beforeEach(() => {
    mockLicenseInfo.isAvailable = () => true;
    set(mockLicenseInfo, 'license.getType', () => 'basic');
  });

  describe('& license is trial, standard, gold, platinum', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

    describe('& license is active', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

      it('should set isAvailable to true', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
      });

      it ('should set showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
      });
```

```
    it ('should set enableLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
    });

    it('should not set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it('should set isAvailable to false', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
    });

    it ('should set showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
    });

    it ('should set enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
});

describe('& license is basic', () => {
  beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => false));

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it('should set isAvailable to false', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
    });

    it ('should set showLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
```

```
    describe('& license is expired', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

      it('should set isAvailable to false', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
      });

      it ('should set showLinks to false', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(false);
      });

      it('should set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
      });
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { i18n } from '@kbn/i18n';

export function checkLicense(xpackLicenseInfo) {
 // If, for some reason, we cannot get the license information
 // from Elasticsearch, assume worst case and disable the Watcher UI
 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     isAvailable: false,
     showLinks: true,
     enableLinks: false,
     message: i18n.translate('xpack.watcher.checkLicense.licenseInformationNotAvailableTextMessage', {
       defaultMessage: 'You cannot use {watcher} because license information is not available at this time.',
       values: {
         watcher: 'Watcher'
       }
     }),
   };
 }

 const VALID_LICENSE_MODES = [
   'trial',
   'standard',
   'gold',
```

```
  'platinum'
];

const isLicenseModeValid = xpackLicenseInfo.license.isOneOf(VALID_LICENSE_MODES);
const isLicenseActive = xpackLicenseInfo.license.isActive();
const licenseType = xpackLicenseInfo.license.getType();

// License is not valid
if (!isLicenseModeValid) {
  return {
    isAvailable: false,
    showLinks: false,
    message: i18n.translate('xpack.watcher.checkLicense.licenseDoesNotSupportTextMessage', {
      defaultMessage: 'Your {licenseType} license does not support {watcher}. Please upgrade your license.',
      values: {
        licenseType,
        watcher: 'Watcher'
      },
    }),
  };
}

// License is valid but not active
if (!isLicenseActive) {
  return {
    isAvailable: false,
    showLinks: true,
    enableLinks: false,
    message: i18n.translate('xpack.watcher.server.checkLicense.licenseExpiredTextMessage', {
      defaultMessage: 'You cannot use {watcher} because your {licenseType} license has expired',
      values: {
        licenseType,
        watcher: 'Watcher'
      },
    }),
  };
}

// License is valid and active
return {
  isAvailable: true,
  showLinks: true,
  enableLinks: true
};
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
```

```
* you may not use this file except in compliance with the Elastic License.
*/

export { checkLicense } from './check_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { uiModules } from 'ui/modules';
import { XPackInfoProvider } from 'plugins/xpack_main/services/xpack_info';
import 'ui/url';
import { LicenseService } from './license_service';

uiModules.get('xpack/watcher')
 .factory('xpackWatcherLicenseService', ($injector) => {
   const $http = $injector.get('$http');
   const Private = $injector.get('Private');
   const xpackInfoService = Private(XPackInfoProvider);
   const kbnUrlService = $injector.get('kbnUrl');
   const $timeout = $injector.get('$timeout');

   return new LicenseService(xpackInfoService, kbnUrlService, $timeout, $http);
 });
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';
import { toastNotifications } from 'ui/notify';
import { MarkdownSimple } from 'ui/markdown';
import chrome from 'ui/chrome';
import { PLUGIN, ROUTES } from '../../../common/constants';

export class LicenseService {
 constructor(xpackInfoService, kbnUrlService, $timeout, $http) {
   this.$http = $http;
   this.xpackInfoService = xpackInfoService;
   this.kbnUrlService = kbnUrlService;
   this.$timeout = $timeout;
   this.basePath = chrome.addBasePath(ROUTES.API_ROOT);
 }

 get showLinks() {
   return Boolean(this.xpackInfoService.get(`features.${PLUGIN.ID}.showLinks`));
```

```
}

get enableLinks() {
  return Boolean(this.xpackInfoService.get(`features.${PLUGIN.ID}.enableLinks`));
}

get isAvailable() {
  return Boolean(this.xpackInfoService.get(`features.${PLUGIN.ID}.isAvailable`));
}

get message() {
  return this.xpackInfoService.get(`features.${PLUGIN.ID}.message`);
}

notifyAndRedirect() {
  toastNotifications.addDanger({
    title: <MarkdownSimple>{this.xpackInfoService.get(`features.${PLUGIN.ID}.message`)}</MarkdownSimple>,
  });
  this.kbnUrlService.redirect('/management');
}

/**
 * @param opts Object options
 * @param opts.onValid Function To execute when license is valid. Optional; default = noop
 * @param opts.onInvalid Function To execute when license is invalid. Optional; default = noop
 */
checkValidity() {
  return new Promise((resolve, reject) => {
    this.$timeout(() => {
      if (this.isAvailable) {
        return resolve();
      }

      this.notifyAndRedirect();
      return reject();
    }, 10); // To allow latest XHR call to update license info
  });
}

refreshLicense() {
  return this.$http.get(`${this.basePath}/license/refresh`)
    .then(response => {
      return response.data.success;
    });
}
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
```

```
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import './license_service.factory';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { get } from 'lodash';

/**
* "View" for XPack Info license information.
*/
export class XPackInfoLicense {
 /**
  * Function that retrieves license information from the XPack info object.
  * @type {Function}
  * @private
  */
 _getRawLicense = null;

 constructor(getRawLicense) {
   this._getRawLicense = getRawLicense;
 }

 /**
  * Returns unique identifier of the license.
  * @returns {string|undefined}
  */
 getUid() {
   return get(this._getRawLicense(), 'uid');
 }

 /**
  * Indicates whether license is still active.
  * @returns {boolean}
  */
 isActive() {
   return get(this._getRawLicense(), 'status') === 'active';
 }

 /**
  * Returns license expiration date in ms.
  *
  * Note: A basic license created after 6.3 will have no expiration, thus returning undefined.
```

```
 *
 * @returns {number|undefined}
 */
getExpiryDateInMillis() {
  return get(this._getRawLicense(), 'expiry_date_in_millis');
}

/**
 * Checks if the license is represented in a specified license list.
 * @param {String} candidateLicenses List of the licenses to check against.
 * @returns {boolean}
 */
isOneOf(candidateLicenses) {
  if (!Array.isArray(candidateLicenses)) {
    candidateLicenses = [candidateLicenses];
  }

  return candidateLicenses.includes(get(this._getRawLicense(), 'mode'));
}

/**
 * Returns type of the license (basic, gold etc.).
 * @returns {string|undefined}
 */
getType() {
  return get(this._getRawLicense(), 'type');
}

/**
 * Returns mode of the license (basic, gold etc.). This is the "effective" type of the license.
 * @returns {string|undefined}
 */
getMode() {
  return get(this._getRawLicense(), 'mode');
}

/**
 * Determine if the current license is active and the supplied {@code type}.
 *
 * @param {Function} typeChecker The license type checker.
 * @returns {boolean}
 */
isActiveLicense(typeChecker) {
  const license = this._getRawLicense();

  return get(license, 'status') === 'active' && typeChecker(get(license, 'mode'));
}
```

```
 /**
  * Determine if the license is an active, basic license.
  *
  * Note: This also verifies that the license is active. Therefore it is not safe to assume that !isBasic() ===
isNotBasic().
  *
  * @returns {boolean}
  */
 isBasic() {
   return this.isActiveLicense(mode => mode === 'basic');
 }

 /**
  * Determine if the license is an active, non-basic license (e.g., standard, gold, platinum, or trial).
  *
  * Note: This also verifies that the license is active. Therefore it is not safe to assume that !isBasic() ===
isNotBasic().
  *
  * @returns {boolean}
  */
 isNotBasic() {
   return this.isActiveLicense(mode => mode !== 'basic');
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { XPackInfoLicense } from './xpack_info_license';

function getXPackInfoLicense(getRawLicense) {
 return new XPackInfoLicense(getRawLicense);
}

describe('XPackInfoLicense', () => {

 const xpackInfoLicenseUndefined = getXPackInfoLicense(() => { });
 let xpackInfoLicense;
 let getRawLicense;

 beforeEach(() => {
   getRawLicense = jest.fn();
   xpackInfoLicense = getXPackInfoLicense(getRawLicense);
 });

 test('getUid returns uid field', () => {
```

```
  const uid = 'abc123';

  getRawLicense.mockReturnValue({ uid });

  expect(xpackInfoLicense.getUid()).toBe(uid);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  expect(xpackInfoLicenseUndefined.getUid()).toBe(undefined);
});

test('isActive returns true if status is active', () => {
  getRawLicense.mockReturnValue({ status: 'active' });

  expect(xpackInfoLicense.isActive()).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(1);
});

test('isActive returns false if status is not active', () => {
  getRawLicense.mockReturnValue({ status: 'aCtIvE' }); // needs to match exactly

  expect(xpackInfoLicense.isActive()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  expect(xpackInfoLicenseUndefined.isActive()).toBe(false);
});

test('getExpiryDateInMillis returns expiry_date_in_millis', () => {
  getRawLicense.mockReturnValue({ expiry_date_in_millis: 123 });

  expect(xpackInfoLicense.getExpiryDateInMillis()).toBe(123);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  expect(xpackInfoLicenseUndefined.getExpiryDateInMillis()).toBe(undefined);
});

test('isOneOf returns true of the mode includes one of the types', () => {
  getRawLicense.mockReturnValue({ mode: 'platinum' });

  expect(xpackInfoLicense.isOneOf('platinum')).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  expect(xpackInfoLicense.isOneOf([ 'platinum' ])).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(2);
  expect(xpackInfoLicense.isOneOf([ 'gold', 'platinum' ])).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(3);
  expect(xpackInfoLicense.isOneOf([ 'platinum', 'gold' ])).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(4);
  expect(xpackInfoLicense.isOneOf([ 'basic', 'gold' ])).toBe(false);
```

```
  expect(getRawLicense).toHaveBeenCalledTimes(5);
  expect(xpackInfoLicense.isOneOf([ 'basic' ])).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(6);

  expect(xpackInfoLicenseUndefined.isOneOf([ 'platinum', 'gold' ])).toBe(false);
});

test('getType returns the type', () => {
  getRawLicense.mockReturnValue({ type: 'basic' });

  expect(xpackInfoLicense.getType()).toBe('basic');
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  getRawLicense.mockReturnValue({ type: 'gold' });

  expect(xpackInfoLicense.getType()).toBe('gold');
  expect(getRawLicense).toHaveBeenCalledTimes(2);

  expect(xpackInfoLicenseUndefined.getType()).toBe(undefined);
});

test('getMode returns the mode', () => {
  getRawLicense.mockReturnValue({ mode: 'basic' });

  expect(xpackInfoLicense.getMode()).toBe('basic');
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  getRawLicense.mockReturnValue({ mode: 'gold' });

  expect(xpackInfoLicense.getMode()).toBe('gold');
  expect(getRawLicense).toHaveBeenCalledTimes(2);

  expect(xpackInfoLicenseUndefined.getMode()).toBe(undefined);
});

test('isActiveLicense returns the true if active and typeChecker matches', () => {
  const expectAbc123 = type => type === 'abc123';

  getRawLicense.mockReturnValue({ status: 'active', mode: 'abc123' });

  expect(xpackInfoLicense.isActiveLicense(expectAbc123)).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  getRawLicense.mockReturnValue({ status: 'NOTactive', mode: 'abc123' });

  expect(xpackInfoLicense.isActiveLicense(expectAbc123)).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(2);
```

```
  getRawLicense.mockReturnValue({ status: 'NOTactive', mode: 'NOTabc123' });

  expect(xpackInfoLicense.isActiveLicense(expectAbc123)).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(3);

  getRawLicense.mockReturnValue({ status: 'active', mode: 'NOTabc123' });

  expect(xpackInfoLicense.isActiveLicense(expectAbc123)).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(4);

  expect(xpackInfoLicenseUndefined.isActive(expectAbc123)).toBe(false);
});

test('isBasic returns the true if active and basic', () => {
  getRawLicense.mockReturnValue({ status: 'active', mode: 'basic' });

  expect(xpackInfoLicense.isBasic()).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  getRawLicense.mockReturnValue({ status: 'NOTactive', mode: 'gold' });

  expect(xpackInfoLicense.isBasic()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(2);

  getRawLicense.mockReturnValue({ status: 'NOTactive', mode: 'trial' });

  expect(xpackInfoLicense.isBasic()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(3);

  getRawLicense.mockReturnValue({ status: 'active', mode: 'platinum' });

  expect(xpackInfoLicense.isBasic()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(4);

  expect(xpackInfoLicenseUndefined.isBasic()).toBe(false);
});


test('isNotBasic returns the true if active and not basic', () => {
  getRawLicense.mockReturnValue({ status: 'active', mode: 'platinum' });

  expect(xpackInfoLicense.isNotBasic()).toBe(true);
  expect(getRawLicense).toHaveBeenCalledTimes(1);

  getRawLicense.mockReturnValue({ status: 'NOTactive', mode: 'gold' });

  expect(xpackInfoLicense.isNotBasic()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(2);
```

```
  getRawLicense.mockReturnValue({ status: 'NOTactive', mode: 'trial' });

  expect(xpackInfoLicense.isNotBasic()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(3);

  getRawLicense.mockReturnValue({ status: 'active', mode: 'basic' });

  expect(xpackInfoLicense.isNotBasic()).toBe(false);
  expect(getRawLicense).toHaveBeenCalledTimes(4);

  expect(xpackInfoLicenseUndefined.isNotBasic()).toBe(false);
 });

});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { get } from 'lodash';

/**
* Determine if an API for Cluster Alerts should respond based on the license and configuration of the monitoring
cluster.
*
* Note: This does not guarantee that any production cluster has a valid license; only that Cluster Alerts in general
can be used!
*
* @param  {Object} server Server object containing config and plugins
* @return {Boolean} {@code true} to indicate that cluster alerts can be used.
*/
export function verifyMonitoringLicense(server) {
 const config = server.config();

 // if cluster alerts are enabled, then ensure that we can use it according to the license
 if (config.get('xpack.monitoring.cluster_alerts.enabled')) {
  const xpackInfo = get(server.plugins.monitoring, 'info');
  if (xpackInfo) {
   const monitoringCluster = xpackInfo.feature('monitoring').getLicenseCheckResults();

   return {
    enabled: monitoringCluster.clusterAlerts.enabled,
    message: monitoringCluster.message
   };
  }
```

```
  return {
    enabled: false,
    message: 'Status of Cluster Alerts feature could not be determined.'
  };
 }

 return {
  enabled: false,
  message: 'Cluster Alerts feature is disabled.'
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { verifyMonitoringLicense } from '../verify_monitoring_license';
import expect from 'expect.js';
import sinon from 'sinon';

describe('Monitoring Verify License', () => {
 describe('Disabled by Configuration', () => {
  const get = sinon.stub().withArgs('xpack.monitoring.cluster_alerts.enabled').returns(false);
  const server = { config: sinon.stub().returns({ get }) };

  it('verifyMonitoringLicense returns false without checking the license', () => {
    const verification = verifyMonitoringLicense(server);

    expect(verification.enabled).to.be(false);
    expect(verification.message).to.be('Cluster Alerts feature is disabled.');

    expect(get.withArgs('xpack.monitoring.cluster_alerts.enabled').calledOnce).to.be(true);
  });
 });

 describe('Enabled by Configuration', () => {
  it('verifyMonitoringLicense returns false if enabled by configuration, but not by license', () => {
    const get = sinon.stub().withArgs('xpack.monitoring.cluster_alerts.enabled').returns(true);
    const server = {
      config: sinon.stub().returns({ get }),
      plugins: { monitoring: { info: { } } }
    };
    const getLicenseCheckResults = sinon.stub().returns({ clusterAlerts: { enabled: false }, message: 'failed!!' });
    const feature = sinon.stub().withArgs('monitoring').returns({ getLicenseCheckResults });

    server.plugins.monitoring.info = { feature };
```

```
    const verification = verifyMonitoringLicense(server);

    expect(verification.enabled).to.be(false);
    expect(verification.message).to.be('failed!!');

    expect(get.withArgs('xpack.monitoring.cluster_alerts.enabled').calledOnce).to.be(true);
    expect(feature.withArgs('monitoring').calledOnce).to.be(true);
    expect(getLicenseCheckResults.calledOnce).to.be(true);
  });

  it('verifyMonitoringLicense returns true if enabled by configuration and by license', () => {
    const get = sinon.stub().withArgs('xpack.monitoring.cluster_alerts.enabled').returns(true);
    const server = {
      config: sinon.stub().returns({ get }),
      plugins: { monitoring: { info: { } } }
    };
    const getLicenseCheckResults = sinon.stub().returns({ clusterAlerts: { enabled: true } });
    const feature = sinon.stub().withArgs('monitoring').returns({ getLicenseCheckResults });

    server.plugins.monitoring.info = { feature };

    const verification = verifyMonitoringLicense(server);

    expect(verification.enabled).to.be(true);
    expect(verification.message).to.be.undefined;

    expect(get.withArgs('xpack.monitoring.cluster_alerts.enabled').calledOnce).to.be(true);
    expect(feature.withArgs('monitoring').calledOnce).to.be(true);
    expect(getLicenseCheckResults.calledOnce).to.be(true);
  });
});

it('Monitoring feature info cannot be determined', () => {
  const get = sinon.stub().withArgs('xpack.monitoring.cluster_alerts.enabled').returns(true);
  const server = {
    config: sinon.stub().returns({ get }),
    plugins: { monitoring: undefined } // simulate race condition
  };

  const verification = verifyMonitoringLicense(server);

  expect(verification.enabled).to.be(false);
  expect(verification.message).to.be('Status of Cluster Alerts feature could not be determined.');

  expect(get.withArgs('xpack.monitoring.cluster_alerts.enabled').calledOnce).to.be(true);
});
});
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { checkLicense, checkLicenseGenerator } from '../check_license';
import expect from 'expect.js';
import sinon from 'sinon';

describe('Monitoring Check License', () => {
 describe('License undeterminable', () => {
   it('null active license - results false with a message', () => {
     const result = checkLicense(null, true, 'test-cluster-abc');
     expect(result).to.eql({
       clusterAlerts: { enabled: false },
       message: `Cluster Alerts are not displayed because the [test-cluster-abc] cluster's license could not be
determined.`
     });
   });
 });

 describe('Inactive license', () => {
   it('platinum inactive license - results false with a message', () => {
     const result = checkLicense('platinum', false, 'test-cluster-def');
     expect(result).to.eql({
       clusterAlerts: { enabled: false },
       message: `Cluster Alerts are not displayed because the [test-cluster-def] cluster's current license [platinum] is
not active.`
     });
   });
 });

 describe('Active license', () => {
   describe('Unsupported license types', () => {
     it('basic active license - results false with a message', () => {
       const result = checkLicense('basic', true, 'test-cluster-ghi');
       expect(result).to.eql({
         clusterAlerts: { enabled: false },
         message: `Cluster Alerts are not displayed if Watcher is disabled or the [test-cluster-ghi] cluster's current
license is basic.`
       });
     });
   });

   describe('Supported license types', () => {
     it('standard active license - results true with no message', () => {
       const result = checkLicense('standard', true, 'test-cluster-jkl');
```

```
      expect(result).to.eql({
        clusterAlerts: { enabled: true }
      });
    });

    it('gold active license - results true with no message', () => {
      const result = checkLicense('gold', true, 'test-cluster-mno');
      expect(result).to.eql({
        clusterAlerts: { enabled: true }
      });
    });

    it('platinum active license - results true with no message', () => {
      const result = checkLicense('platinum', true, 'test-cluster-pqr');
      expect(result).to.eql({
        clusterAlerts: { enabled: true }
      });
    });

    describe('Watcher is not enabled', () => {
      it('platinum active license - watcher disabled - results false with message', () => {
        const result = checkLicense('platinum', true, 'test-cluster-pqr', false);
        expect(result).to.eql({
          clusterAlerts: { enabled: false },
          message: 'Cluster alerts is not enabled because Watcher is disabled.'
        });
      });
    });
  });
});

describe('XPackInfo checkLicenseGenerator', () => {
  it('with deleted license', () => {
    const expected = 123;
    const checker = sinon.stub().returns(expected);
    const result = checkLicenseGenerator(null, checker);

    expect(result).to.be(expected);
    expect(checker.withArgs(undefined, false, 'monitoring', false).called).to.be(true);
  });

  it('license without watcher', () => {
    const expected = 123;
    const xpackInfo = {
      license: {
        getType: () => 'fake-type',
        isActive: () => true
      },
```

```
      feature: () => null
    };
    const checker = sinon.stub().returns(expected);
    const result = checkLicenseGenerator(xpackInfo, checker);

    expect(result).to.be(expected);
    expect(checker.withArgs('fake-type', true, 'monitoring', false).called).to.be(true);
  });

  it('mock license with watcher', () => {
    const expected = 123;
    const feature = sinon.stub().withArgs('watcher').returns({ isEnabled: () => true });
    const xpackInfo = {
      license: {
        getType: () => 'another-type',
        isActive: () => true
      },
      feature
    };
    const checker = sinon.stub().returns(expected);
    const result = checkLicenseGenerator(xpackInfo, checker);

    expect(result).to.be(expected);
    expect(feature.withArgs('watcher').calledOnce).to.be(true);
    expect(checker.withArgs('another-type', true, 'monitoring', true).called).to.be(true);
  });

  it('platinum license with watcher', () => {
    const xpackInfo = {
      license: {
        getType: () => 'platinum',
        isActive: () => true
      },
      feature: () => {
        return {
          isEnabled: () => true
        };
      }
    };
    const result = checkLicenseGenerator(xpackInfo);

    expect(result).to.eql({ clusterAlerts: { enabled: true } });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
```

```
* you may not use this file except in compliance with the Elastic License.
*/

import { includes } from 'lodash';

/**
* Function to do the work of checking license for cluster alerts feature support
* Can be used to power XpackInfo license check results as well as checking license of monitored clusters
*
* @param {String} type License type if a valid license. {@code null} if license was deleted.
* @param {Boolean} active Indicating that the overall license is active
* @param {String} clusterSource 'monitoring' or 'production'
* @param {Boolean} watcher {@code true} if Watcher is provided (or if its availability should not be checked)
*/
export function checkLicense(type, active, clusterSource, watcher = true) {
 // return object, set up with safe defaults
 const licenseInfo = {
   clusterAlerts: { enabled: false }
 };

 // Disabled because there is no license
 if (!type) {
   return Object.assign(licenseInfo, {
     message: `Cluster Alerts are not displayed because the [${clusterSource}] cluster's license could not be
determined.`
   });
 }

 // Disabled because the license type is not valid (basic)
 if (!includes([ 'trial', 'standard', 'gold', 'platinum' ], type)) {
   return  Object.assign(licenseInfo, {
     message: `Cluster Alerts are not displayed if Watcher is disabled or the [${clusterSource}] cluster's current
license is basic.`
   });
 }

 // Disabled because the license is inactive
 if (!active) {
   return Object.assign(licenseInfo, {
     message: `Cluster Alerts are not displayed because the [${clusterSource}] cluster's current license [${type}] is
not active.`
   });
 }

 // Disabled because Watcher is not enabled (it may or may not be available)
 if (!watcher) {
   return Object.assign(licenseInfo, { message: 'Cluster alerts is not enabled because Watcher is disabled.' });
 }
```

```
 return Object.assign(licenseInfo, { clusterAlerts: { enabled: true } });
}

/**
* Function to "generate" license check results for {@code xpackInfo}.
*
* @param {Object} xpackInfo license information for the _Monitoring_ cluster
* @param {Function} _checkLicense Method exposed for easier unit testing
* @returns {Object} Response from {@code checker}
*/
export function checkLicenseGenerator(xpackInfo, _checkLicense = checkLicense) {
 let type;
 let active = false;
 let watcher = false;

 if (xpackInfo && xpackInfo.license) {
   const watcherFeature = xpackInfo.feature('watcher');

   if (watcherFeature) {
     watcher = watcherFeature.isEnabled();
   }

   type = xpackInfo.license.getType();
   active = xpackInfo.license.isActive();
 }

 return _checkLicense(type, active, 'monitoring', watcher);
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { get } from 'lodash';
import { checkParam } from '../error_missing_required';
import { createQuery } from '../create_query';
import { ElasticsearchMetric } from '../metrics';

export function getClusterLicense(req, esIndexPattern, clusterUuid) {
 checkParam(esIndexPattern, 'esIndexPattern in getClusterLicense');

 const params = {
   index: esIndexPattern,
   size: 1,
   ignoreUnavailable: true,
   filterPath: 'hits.hits._source.license',
```

```
  body: {
    sort: { timestamp: { order: 'desc' } },
    query: createQuery({
      type: 'cluster_stats',
      clusterUuid,
      metric: ElasticsearchMetric.getMetricFields()
    })
  }
};

const { callWithRequest } = req.server.plugins.elasticsearch.getCluster('monitoring');
return callWithRequest(req, 'search', params)
  .then(response => {
    return get(response, 'hits.hits[0]._source.license', {});
  });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { contains } from 'lodash';
import { uiModules } from 'ui/modules';
import { ML_SUPPORTED_LICENSES } from '../../common/constants';

const uiModule = uiModules.get('monitoring/license', []);
uiModule.service('license', () => {
 return new class LicenseService {
  constructor() {
    // do not initialize with usable state
    this.license = {
      type: null,
      expiry_date_in_millis: -Infinity
    };
  }

  // we're required to call this initially
  setLicense(license) {
    this.license = license;
  }

  isBasic() {
    return this.license.type === 'basic';
  }

  mlIsSupported() {
    return contains(ML_SUPPORTED_LICENSES, this.license.type);
```

```
  }

  doesExpire() {
    const { expiry_date_in_millis: expiryDateInMillis } = this.license;
    return expiryDateInMillis !== undefined;
  }

  isActive() {
    const { expiry_date_in_millis: expiryDateInMillis } = this.license;
    return (new Date()).getTime() < expiryDateInMillis;
  }

  isExpired() {
    if (this.doesExpire()) {
      const { expiry_date_in_millis: expiryDateInMillis } = this.license;
      return (new Date()).getTime() >= expiryDateInMillis;
    }
    return false;
  }
 };
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React, { Fragment } from 'react';
import moment from 'moment-timezone';
import { capitalize } from 'lodash';
import { EuiLink } from '@elastic/eui';

const formatDateLocal = input => moment.tz(input, moment.tz.guess()).format('LL');

const WillExpireOn = ({ expiryDate }) => {
 if (expiryDate === undefined) {
   return null;
 }

 return <Fragment> will expire on {formatDateLocal(expiryDate)}</Fragment>;
};

export function LicenseText({ license, showLicenseExpiration }) {
 if (!showLicenseExpiration) {
   return null;
 }

 return (
```

```
  <EuiLink href="#/license">
    {capitalize(license.type)} license <WillExpireOn expiryDate={license.expiry_date_in_millis} />
  </EuiLink>
 );
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';
import {
 EuiPage,
 EuiPageBody,
 EuiSpacer,
 EuiCodeBlock,
 EuiPanel
} from '@elastic/eui';
import { LicenseStatus, AddLicense } from 'plugins/xpack_main/components';

const LicenseUpdateInfoForPrimary = ({ isPrimaryCluster, uploadLicensePath }) => {
 if (!isPrimaryCluster) {
  return null;
 }

 // viewed license is for the cluster directly connected to Kibana
 return <AddLicense uploadPath={uploadLicensePath} />;
};

const LicenseUpdateInfoForRemote = ({ isPrimaryCluster }) => {
 if (isPrimaryCluster) {
  return null;
 }

 // viewed license is for a remote monitored cluster not directly connected to Kibana
 return (
  <EuiPanel>
    <p>
      To update the license for this cluster, provide the license file through
      the Elasticsearch API:
    </p>
    <EuiSpacer />
    <EuiCodeBlock>
      {`curl -XPUT -u <user> 'https://<host>:<port>/_xpack/license' -H 'Content-Type: application/json' -d
@license.json`}
    </EuiCodeBlock>
  </EuiPanel>
```

```
 );
};

export function License(props) {
 const { status, type, isExpired, expiryDate } = props;
 return (
   <EuiPage className="licenseManagement">
     <EuiPageBody className="licManagement__pageBody">
       <div className="licManagement__contain">
         <LicenseStatus
           isExpired={isExpired}
           status={status}
           type={type}
           expiryDate={expiryDate}
         />

         <EuiSpacer size="l" />

         <LicenseUpdateInfoForPrimary {...props} />
         <LicenseUpdateInfoForRemote {...props} />
       </div>
     </EuiPageBody>
   </EuiPage>
 );
}
<monitoring-main data-test-subj="xpackLicensePage">
 <div id="licenseReact" class="licFeature"></div>
</monitoring-main>
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { get, find } from 'lodash';
import React from 'react';
import { render, unmountComponentAtNode } from 'react-dom';
import chrome from 'ui/chrome';
import { formatDateTimeLocal } from '../../../common/formatting';
import { MANAGEMENT_BASE_PATH } from 'plugins/xpack_main/components';
import { License } from 'plugins/monitoring/components';
import { timefilter } from 'ui/timefilter';

const REACT_NODE_ID = 'licenseReact';

export class LicenseViewController {
 constructor($injector, $scope) {
   timefilter.disableTimeRangeSelector();
```

```
  timefilter.disableAutoRefreshSelector();

  $scope.$on('$destroy', () => {
    unmountComponentAtNode(document.getElementById(REACT_NODE_ID));
  });

  this.init($injector, $scope);
}

init($injector, $scope) {
  const globalState = $injector.get('globalState');
  const title = $injector.get('title');
  const $route = $injector.get('$route');

  const cluster = find($route.current.locals.clusters, { cluster_uuid: globalState.cluster_uuid });
  $scope.cluster = cluster;
  title($scope.cluster, 'License');

  this.license = cluster.license;
  this.isExpired = Date.now() > get(cluster, 'license.expiry_date_in_millis');
  this.isPrimaryCluster = cluster.isPrimary;

  const basePath = chrome.getBasePath();
  this.uploadLicensePath = basePath + '/app/kibana#' + MANAGEMENT_BASE_PATH + 'upload_license';

  this.renderReact($scope);
}

renderReact($scope) {
  $scope.$evalAsync(() => {
    const { isPrimaryCluster, license, isExpired, uploadLicensePath } = this;
    let expiryDate = license.expiry_date_in_millis;
    if (license.expiry_date_in_millis !== undefined) {
      expiryDate = formatDateTimeLocal(license.expiry_date_in_millis);
    }

    // Mount the React component to the template
    render(
      <License
        isPrimaryCluster={isPrimaryCluster}
        status={license.status}
        type={license.type}
        isExpired={isExpired}
        expiryDate={expiryDate}
        uploadLicensePath={uploadLicensePath}
      />,
      document.getElementById(REACT_NODE_ID)
    );
```

```
  });
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import uiRoutes from 'ui/routes';
import { routeInitProvider } from 'plugins/monitoring/lib/route_init';
import template from './index.html';
import { LicenseViewController } from './controller';

uiRoutes.when('/license', {
 template,
 resolve: {
   clusters: (Private) => {
     const routeInit = Private(routeInitProvider);
     return routeInit();
   }
 },
 controllerAs: 'licenseView',
 controller: LicenseViewController
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/



import expect from 'expect.js';
import sinon from 'sinon';
import { set } from 'lodash';
import { checkLicense, isBasicLicense } from '../check_license';

describe('check_license', () => {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is undefined', () => {
   beforeEach(() => mockLicenseInfo = undefined);

   it('should set isAvailable to false', () => {
     expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
```

```
  });

  it('should set showLinks to true', () => {
    expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
  });

  it('should set enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('license information is not available', () => {
  beforeEach(() => mockLicenseInfo.isAvailable = () => false);

  it('should set isAvailable to false', () => {
    expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
  });

  it('should set showLinks to true', () => {
    expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
  });

  it('should set enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('license information is available', () => {
  beforeEach(() => {
    mockLicenseInfo.isAvailable = () => true;
    set(mockLicenseInfo, 'license.getType', () => 'basic');
  });

  describe('& ML is disabled in Elasticsearch', () => {
    beforeEach(() => {
      set(mockLicenseInfo, 'feature', sinon.stub().withArgs('ml').returns({ isEnabled: () => false }));
    });

    it ('should set showLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(false);
```

```
  });

  it ('should set isAvailable to false', () => {
    expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
  });

  it('should set enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('& ML is enabled in Elasticsearch', () => {
  beforeEach(() => {
    set(mockLicenseInfo, 'feature', sinon.stub().withArgs('ml').returns({ isEnabled: () => true }));
  });

  describe('& license is trial or platinum', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

    describe('& license is active', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

      it('should set isAvailable to true', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
      });

      it ('should set showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
      });

      it ('should set enableLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
      });

      it('should not set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
      });
    });

    describe('& license is expired', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

      it('should set isAvailable to true', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
```

```
        });

        it ('should set showLinks to true', () => {
          expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
        });

        it ('should set enableLinks to true', () => {
          expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
        });

        it('should set a message', () => {
          expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
        });
      });
    });

    describe('& license is basic', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => false));

      describe('& license is active', () => {
        beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

        it('should set isAvailable to true', () => {
          expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
        });

        it ('should set showLinks to true', () => {
          expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
        });
      });
    });
  });
});
});

function mockServerFactory(licenseType = 'basic') {
 return {
   plugins: {
     xpack_main: {
       info: {
         license: {
           getType: () => licenseType
         }
       }
     }
   }
 };
}
```

```
describe('isBasicLicense', () => {
 describe('license information is basic', () => {
   it('should be true', () => {
     expect(isBasicLicense(mockServerFactory())).to.be(true);
   });
 });

 describe('license information is gold', () => {
   it('should be true', () => {
     expect(isBasicLicense(mockServerFactory('gold'))).to.be(false);
   });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/


import { LICENSE_TYPE } from '../../../common/constants/license';

export function checkLicense(xpackLicenseInfo) {
 // If, for some reason, we cannot get the license information
 // from Elasticsearch, assume worst case and disable the Machine Learning UI
 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     isAvailable: false,
     showLinks: true,
     enableLinks: false,
     message: 'You cannot use Machine Learning because license information is not available at this time.'
   };
 }

 const featureEnabled = xpackLicenseInfo.feature('ml').isEnabled();
 if (!featureEnabled) {
   return {
     isAvailable: false,
     showLinks: false,
     enableLinks: false,
     message: 'Machine Learning is unavailable'
   };
 }

 const VALID_FULL_LICENSE_MODES = [
   'trial',
   'platinum'
```

```
 ];

 const isLicenseModeValid = xpackLicenseInfo.license.isOneOf(VALID_FULL_LICENSE_MODES);
 const licenseType = (isLicenseModeValid === true) ? LICENSE_TYPE.FULL : LICENSE_TYPE.BASIC;
 const isLicenseActive = xpackLicenseInfo.license.isActive();
 const licenseTypeName = xpackLicenseInfo.license.getType();

 // Platinum or trial license is valid but not active, i.e. expired
 if (licenseType === LICENSE_TYPE.FULL && isLicenseActive === false) {
   return {
     isAvailable: true,
     showLinks: true,
     enableLinks: true,
     hasExpired: true,
     licenseType,
     message: `Your ${licenseTypeName} Machine Learning license has expired.`
   };
 }

 // License is valid and active
 return {
   isAvailable: true,
   showLinks: true,
   enableLinks: true,
   licenseType,
   hasExpired: false,
 };
}

export function isBasicLicense(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const xpackInfo = (xpackMainPlugin && xpackMainPlugin.info);
 return (xpackInfo.license.getType() === 'basic');
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/



export { checkLicense, isBasicLicense } from './check_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
export const LICENSE_TYPE = {
 BASIC: 0,
 FULL: 1,
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/



import expect from 'expect.js';
import {
 xpackFeatureProvider,
} from '../check_license';

function Private() {
 return {
   get(str) {
     if (str === 'features.watcher.isAvailable') {
       return true;
     } else {
       return false;
     }
   }
 };
}

describe('ML - check license', () => {

 describe('xpackFeatureProvider', () => {
   it('returns true for enabled feature', () => {
     const xpackFeature = xpackFeatureProvider(Private);
     const result = xpackFeature.isAvailable('watcher');
     expect(result).to.be(true);
   });

   it('returns false for disabled feature', () => {
     const xpackFeature = xpackFeatureProvider(Private);
     const result = xpackFeature.isAvailable('noSuchFeature');
     expect(result).to.be(false);
   });
 });
});
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/


import React from 'react';
import { XPackInfoProvider } from 'plugins/xpack_main/services/xpack_info';
import { banners, addAppRedirectMessageToUrl } from 'ui/notify';
import { LICENSE_TYPE } from '../../common/constants/license';

import chrome from 'ui/chrome';
import { EuiCallOut } from '@elastic/eui';

let licenseHasExpired = true;
let licenseType = null;
let expiredLicenseBannerId;

export function checkFullLicense(Private, kbnBaseUrl, kbnUrl) {
 const features = getFeatures(Private);
 licenseType = features.licenseType;

 if (features.isAvailable === false) {
   // ML is not enabled
   return redirectToKibana(features, kbnBaseUrl);

 } else if (features.licenseType === LICENSE_TYPE.BASIC) {

   // ML is enabled, but only with a basic or gold license
   return redirectToBasic(kbnUrl);

 } else {

   // ML is enabled
   licenseHasExpired = (features.hasExpired || false);
   // If the license has expired ML app will still work for 7 days and then
   // the job management endpoints (e.g. create job, start datafeed) will be restricted.
   // Therefore we need to keep the app enabled but show an info banner to the user.
   if(licenseHasExpired) {
     const message = features.message;
     if (expiredLicenseBannerId === undefined) {
       // Only show the banner once with no way to dismiss it
       expiredLicenseBannerId = banners.add({
         component: (
           <EuiCallOut
             iconType="iInCircle"
             color="warning"
```

```
            title={message}
          />
        ),
      });
    }
  }
  return Promise.resolve(features);
 }
}

export function checkBasicLicense(Private, kbnBaseUrl) {
 const features = getFeatures(Private);
 licenseType = features.licenseType;

 if (features.isAvailable === false) {
   // ML is not enabled
   return redirectToKibana(features, kbnBaseUrl);

 } else {

   // ML is enabled
   return Promise.resolve(features);
 }
}

// a wrapper for checkFullLicense which doesn't resolve if the license has expired.
// this is used by all create jobs pages to redirect back to the jobs list
// if the user's license has expired.
export function checkLicenseExpired(Private, kbnBaseUrl, kbnUrl) {
 return checkFullLicense(Private, kbnBaseUrl, kbnUrl)
   .then((features) => {
     if (features.hasExpired) {
       kbnUrl.redirect('/jobs');
       return Promise.halt();
     } else {
       return Promise.resolve(features);
     }
   })
   .catch(() => {
     return Promise.halt();
   });
}

function getFeatures(Private) {
 const xpackInfo = Private(XPackInfoProvider);
 return xpackInfo.get('features.ml');
}
```

```
function redirectToKibana(features, kbnBaseUrl) {
 const { message } = features;
 const newUrl = addAppRedirectMessageToUrl(chrome.addBasePath(kbnBaseUrl), (message || ''));
 window.location.href = newUrl;
 return Promise.halt();
}

function redirectToBasic(kbnUrl) {
 kbnUrl.redirect('/datavisualizer');
 return Promise.halt();
}

export function hasLicenseExpired() {
 return licenseHasExpired;
}

export function isFullLicense() {
 return (licenseType === LICENSE_TYPE.FULL);
}

export function xpackFeatureProvider(Private) {
 const xpackInfo = Private(XPackInfoProvider);

 return {
   isAvailable(feature) {
     return xpackInfo.get(`features.${feature}.isAvailable`, false);
   }
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

const Boom = require('boom');

export function routePreCheckLicense(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const pluginId = 'security';
 return function forbidApiAccess(request, reply) {
   const licenseCheckResults = xpackMainPlugin.info.feature(pluginId).getLicenseCheckResults();
   if (!licenseCheckResults.showLinks) {
     reply(Boom.forbidden(licenseCheckResults.linksMessage));
   } else {
     reply();
   }
 };
```

```
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import sinon from 'sinon';
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockXPackInfo;

 beforeEach(function () {
   mockXPackInfo = {
     isAvailable: sinon.stub(),
     isXpackUnavailable: sinon.stub(),
     feature: sinon.stub(),
     license: sinon.stub({
       isOneOf() { },
     })
   };

   mockXPackInfo.isAvailable.returns(true);
 });

 it('should display error when ES is unavailable', () => {
   mockXPackInfo.isAvailable.returns(false);
   mockXPackInfo.isXpackUnavailable.returns(false);

   const licenseCheckResults = checkLicense(mockXPackInfo);
   expect(licenseCheckResults).to.be.eql({
     showLogin: true,
     allowLogin: false,
     showLinks: false,
     allowRoleDocumentLevelSecurity: false,
     allowRoleFieldLevelSecurity: false,
     layout: 'error-es-unavailable',
     allowRbac: false,
   });
 });

 it('should display error when X-Pack is unavailable', () => {
   mockXPackInfo.isAvailable.returns(false);
   mockXPackInfo.isXpackUnavailable.returns(true);
```

```
  const licenseCheckResults = checkLicense(mockXPackInfo);
  expect(licenseCheckResults).to.be.eql({
    showLogin: true,
    allowLogin: false,
    showLinks: false,
    allowRoleDocumentLevelSecurity: false,
    allowRoleFieldLevelSecurity: false,
    layout: 'error-xpack-unavailable',
    allowRbac: false,
  });
});


it('should not show login page or other security elements if license is basic.', () => {
  mockXPackInfo.license.isOneOf.withArgs(['basic']).returns(true);
  mockXPackInfo.feature.withArgs('security').returns({
    isEnabled: () => { return true; }
  });

  const licenseCheckResults = checkLicense(mockXPackInfo);
  expect(licenseCheckResults).to.be.eql({
    showLogin: false,
    allowLogin: false,
    showLinks: false,
    allowRoleDocumentLevelSecurity: false,
    allowRoleFieldLevelSecurity: false,
    allowRbac: false,
    linksMessage: 'Your Basic license does not support Security. Please upgrade your license.'
  });
});

it('should not show login page or other security elements if security is disabled in Elasticsearch.', () => {
  mockXPackInfo.license.isOneOf.withArgs(['basic']).returns(false);
  mockXPackInfo.feature.withArgs('security').returns({
    isEnabled: () => { return false; }
  });

  const licenseCheckResults = checkLicense(mockXPackInfo);
  expect(licenseCheckResults).to.be.eql({
    showLogin: false,
    allowLogin: false,
    showLinks: false,
    allowRoleDocumentLevelSecurity: false,
    allowRoleFieldLevelSecurity: false,
    allowRbac: false,
    linksMessage: 'Access is denied because Security is disabled in Elasticsearch.'
  });
});
```

```
 it('should allow to login and allow RBAC but forbid document level security if license is not platinum or trial.', ()
=> {
   mockXPackInfo.license.isOneOf
     .returns(false)
     .withArgs(['platinum', 'trial']).returns(false);
   mockXPackInfo.feature.withArgs('security').returns({
     isEnabled: () => { return true; }
   });

   expect(checkLicense(mockXPackInfo)).to.be.eql({
     showLogin: true,
     allowLogin: true,
     showLinks: true,
     allowRoleDocumentLevelSecurity: false,
     allowRoleFieldLevelSecurity: false,
     allowRbac: true,
   });
 });

 it('should allow to login, allow RBAC and document level security if license is platinum or trial.', () => {
   mockXPackInfo.license.isOneOf
     .returns(false)
     .withArgs(['platinum', 'trial']).returns(true);
   mockXPackInfo.feature.withArgs('security').returns({
     isEnabled: () => { return true; }
   });

   expect(checkLicense(mockXPackInfo)).to.be.eql({
     showLogin: true,
     allowLogin: true,
     showLinks: true,
     allowRoleDocumentLevelSecurity: true,
     allowRoleFieldLevelSecurity: true,
     allowRbac: true,
   });
 });

});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

/**
* @typedef {Object} LicenseCheckResult Result of the license check.
* @property {boolean} showLogin Indicates whether we show login page or skip it.
```

```
* @property {boolean} allowLogin Indicates whether we allow login or disable it on the login page.
* @property {boolean} showLinks Indicates whether we show security links throughout the kibana app.
* @property {boolean} allowRoleDocumentLevelSecurity Indicates whether we allow users to define document
level
* security in roles.
* @property {boolean} allowRoleFieldLevelSecurity Indicates whether we allow users to define field level security
* in roles
* @property {string} [linksMessage] Message to show when security links are clicked throughout the kibana app.
*/

/**
* Returns object that defines behavior of the security related areas (login page, user management etc.) based
* on the license information extracted from the xPackInfo.
* @param {XPackInfo} xPackInfo XPackInfo instance to extract license information from.
* @returns {LicenseCheckResult}
*/
export function checkLicense(xPackInfo) {
 // If, for some reason, we cannot get license information from Elasticsearch,
 // assume worst-case and lock user at login screen.
 if (!xPackInfo.isAvailable()) {
   return {
     showLogin: true,
     allowLogin: false,
     showLinks: false,
     allowRoleDocumentLevelSecurity: false,
     allowRoleFieldLevelSecurity: false,
     allowRbac: false,
     layout: xPackInfo.isXpackUnavailable() ? 'error-xpack-unavailable' : 'error-es-unavailable'
   };
 }

 const isLicenseBasic = xPackInfo.license.isOneOf(['basic']);
 const isEnabledInES = xPackInfo.feature('security').isEnabled();
 if (!isEnabledInES || isLicenseBasic) {
   return {
     showLogin: false,
     allowLogin: false,
     showLinks: false,
     allowRoleDocumentLevelSecurity: false,
     allowRoleFieldLevelSecurity: false,
     allowRbac: false,
     linksMessage: isLicenseBasic
       ? 'Your Basic license does not support Security. Please upgrade your license.'
       : 'Access is denied because Security is disabled in Elasticsearch.'
   };

 }
```

```
 const isLicensePlatinumOrTrial = xPackInfo.license.isOneOf(['platinum', 'trial']);
 return {
   showLogin: true,
   allowLogin: true,
   showLinks: true,
   // Only platinum and trial licenses are compliant with field- and document-level security.
   allowRoleDocumentLevelSecurity: isLicensePlatinumOrTrial,
   allowRoleFieldLevelSecurity: isLicensePlatinumOrTrial,
   allowRbac: true,
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { XPackInfoProvider } from 'plugins/xpack_main/services/xpack_info';
import { Notifier } from 'ui/notify';

export function checkLicenseError(kbnUrl, Promise, Private) {
 const xpackInfo = Private(XPackInfoProvider);
 const genericNotifier = new Notifier({ location: 'Security' });

 return err => {
   if (!xpackInfo.get('features.security.showLinks')) {
     genericNotifier.error(xpackInfo.get('features.security.linksMessage'));
     kbnUrl.redirect('/management');
     return Promise.halt();
   }
   return Promise.reject(err);
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { mirrorPluginStatus } from '../../../../../server/lib/mirror_plugin_status';
import { checkLicense } from '../check_license';
import { PLUGIN } from '../../../common/constants';

export function registerLicenseChecker(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const grokdebuggerPlugin = server.plugins[PLUGIN.ID];

 mirrorPluginStatus(xpackMainPlugin, grokdebuggerPlugin);
```

```
 xpackMainPlugin.status.once('green', () => {
   // Register a function that is called whenever the xpack info changes,
   // to re-compute the license check results for this plugin
   xpackMainPlugin.info.feature(PLUGIN.ID).registerLicenseCheckResultsGenerator(checkLicense);
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import Boom from 'boom';
import { PLUGIN } from '../../../common/constants';

export const licensePreRoutingFactory = (server) => {
 const xpackMainPlugin = server.plugins.xpack_main;

 // License checking and enable/disable logic
 function licensePreRouting(request, reply) {
   const licenseCheckResults = xpackMainPlugin.info.feature(PLUGIN.ID).getLicenseCheckResults();
   if (!licenseCheckResults.enableAPIRoute) {
     reply(Boom.forbidden(licenseCheckResults.message));
   } else {
     reply();
   }
 }

 return licensePreRouting;
};
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import Boom from 'boom';
import { licensePreRoutingFactory } from '../license_pre_routing_factory';

describe('license_pre_routing_factory', () => {
 describe('#grokDebuggerFeaturePreRoutingFactory', () => {
   let mockServer;
   let mockLicenseCheckResults;

   beforeEach(() => {
     mockServer = {
       plugins: {
```

```
        xpack_main: {
          info: {
            feature: () => ({
              getLicenseCheckResults: () => mockLicenseCheckResults
            })
          }
        }
      }
    };
  });

  describe('isAvailable is false', () => {
    beforeEach(() => {
      mockLicenseCheckResults = {
        isAvailable: false
      };
    });

    it ('replies with 403', (done) => {
      const licensePreRouting = licensePreRoutingFactory(mockServer);
      const stubRequest = {};
      licensePreRouting(stubRequest, (response) => {
        expect(response).to.be.an(Error);
        expect(response.isBoom).to.be(true);
        expect(response.output.statusCode).to.be(403);
        done();
      });
    });
  });

  describe('isAvailable is true', () => {
    beforeEach(() => {
      mockLicenseCheckResults = {
        isAvailable: true
      };
    });

    it ('replies with nothing', (done) => {
      const licensePreRouting = licensePreRoutingFactory(mockServer);
      const stubRequest = {};
      licensePreRouting(stubRequest, (response) => {
        expect(response).to.eql(Boom.forbidden());
        done();
      });
    });
  });
 });
});
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is undefined', () => {
   beforeEach(() => mockLicenseInfo = undefined);

   it('should set enableLink to false', () => {
     expect(checkLicense(mockLicenseInfo).enableLink).to.be(false);
   });

   it('should set enableAPIRoute to false', () => {
     expect(checkLicense(mockLicenseInfo).enableAPIRoute).to.be(false);
   });

   it('should set a message', () => {
     expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
   });
 });

 describe('license information is not available', () => {
   beforeEach(() => mockLicenseInfo.isAvailable = () => false);

   it('should set enableLink to false', () => {
     expect(checkLicense(mockLicenseInfo).enableLink).to.be(false);
   });

   it('should set enableAPIRoute to false', () => {
     expect(checkLicense(mockLicenseInfo).enableAPIRoute).to.be(false);
   });

   it('should set a message', () => {
     expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
   });
 });

 describe('license information is available', () => {
```

```
  beforeEach(() => mockLicenseInfo = {
    isAvailable: () => true,
    license: {
      getType: () => 'foobar'
    }
  });

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it ('should set enableLink to true', () => {
      expect(checkLicense(mockLicenseInfo).enableLink).to.be(true);
    });

    it('should set enableAPIRoute to true', () => {
      expect(checkLicense(mockLicenseInfo).enableAPIRoute).to.be(true);
    });

    it('should NOT set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it ('should set enableLink to false', () => {
      expect(checkLicense(mockLicenseInfo).enableLink).to.be(false);
    });

    it('should set enableAPIRoute to false', () => {
      expect(checkLicense(mockLicenseInfo).enableAPIRoute).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export function checkLicense(xpackLicenseInfo) {
 // If, for some reason, we cannot get the license information
```

```
 // from Elasticsearch, assume worst case and disable the Watcher UI
 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     enableLink: false,
     enableAPIRoute: false,
     message: 'You cannot use the Grok Debugger because license information is not available at this time.'
   };
 }

 const isLicenseActive = xpackLicenseInfo.license.isActive();
 const licenseType = xpackLicenseInfo.license.getType();

 // License is not valid
 if (!isLicenseActive) {
   return {
     enableLink: false,
     enableAPIRoute: false,
     message: `You cannot use the Grok Debugger because your ${licenseType} license has expired.`
   };
 }

 // License is valid and active
 return {
   enableLink: true,
   enableAPIRoute: true
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { mirrorPluginStatus } from '../../../../../server/lib/mirror_plugin_status';
import { checkLicense } from '../check_license';
import { PLUGIN } from '../../../common/constants';

export function registerLicenseChecker(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const indexManagementPlugin = server.plugins.index_management;

 mirrorPluginStatus(xpackMainPlugin, indexManagementPlugin);
 xpackMainPlugin.status.once('green', () => {
   // Register a function that is called whenever the xpack info changes,
   // to re-compute the license check results for this plugin
   xpackMainPlugin.info.feature(PLUGIN.ID).registerLicenseCheckResultsGenerator(checkLicense);
 });
}
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is undefined', () => {
   beforeEach(() => mockLicenseInfo = undefined);

   it('should set isAvailable to false', () => {
     expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
   });

   it('should set showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
   });

   it('should set enableLinks to false', () => {
     expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
   });

   it('should set a message', () => {
     expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
   });
 });

 describe('license information is not available', () => {
   beforeEach(() => mockLicenseInfo.isAvailable = () => false);

   it('should set isAvailable to false', () => {
     expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
   });

   it('should set showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
   });

   it('should set enableLinks to false', () => {
     expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
```

```
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('license information is available', () => {
  beforeEach(() => {
    mockLicenseInfo.isAvailable = () => true;
    set(mockLicenseInfo, 'license.getType', () => 'basic');
  });

  describe('& license is trial, standard, gold, platinum', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

    describe('& license is active', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

      it('should set isAvailable to true', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
      });

      it ('should set showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
      });

      it ('should set enableLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
      });

      it('should not set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
      });
    });

    describe('& license is expired', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

      it('should set isAvailable to false', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
      });

      it ('should set showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
      });

      it ('should set enableLinks to false', () => {
```

```
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
});

describe('& license is basic', () => {
  beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it('should set isAvailable to true', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
    });

    it ('should set showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
    });

    it ('should set enableLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
    });

    it('should not set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it('should set isAvailable to false', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
    });

    it ('should set showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
});
```

```
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export function checkLicense(xpackLicenseInfo) {
 const pluginName = 'Index Management';

 // If, for some reason, we cannot get the license information
 // from Elasticsearch, assume worst case and disable
 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     isAvailable: false,
     showLinks: true,
     enableLinks: false,
     message: `You cannot use ${pluginName} because license information is not available at this time.`
   };
 }

 const VALID_LICENSE_MODES = [
   'trial',
   'basic',
   'standard',
   'gold',
   'platinum'
 ];

 const isLicenseModeValid = xpackLicenseInfo.license.isOneOf(VALID_LICENSE_MODES);
 const isLicenseActive = xpackLicenseInfo.license.isActive();
 const licenseType = xpackLicenseInfo.license.getType();

 // License is not valid
 if (!isLicenseModeValid) {
   return {
     isAvailable: false,
     showLinks: false,
     message: `Your ${licenseType} license does not support ${pluginName}. Please upgrade your license.`
   };
 }

 // License is valid but not active
 if (!isLicenseActive) {
   return {
     isAvailable: false,
     showLinks: true,
```

```
    enableLinks: false,
    message: `You cannot use ${pluginName} because your ${licenseType} license has expired.`
  };
}

// License is valid and active
return {
  isAvailable: true,
  showLinks: true,
  enableLinks: true
};
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { mirrorPluginStatus } from '../../../../../server/lib/mirror_plugin_status';
import { checkLicense } from '../check_license';
import { PLUGIN } from '../../../common/constants';

export function registerLicenseChecker(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const logstashPlugin = server.plugins.logstash;

 mirrorPluginStatus(xpackMainPlugin, logstashPlugin);
 xpackMainPlugin.status.once('green', () => {
   // Register a function that is called whenever the xpack info changes,
   // to re-compute the license check results for this plugin
   xpackMainPlugin.info.feature(PLUGIN.ID).registerLicenseCheckResultsGenerator(checkLicense);
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { licensePreRoutingFactory } from '../license_pre_routing_factory';

describe('license_pre_routing_factory', () => {
 describe('#logstashFeaturePreRoutingFactory', () => {
   let mockServer;
   let mockLicenseCheckResults;

   beforeEach(() => {
```

```
  mockServer = {
    plugins: {
      xpack_main: {
        info: {
          feature: () => ({
            getLicenseCheckResults: () => mockLicenseCheckResults
          })
        }
      }
    }
  };
});

it('only instantiates one instance per server', () => {
  const firstInstance = licensePreRoutingFactory(mockServer);
  const secondInstance = licensePreRoutingFactory(mockServer);

  expect(firstInstance).to.be(secondInstance);
});

describe('isAvailable is false', () => {
  beforeEach(() => {
    mockLicenseCheckResults = {
      isAvailable: false
    };
  });

  it ('replies with 403', (done) => {
    const licensePreRouting = licensePreRoutingFactory(mockServer);
    const stubRequest = {};
    licensePreRouting(stubRequest, (response) => {
      expect(response).to.be.an(Error);
      expect(response.isBoom).to.be(true);
      expect(response.output.statusCode).to.be(403);
      done();
    });
  });
});

describe('isAvailable is true', () => {
  beforeEach(() => {
    mockLicenseCheckResults = {
      isAvailable: true
    };
  });

  it ('replies with nothing', (done) => {
    const licensePreRouting = licensePreRoutingFactory(mockServer);
```

```
      const stubRequest = {};
      licensePreRouting(stubRequest, (response) => {
        expect(response).to.be(undefined);
        done();
      });
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is undefined', () => {
  beforeEach(() => mockLicenseInfo = undefined);

  it('should set isAvailable to false', () => {
    expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
  });

  it('should set enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set isReadOnly to false', () => {
    expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
 });

 describe('license information is not available', () => {
  beforeEach(() => mockLicenseInfo.isAvailable = () => false);

  it('should set isAvailable to false', () => {
```

```
    expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
  });

  it('should set enableLinks to false', () => {
    expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
  });

  it('should set isReadOnly to false', () => {
    expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(false);
  });

  it('should set a message', () => {
    expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
  });
});

describe('license information is available', () => {
  beforeEach(() => {
    mockLicenseInfo.isAvailable = () => true;
    set(mockLicenseInfo, 'license.getType', () => 'basic');
  });

  describe('& license is trial, standard, gold, platinum', () => {
    beforeEach(() => {
      set(mockLicenseInfo, 'license.isOneOf', () => true);
      mockLicenseInfo.feature = () => ({ isEnabled: () => true }); // Security feature is enabled
    });

    describe('& license is active', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

      it('should set isAvailable to true', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
      });

      it ('should set enableLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
      });

      it ('should set isReadOnly to false', () => {
        expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(false);
      });

      it('should not set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
      });
    });
```

```
  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it('should set isAvailable to true', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
    });

    it ('should set enableLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
    });

    it ('should set isReadOnly to true', () => {
      expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(true);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
});

describe('& license is basic', () => {
  beforeEach(() => {
    set(mockLicenseInfo, 'license.isOneOf', () => false);
    mockLicenseInfo.feature = () => ({ isEnabled: () => true }); // Security feature is enabled
  });

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it('should set isAvailable to false', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
    });

    it ('should set enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
    });

    it ('should set isReadOnly to false', () => {
      expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });

  describe('& license is expired', () => {
```

```
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

      it('should set isAvailable to false', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
      });

      it ('should set enableLinks to false', () => {
        expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
      });

      it ('should set isReadOnly to false', () => {
        expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(false);
      });

      it('should set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
      });
    });
  });

  describe('& security is disabled', () => {
    beforeEach(() => {
      mockLicenseInfo.feature = () => ({ isEnabled: () => false }); // Security feature is disabled
      set(mockLicenseInfo, 'license.isOneOf', () => true);
      set(mockLicenseInfo, 'license.isActive', () => true);
    });

    it('should set isAvailable to false', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
    });

    it ('should set enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
    });

    it ('should set isReadOnly to false', () => {
      expect(checkLicense(mockLicenseInfo).isReadOnly).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
```

```
* you may not use this file except in compliance with the Elastic License.
*/

export function checkLicense(xpackLicenseInfo) {
 // If, for some reason, we cannot get the license information
 // from Elasticsearch, assume worst case and disable the Logstash pipeline UI
 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     isAvailable: false,
     enableLinks: false,
     isReadOnly: false,
     message: 'You cannot manage Logstash pipelines because license information is not available at this time.'
   };
 }

 const VALID_LICENSE_MODES = [
   'trial',
   'standard',
   'gold',
   'platinum'
 ];

 const isLicenseModeValid = xpackLicenseInfo.license.isOneOf(VALID_LICENSE_MODES);
 const isLicenseActive = xpackLicenseInfo.license.isActive();
 const licenseType = xpackLicenseInfo.license.getType();
 const isSecurityEnabled = xpackLicenseInfo.feature('security').isEnabled();

 // Security is not enabled in ES
 if (!isSecurityEnabled) {
   const message = 'Security must be enabled in order to use Logstash pipeline management features.'
     + ' Please set xpack.security.enabled: true in your elasticsearch.yml.';
   return {
     isAvailable: false,
     enableLinks: false,
     isReadOnly: false,
     message
   };
 }

 // License is not valid
 if (!isLicenseModeValid) {
   return {
     isAvailable: false,
     enableLinks: false,
     isReadOnly: false,
     message: `Your ${licenseType} license does not support Logstash pipeline management features. Please upgrade
your license.`
   };
```

```
 }

 // License is valid but not active, we go into a read-only mode.
 if (!isLicenseActive) {
   return {
     isAvailable: true,
     enableLinks: true,
     isReadOnly: true,
     message: `You cannot edit, create, or delete your Logstash pipelines because your ${licenseType} license has
expired.`
   };
 }

 // License is valid and active
 return {
   isAvailable: true,
   enableLinks: true,
   isReadOnly: false
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { uiModules } from 'ui/modules';
import { XPackInfoProvider } from 'plugins/xpack_main/services/xpack_info';
import 'ui/url';
import { LogstashLicenseService } from './logstash_license_service';

uiModules.get('xpack/logstash')
 .factory('logstashLicenseService', ($injector) => {
   const Private = $injector.get('Private');
   const xpackInfoService = Private(XPackInfoProvider);
   const kbnUrlService = $injector.get('kbnUrl');
   const $timeout = $injector.get('$timeout');

   return new LogstashLicenseService(xpackInfoService, kbnUrlService, $timeout);
 });
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';
import { toastNotifications } from 'ui/notify';
```

```
import { MarkdownSimple } from 'ui/markdown';
import { PLUGIN } from '../../../common/constants';

export class LogstashLicenseService {
 constructor(xpackInfoService, kbnUrlService, $timeout) {
   this.xpackInfoService = xpackInfoService;
   this.kbnUrlService = kbnUrlService;
   this.$timeout = $timeout;
 }

 get enableLinks() {
   return Boolean(this.xpackInfoService.get(`features.${PLUGIN.ID}.enableLinks`));
 }

 get isAvailable() {
   return Boolean(this.xpackInfoService.get(`features.${PLUGIN.ID}.isAvailable`));
 }

 get isReadOnly() {
   return Boolean(this.xpackInfoService.get(`features.${PLUGIN.ID}.isReadOnly`));
 }

 get message() {
   return this.xpackInfoService.get(`features.${PLUGIN.ID}.message`);
 }

 notifyAndRedirect() {
   toastNotifications.addDanger({
     title: <MarkdownSimple>{this.xpackInfoService.get(`features.${PLUGIN.ID}.message`)}</MarkdownSimple>,
   });
   this.kbnUrlService.redirect('/management');
 }

 /**
  * Checks if the license is valid or the license can perform downgraded UI tasks.
  * Otherwise, notifies and redirects.
  */
 checkValidity() {
   return new Promise((resolve, reject) => {
     this.$timeout(() => {
       if (this.isAvailable) {
         return resolve();
       }

       this.notifyAndRedirect();
       return reject();
     }, 10); // To allow latest XHR call to update license info
   });
```

```
 }
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
import React from 'react';
import { createInitialDataSelector } from './helpers';
import { Request } from 'react-redux-request';
import { loadLicense } from '../../services/rest/apm';

const ID = 'license';
const INITIAL_DATA = {
 features: {
   watcher: { isAvailable: false },
   ml: { isAvailable: false }
 },
 license: { isActive: false }
};

const withInitialData = createInitialDataSelector(INITIAL_DATA);

export function getLicense(state) {
 return withInitialData(state.reactReduxRequest[ID]);
}

export function LicenceRequest({ render }) {
 return (
   <Request id={ID} fn={loadLicense} selector={getLicense} render={render} />
 );
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { mirrorPluginStatus } from '../../../../../server/lib/mirror_plugin_status';
import { checkLicense } from '../check_license';
import { PLUGIN } from '../../../common';

export function registerLicenseChecker(server) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const rollupPlugin = server.plugins[PLUGIN.ID];

 mirrorPluginStatus(xpackMainPlugin, rollupPlugin);
 xpackMainPlugin.status.once('green', () => {
```

```
  // Register a function that is called whenever the xpack info changes,
  // to re-compute the license check results for this plugin
  xpackMainPlugin.info.feature(PLUGIN.ID).registerLicenseCheckResultsGenerator(checkLicense);
 });
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { registerLicenseChecker } from './register_license_checker';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import { once } from 'lodash';
import { wrapCustomError } from '../error_wrappers';
import { PLUGIN } from '../../../common';

export const licensePreRoutingFactory = once((server) => {
 const xpackMainPlugin = server.plugins.xpack_main;

 // License checking and enable/disable logic
 function licensePreRouting(request, reply) {
  const licenseCheckResults = xpackMainPlugin.info.feature(PLUGIN.ID).getLicenseCheckResults();
  if (!licenseCheckResults.isAvailable) {
   const error = new Error(licenseCheckResults.message);
   const statusCode = 403;
   const wrappedError = wrapCustomError(error, statusCode);
   reply(wrappedError);
  } else {
   reply();
  }
 }

 return licensePreRouting;
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
```

```
import { checkLicense } from '../check_license';

describe('check_license', function () {

 let mockLicenseInfo;
 beforeEach(() => mockLicenseInfo = {});

 describe('license information is undefined', () => {
   beforeEach(() => mockLicenseInfo = undefined);

   it('should set isAvailable to false', () => {
     expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
   });

   it('should set showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
   });

   it('should set enableLinks to false', () => {
     expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
   });

   it('should set a message', () => {
     expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
   });
 });

 describe('license information is not available', () => {
   beforeEach(() => mockLicenseInfo.isAvailable = () => false);

   it('should set isAvailable to false', () => {
     expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
   });

   it('should set showLinks to true', () => {
     expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
   });

   it('should set enableLinks to false', () => {
     expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
   });

   it('should set a message', () => {
     expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
   });
 });

 describe('license information is available', () => {
```

```
beforeEach(() => {
  mockLicenseInfo.isAvailable = () => true;
  set(mockLicenseInfo, 'license.getType', () => 'basic');
});

describe('& license is trial, standard, gold, platinum', () => {
  beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

  describe('& license is active', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

    it('should set isAvailable to true', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
    });

    it ('should set showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
    });

    it ('should set enableLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
    });

    it('should not set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
    });
  });

  describe('& license is expired', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

    it('should set isAvailable to false', () => {
      expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
    });

    it ('should set showLinks to true', () => {
      expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
    });

    it ('should set enableLinks to false', () => {
      expect(checkLicense(mockLicenseInfo).enableLinks).to.be(false);
    });

    it('should set a message', () => {
      expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
    });
  });
});
```

```
  describe('& license is basic', () => {
    beforeEach(() => set(mockLicenseInfo, 'license.isOneOf', () => true));

    describe('& license is active', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => true));

      it('should set isAvailable to true', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(true);
      });

      it ('should set showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
      });

      it ('should set enableLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).enableLinks).to.be(true);
      });

      it('should not set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.be(undefined);
      });
    });

    describe('& license is expired', () => {
      beforeEach(() => set(mockLicenseInfo, 'license.isActive', () => false));

      it('should set isAvailable to false', () => {
        expect(checkLicense(mockLicenseInfo).isAvailable).to.be(false);
      });

      it ('should set showLinks to true', () => {
        expect(checkLicense(mockLicenseInfo).showLinks).to.be(true);
      });

      it('should set a message', () => {
        expect(checkLicense(mockLicenseInfo).message).to.not.be(undefined);
      });
    });
  });
 });
});
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/
```

```
export function checkLicense(xpackLicenseInfo) {
 const pluginName = 'Rollups';

 // If, for some reason, we cannot get the license information
 // from Elasticsearch, assume worst case and disable
 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     isAvailable: false,
     showLinks: true,
     enableLinks: false,
     message: `You cannot use ${pluginName} because license information is not available at this time.`
   };
 }

 const VALID_LICENSE_MODES = [
   'trial',
   'basic',
   'standard',
   'gold',
   'platinum'
 ];

 const isLicenseModeValid = xpackLicenseInfo.license.isOneOf(VALID_LICENSE_MODES);
 const isLicenseActive = xpackLicenseInfo.license.isActive();
 const licenseType = xpackLicenseInfo.license.getType();

 // License is not valid
 if (!isLicenseModeValid) {
   return {
     isAvailable: false,
     showLinks: false,
     message: `Your ${licenseType} license does not support ${pluginName}. Please upgrade your license.`
   };
 }

 // License is valid but not active
 if (!isLicenseActive) {
   return {
     isAvailable: false,
     showLinks: true,
     enableLinks: false,
     message: `You cannot use ${pluginName} because your ${licenseType} license has expired.`
   };
 }

 // License is valid and active
 return {
   isAvailable: true,
```

```
  showLinks: true,
  enableLinks: true
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export { checkLicense } from './check_license';
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import Boom from 'boom';

export function routePreCheckLicense(server: any) {
 const xpackMainPlugin = server.plugins.xpack_main;
 const pluginId = 'spaces';
 return function forbidApiAccess(request: any, reply: any) {
  const licenseCheckResults = xpackMainPlugin.info.feature(pluginId).getLicenseCheckResults();
  if (!licenseCheckResults.showSpaces) {
   reply(Boom.forbidden(licenseCheckResults.linksMessage));
  } else {
   reply();
  }
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export interface LicenseCheckResult {
 showSpaces: boolean;
}

/**
* Returns object that defines behavior of the spaces related features based
* on the license information extracted from the xPackInfo.
* @param {XPackInfo} xPackInfo XPackInfo instance to extract license information from.
* @returns {LicenseCheckResult}
*/
export function checkLicense(xPackInfo: any): LicenseCheckResult {
```

```
 if (!xPackInfo.isAvailable()) {
   return {
     showSpaces: false,
   };
 }

 const isAnyXpackLicense = xPackInfo.license.isOneOf([
   'basic',
   'standard',
   'gold',
   'platinum',
   'trial',
 ]);

 if (!isAnyXpackLicense) {
   return {
     showSpaces: false,
   };
 }

 return {
   showSpaces: true,
 };
}
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

import expect from 'expect.js';
import { set } from 'lodash';
import sinon from 'sinon';
import { checkLicense } from '../check_license';

describe('check_license: ', function () {

 let mockLicenseInfo;
 let licenseCheckResult;

 beforeEach(() => {
   mockLicenseInfo = {
     isAvailable: () => true
   };
 });

 describe('mockLicenseInfo is not set', () => {
   beforeEach(() => {
```

```
    mockLicenseInfo = null;
    licenseCheckResult = checkLicense(mockLicenseInfo);
  });

  it ('should set showAppLink to true', () => {
    expect(licenseCheckResult.showAppLink).to.be(true);
  });

  it ('should set enableAppLink to false', () => {
    expect(licenseCheckResult.enableAppLink).to.be(false);
  });
});

describe('mockLicenseInfo is set but not available', () => {
  beforeEach(() => {
    mockLicenseInfo = { isAvailable: () => false };
    licenseCheckResult = checkLicense(mockLicenseInfo);
  });

  it ('should set showAppLink to true', () => {
    expect(licenseCheckResult.showAppLink).to.be(true);
  });

  it ('should set enableAppLink to false', () => {
    expect(licenseCheckResult.enableAppLink).to.be(false);
  });
});

describe('graph is disabled in Elasticsearch', () => {
  beforeEach(() => {
    set(mockLicenseInfo, 'feature', sinon.stub().withArgs('graph').returns({ isEnabled: () => false }));
    licenseCheckResult = checkLicense(mockLicenseInfo);
  });

  it ('should set showAppLink to false', () => {
    expect(licenseCheckResult.showAppLink).to.be(false);
  });
});

describe('graph is enabled in Elasticsearch', () => {
  beforeEach(() => {
    set(mockLicenseInfo, 'feature', sinon.stub().withArgs('graph').returns({ isEnabled: () => true }));
  });

  describe('& license is trial or platinum', () => {
    beforeEach(() => {
      set(mockLicenseInfo, 'license.isOneOf', sinon.stub().withArgs([ 'trial', 'platinum' ]).returns(true));
      set(mockLicenseInfo, 'license.getType', () => 'trial');
```

```
    });

    describe('& license is active', () => {
      beforeEach(() => {
        set(mockLicenseInfo, 'license.isActive', () => true);
        licenseCheckResult = checkLicense(mockLicenseInfo);
      });

      it ('should set showAppLink to true', () => {
        expect(licenseCheckResult.showAppLink).to.be(true);
      });

      it ('should set enableAppLink to true', () => {
        expect(licenseCheckResult.enableAppLink).to.be(true);
      });
    });

    describe('& license is expired', () => {
      beforeEach(() => {
        set(mockLicenseInfo, 'license.isActive', () => false);
        licenseCheckResult = checkLicense(mockLicenseInfo);
      });

      it ('should set showAppLink to true', () => {
        expect(licenseCheckResult.showAppLink).to.be(true);
      });

      it ('should set enableAppLink to false', () => {
        expect(licenseCheckResult.enableAppLink).to.be(false);
      });
    });

  });

  describe('& license is neither trial nor platinum', () => {
    beforeEach(() => {
      set(mockLicenseInfo, 'license.isOneOf', () => false);
      set(mockLicenseInfo, 'license.getType', () => 'basic');
      set(mockLicenseInfo, 'license.isActive', () => true);
      licenseCheckResult = checkLicense(mockLicenseInfo);
    });

    it ('should set showAppLink to false', () => {
      expect(licenseCheckResult.showAppLink).to.be(false);
    });
  });
 });
});
```

```
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;
* you may not use this file except in compliance with the Elastic License.
*/

export function checkLicense(xpackLicenseInfo) {

 if (!xpackLicenseInfo || !xpackLicenseInfo.isAvailable()) {
   return {
     showAppLink: true,
     enableAppLink: false,
     message: 'Graph is unavailable - license information is not available at this time.'
   };
 }

 const graphFeature = xpackLicenseInfo.feature('graph');
 if (!graphFeature.isEnabled()) {
   return {
     showAppLink: false,
     enableAppLink: false,
     message: 'Graph is unavailable'
   };
 }

 const isLicenseActive = xpackLicenseInfo.license.isActive();
 let message;
 if (!isLicenseActive) {
   message = `Graph is unavailable - license has expired.`;
 }

 if (xpackLicenseInfo.license.isOneOf([ 'trial', 'platinum' ])) {
   return {
     showAppLink: true,
     enableAppLink: isLicenseActive,
     message
   };
 }

 message = `Graph is unavailable for the current ${xpackLicenseInfo.license.getType()} license. Please upgrade
your license.`;
 return {
   showAppLink: false,
   enableAppLink: false,
   message
 };
}
```

```
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { resolve } from 'path';

import expect from 'expect.js';

import { assertLicensesValid } from '../valid';

const ROOT = resolve(__dirname, '../../../../');
const NODE_MODULES = resolve(ROOT, './node_modules');

const PACKAGE = {
 name: '@elastic/httpolyglot',
 version: '0.1.2-elasticpatch1',
 licenses: ['MIT'],
 directory: resolve(NODE_MODULES, '@elastic/httpolyglot'),
 relative: 'node_modules/@elastic/httpolyglot',
};

describe('tasks/lib/licenses', () => {
 describe('assertLicensesValid()', () => {
   it('returns undefined when package has valid license', () => {
     expect(assertLicensesValid({
       packages: [PACKAGE],
       validLicenses: [...PACKAGE.licenses]
     })).to.be(undefined);
   });

   it('throw an error when the packages license is invalid', () => {
     expect(() => {
       assertLicensesValid({
```

```
      packages: [PACKAGE],
      validLicenses: [`not ${PACKAGE.licenses[0]}`]
    });
  }).to.throwError(PACKAGE.name);
});

it('throws an error when the package has no licenses', () => {
  expect(() => {
    assertLicensesValid({
      packages: [
        {
          ...PACKAGE,
          licenses: []
        }
      ],
      validLicenses: [...PACKAGE.licenses]
    });
  }).to.throwError(PACKAGE.name);
});

it('includes the relative path to packages in error message', () => {
  try {
    assertLicensesValid({
      packages: [PACKAGE],
      validLicenses: ['none']
    });
    throw new Error('expected assertLicensesValid() to throw');
  } catch (error) {
    expect(error.message).to.contain(PACKAGE.relative);
    expect(error.message).to.not.contain(PACKAGE.directory);
  }
});
});
});
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
```

```
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

const describeInvalidLicenses = getInvalid => pkg => (
 `
 ${pkg.name}
   version: ${pkg.version}
   all licenses: ${pkg.licenses}
   invalid licenses: ${getInvalid(pkg.licenses).join(', ')}
   path: ${pkg.relative}
`
);

/**
*  When given a list of packages and the valid license
*  options, either throws an error with details about
*  violations or returns undefined.
*
*  @param  {Object} [options={}]
*  @property {Array<Package>} options.packages List of packages to check, see
*                                  getInstalledPackages() in ../packages
*  @property {Array<string>} options.validLicenses
*  @return {undefined}
*/
export function assertLicensesValid(options = {}) {
 const {
   packages,
   validLicenses
 } = options;

 if (!packages || !validLicenses) {
   throw new Error('packages and validLicenses options are required');
 }

 const getInvalid = licenses => (
   licenses.filter(license => !validLicenses.includes(license))
 );

 const isPackageInvalid = pkg => (
   !pkg.licenses.length || getInvalid(pkg.licenses).length > 0
 );

 const invalidMsgs = packages
   .filter(isPackageInvalid)
   .map(describeInvalidLicenses(getInvalid));
```

```
 if (invalidMsgs.length) {
   throw new Error(`Non-conforming licenses: ${invalidMsgs.join(')}`);
 }
}
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

export { LICENSE_WHITELIST, LICENSE_OVERRIDES } from './config';
export { assertLicensesValid } from './valid';
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

export const LICENSE_WHITELIST = [
 'Elastic-License',
 '(BSD-2-Clause OR MIT OR Apache-2.0)',
 '(BSD-2-Clause OR MIT)',
```

```
  '(GPL-2.0 OR MIT)',
  '(MIT AND CC-BY-3.0)',
  '(MIT AND Zlib)',
  '(MIT OR Apache-2.0)',
  '(WTFPL OR MIT)',
  'AFLv2.1',
  'Apache 2.0',
  'Apache License, v2.0',
  'Apache License, Version 2.0',
  'Apache',
  'Apache*',
  'Apache, Version 2.0',
  'Apache-2.0',
  'BSD 3-Clause',
  'BSD New',
  'BSD',
  'BSD*',
  'BSD-2-Clause',
  'BSD-3-Clause AND MIT',
  'BSD-3-Clause OR MIT',
  'BSD-3-Clause',
  'BSD-like',
  'CC0-1.0',
  'CC-BY',
  'CC-BY-3.0',
  'CC-BY-4.0',
  'ISC',
  'MIT OR GPL-2.0',
  'MIT',
  'MIT*',
  'MIT/X11',
  'new BSD, and MIT',
  'OFL-1.1 AND MIT',
  'Public Domain',
  'Unlicense',
  'WTFPL OR ISC',
  'WTFPL',
];

export const LICENSE_OVERRIDES = {
  'scriptjs@2.5.8': ['MIT'], // license header appended in the dist
  'react-lib-adler32@1.0.1': ['BSD'], // adler32 extracted from react source

  // TODO can be removed once we upgrade past elasticsearch-browser@14.0.0
  'elasticsearch-browser@13.0.1': ['Apache-2.0'],

  // TODO can be removed once we upgrade past colors.js@1.0.0
  'colors@0.5.1': ['MIT'],
```

```
 // TODO can be removed once we upgrade past map-stream@0.5.0
 'map-stream@0.1.0': ['MIT'],
 'uglify-js@2.2.5': ['BSD'],
 'png-js@0.1.1': ['MIT'],
 'sha.js@2.4.11': ['BSD-3-Clause AND MIT'],
};
/*
* Licensed to Elasticsearch B.V. under one or more contributor
* license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright
* ownership. Elasticsearch B.V. licenses this file to you under
* the Apache License, Version 2.0 (the "License"); you may
* not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*    http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied.  See the License for the
* specific language governing permissions and limitations
* under the License.
*/

import { generatePackageNoticeText } from './generate_package_notice_text';
import { generateNodeNoticeText } from './generate_node_notice_text';

/**
*  When given a list of packages and the directory to the
*  node distribution that will be shipping with Kibana,
*  generates the text for NOTICE.txt
*
*  @param  {Object} [options={}]
*  @property {Array<Package>} options.packages List of packages to check, see
*                                  getInstalledPackages() in ../packages
*  @property {string} options.nodeDir The directory containing the version of node.js
*                              that will ship with Kibana
*  @return {undefined}
*/
export async function generateBuildNoticeText(options = {}) {
 const { packages, nodeDir, nodeVersion, noticeFromSource } = options;

 const packageNotices = await Promise.all(
   packages.map(generatePackageNoticeText)
 );
```

```
return [
  noticeFromSource,
  ...packageNotices,
  generateNodeNoticeText(nodeDir, nodeVersion),
].join('\n---\n');
}
```

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

PNG

IHDR.GsRGB@IDATx`VO
{#KPDGmk[kZWu{C{%7||29W?/w9{_\u_|BQ>;T+0F#`hd7$$D%Q*i0F#`IMF#`0Fq0q#`0FLq9
o0F#`S\G;[-0F#`IMF#`0Fq0q#`0FLq9
o0F#`FJzrl<|^N~v IjZFIdfHFJd7t.\b<*)W;"v&Ki=]C3fm9 k]Z\$YRXTK
d9FK]E}0SJjSE
v#`86^q6g~3k,/#p
5M?k=#}U2kk&/M*CyDjk*TiQ"zB9$=`^Sz'|,_IFx?O.zq,U)bvR/I59i/N9uLTdgfAv@iK{R%;F#PB02l
ekJiE=JU z@:U 1edw GJKWITx>UX'79RJ<m[Fx%<sQLGpL$+FAJ%IQu"#H
CAWJso%.3OrX\"bbkHnm'Y{ZF{nyGd2Zch-TA mKx6F&PXrtK*+R+= Mesnl"Np-{e-
1SBBA=%88Hd4LYUJm}CTR)3TQZ=I2
k8ousu%vR#_/wHxfr~rF e|'r`6>Ar}dbBnF=w,)Q--s[oe|lS&O#->g;(:]kPEAA2s*.ddv*y*\Y;D6Q Pl+JJatn)!]cBY3jH
BQL[YZ_$Xzvh#7$[d}kl+wJI-
w7oYEk;y(ktn,YZ#{ITP)m3lw1\1eg7i,,TN/ eU6Z`\>$^~+?V,?&l4JT]R3]Vj3oPE&2~h/4^
;v!e0wgA4,O$]{f,i>rF^Eygf k-g
!C6n1ee#-UkgN/%lyeek\p^vLGoM"F.Wz]]f_'q[57'IqX9wXoJi s])9u*hzLYOGT(aLoO1up$$>%2lWlOQ-
qei\7I|8E(kwIZqU;*
y[U! 6o{2Ytt_9-$&&S~uD?Z 0F#pj?B"9Q(Wcb!F;Zj]*vqb!FLR[$!JFV\,^^mAe<?o\BI-w7JK0}Ve`yxr*),9O;JOyD-
_%#-M4y[CHLL[?\,MD+[>Z([%<o/ =YxifEj. .T
!,R%Y*aY=/1]JC0@Rs$FT,;GZv,~6vSVm&RT$w@KW%o/M]vR%M qOd 6_.)
ON-B6h\D3OINxSf7mS5
7n/)^Z.}}IV%3(D'syru2/KK`h/})!A*&ergEKnQL~{<6/)ysI\a]p ;S~ZY\BJ/|)/?
Lvy2FNCW.>\FIQ0{tyr!n.JPA?'7N(zOvP]U?Uz:9eNftkTz|E
RJCTi[GfGJUHSr$?.\+WDPCo^J~hDuz]YL(^)K>D.~wn-
C:4nbUb\#4GJ>o~2rTiqI>Z4f?X)YK70i4Hhgg=.#dmy$3%MUA| `#`@,.*2~mzK2>:<.xQ\Z*\<<e2oo%
d]]R|rGfGr%OuAq:wH8z]2]tG#!Mf|-w.LR{@%js.2gm.KPQ~4'
r2pF#:bi2RYr^d^F,gY[d%L]
TP^DTM-
&tt:@O)VW]&kX<~|ylZ)PbU(rvxrxe,mK;>l[[7)kh[jZ6=Q~Y])uCYf#/+8V,*jYu"<c*R'~FbwoaV|d!aH:PV;EDNrU
z*m%3o/

=[{z
e/m.orUJ}`ZWvYq{Ter&|sX^*eC36EO_:]mj
HemK32?N-d$<}PZ
VX'/nb`|}LO.WYqf2#c'9?v|\ZJeZgTxYxim&v^!gL,{wQ&Ff1AlUe4)n\p85Y52I(d'i!ooI1CzY
RzJXa<2|z~[+Jfm/zO];6J;W9<}H\ArVgwX(gQ~1 Bjvn#az(q<vy@o?O~}Yy^^0FGv
B#ztS8;k*=C(Tg:Tq,_&Qm\u}W0vgCRW*y:SeGmVWBB?G"n6!T<u
~ieX7ITYn_{7\, :yHtUVT!ro.G35Be7(Ut Epua":Z&v+sGnyE*Q*V.h;/I*
OhZhLg}KTda*'/G6L"]P$t[d_UJ9iFy;J>pF ISM(=CH\ym
UUP<?S\BT+ZW'bfVUZ3 G7?ArSpl20aw8.1JwB:Q
vGGBJU*+!N_jHZ6dT\g Q2EGQvfl_;r|/DP:Gw2PQj(Y*.r<\fyA?|DJK;>$9,Mn(9+SZ8`n~{od73xDAJ
>b2"'#Mg,yUJB}n|.zEx)-dMP%:evjR{^rP'6C=&qKlQ{=$"JNQk { {)|k0F NqJQZ^u;^L0;Vy%J
w
n'?\GHtNVIr{wP9c$UHZ>d`1-pU
Vn;[!Ewm+_~&eWGA!=[nV+SjHKd`"TZ*NR$H%h-7j)-EN9_SY{
eGgKVA,I%+6=GwGOi.1jUsZ\xTD]
LvsnhcUi;k\)eNyu3T?0g9%NmZ+3ul<w9u[(F4>d|\fCo7Uy-VNuS
j(Z}c$a8|"ZCQ3\]PyGDV%z<\dg2w$m[vNh~XXbWvOmc[u2LbIUJUtVm7C$'4R+K
/'.l,O](4WU6K~xthR$[y; D=Lp\ZlZ"/!DZ!I,<\eLktozg+/
#e<U<jCS7Rum
{]WTq~#`CV\0Z%Pk&2K;Sz_su1osOWy\T ,\Me]uF8deO`L BTM_Y s8|<g@"FNu5P/KuR=Jsy$EK56gI5B
J.:O~tL[kR^#=I./SrD^]P+^Ib!~kTdmJD
^{~9HKmIzs{_kCm~enx,_'Swj(M%|.ypjYoucT}BZFvPnp|D4m5EfZU Oom\yE]%R_:<8.+/ O+ygzY[1[bTxI>
[Wnpl|kmdEyAUT^ZJ[eK1{LO=U4x2GY_ Um*9W_R=DD_.h9 wD]TWU
e#`0uDIr8d y`P]Zs
jexG*.LG`NE*`Ws<wWUDd'&cABAUUc6w,GK
;h
IEIp- gPIr_u*_KeS]^U_9C;:]x
wP%u[e5d$uR"?'HQT@g2UePb.6sXZw$jSZ /\nJX|x<4u$^ww8[]5.mtwQrGV
StMQ)[I(G4n!O,N*q(p}:Uz6W.K Zo-q}dHOPCim%]H!2zP;nbrqo8nbS<uS\9l}1wd`o&?4BjLzPJ

*!5eR jX7Y7Dn"GF/T_&_bOr3Ru>*FMxh+Iix;yZvoF;-]"W8K7g(u_T51=m:
jyr)mTA(7G+3tsz#ac?0F%sGB=Df/)7{XyKUx`Phh'oMP6;ox/ya"9I;Xu15\.@VwHi1VZ~/ct{RU|l}xL&^Y}Ui}+=
s>M`LN"D^U9a(<:q!/K%<KNnzd`\DYyo^=416=eBx-cBbT1oyU %:8_gJ!"a=y;<d`]L=/Y-.MN-
TD]p#UQu1&i)Jw_<AQm:^*s>O\.Ov1BwV+
g.%
i3nqgLYE.{.;H_fS cd*5N2<sF<"ys:U`_[z<XKx{uDJ#M]hBU*Pa#\h/3X<bJrAw?O'/^1LrL~wus][KI_}](M*
o}XR,ZGHnVnX;jrd;^mZ&eyf"i/:U%2Vwrf6D?L[,(Ua,PNE%Xt(H[`FKdh<pnT'tSkt.u_N_an]j=hVY?]Kcife}=k/|R
#NFly 'e_F#`'S9k[RSqaOGEJ*@pL:dt'+J54 2]P-XBN]DTaP+*XCu-HB,rrdO
+LQ^EETQub=uPl<yb:;w,~\l}dYnq<LnwJlMXdu<Pu/UruB{Pv%Zr
uP>IgVw`dX-8_k]H#`Dg|WHl @*en;Psz ed)Vt'Nb[Fd"jtBR.E=N@y.NbrfXhU+R-
cn.DzmoE($Inn!=FkNtQvmKXJy/AqAsU7}iL`>0(G$uLxi&G>vmDOWa:Z7(WvgV=DS;anJiH&2
&J:BrPwTjeR5Y!V+Twb1FbU25O(>mZJ|lef}b^-WjbU&zRD@Z#t[31[:EF-KWm?E""'uBJHSzRkhvi*{]:FZw-
{OdkH;;y,9ja+rOuiZ HPd(TOn{PLskpzX({e[(GPm)X7%&F]m"'-OHd^FtOk>$\U1F#@BB]Vbu07(%\v/(GRq97zQ
yZQD^"/`~#pRX%=ule dg7<2\v5F#pJq=a5e + 84B?N~pLJS:t|GLo\qj#Pz [,uG U;A5Li0F!hSU})!a:^e .+=%]o#W1D.
`:^E|s;ui?8OfJQp@i(w]L-A#`0
W*_wyexl{@X^A
gq!^Os/=.Z&F#` :{>@;#`0F4'Tqa]ktM -JCh+0F#`0U8aXE^P-
7TQH;l0F#`@&Pk9vpMtOg{/;fcO4F#`0FKI/CM7Bk,'e//

;9&~4Sr~qq#`0F#`Jd@Vj_oT+m{$G_ Q{C{OJf#{oF#`0FZsSK
#ChTb[z$KCY)2(1lyEOU;bG0F#`@Bn}p#KBTUbD0/\,_lJKM
0F#`0GrU#LD>9%jb0F#`8L:.f.7)XnI#`0F#m'PK^AJ.m[c7~G^jAtK2,#`0F#`j$PS2=DOK3GH0g#`0F#`@uMq]/,/(*
FvSKKTDsyW'7)6o#`0F#`*7%-+G.r2S;<~$Tn;b0F#`@9zQ\2sr)-
T&eJIS{\k}F#`0FT&PKN~X@E.X.|}ieqUKV6Q0F#`%PK~a$RRUE
UY Tw_\<^Z4>l0F#`0{Q`1_KJd3<7@,/w{&`-VE&P:v#`0FEN2KJV%KCb.d.^8v~rXAo#`0F#`m32%
R%r]wc#`0FFJ[\mso{b)":%e|( 1X9#`0F#X|#%C=lIAiAPYrzxY]j#`0Fo1cV\xdn{_A,`t5
Ml~>v#`0F4&xWu!EEM6Zf@v2?V{1e Fv#`0F4Go{NbXn;1MR 0F#`0@JK&;e^%OXQ3F#`0F4NV\db_(-y;9-
tK$fI#`0F#pR.fcHRa{Sw6Q-br0F#`Lv-Yb~_K#V-[L+s#`0FF@Kj(XV|?%Uw[rX|,.OZ0F#`@%Tq}-/xj%Bahtm-
l#`0FKvKH$"<|UhJRcZt0F#`8INqQa_rjJZZr$FX0F#`0FS_#`0FBL
!F#`0FTM#`0F#`S\HCX10F#`).U3F#`0F44b#`0F#P5S\fcg0F#`h Lqi
a0F#`0FjT#`0F#@@a0F#`@B>u
!RX\rVm0F#` SRRoS\JK%22R%0F#`0
@HHHRRR~bhh Ta;`0F#`@$XE |a(JU*jyc#`0F@u-
?9Ji0F#`5S\u[0F#`AF#`0FQ0Q7U#`0FLq99Ji0F#`5S\u[0F#`AF#`0FQd$(4V)c|oeF#`0F[D~4^XX$c[EFGIXHdd]R
RJL[*F#`0F
:On9{thR.?<tj<i2o6&a#`0F#`Vq).Hy<d_T$[]ddn2I1cv;_|WT]Lyni0F#`@eu{Xs]]uMEEfKN?F3dZ`lKF#F#`0F]tM
N$(-} S}G +9T/M/^.mZ#%`0F#`8LnUT:m-^xk3Q{UjDZFe
WN>57?.~]Gz~#`0F#`jm;lyj,e[qvr2ap/t7
y+sFR3X|>0F#`hdnDjiQdL2;OPZDD^{E/uK6.#`0F#`@9:S\:D{InIHRxB+F
dO-['r`;ef[_^@2F#`0FRB^(IP%E3
j^I]P,]_=,y1F#`0Fxl;L/N1/TfDC[|$edqxNZG6H"9O.Yy:0F#`0G3"UF#qmTyQe#Q,XLWK})4uE-
,{tdF#`0FrucqQ=c]b
gM' Y9y2l:l&zN/1L>kWk6KI0k #`0F#`@Q\BBe~3g~XJZv<ZbNii
s\&]b[{(jm)[+SQBa0F#`@'Pg>Y%}[^ dn\ZPX6]D 5B6MwJ`0F#` TU"<dsu`y0E
uKtDs7^7B:mOg?]f/m0F#`@'Pw(u
_n\K3j)QD_ILMg>%OO%,c#`0F#GnWdr#/EC>$ZMnw;
C_X}1iO#`0F#`W\HYa_!/^-u{fMt]~.q)dJEB5{v`0F#`@i
XPp+zT`0F#`
S\\,[\m0F#`@#"`E#jl0F#`8Y r#`0F4"4#`0F@q9YX0AY
R$!!!JQQQbrV*H) Z>abi@%`Km+o8\?@u;?P

$;;["""<X_|SeYRR"~o9/999
;*S=P}}>EAM-Lg222-[kk1@0~ZF^ 0z5kVl:'BF^^W>FxlV'5B~c1d5.}nU~_;kK>}[aaO~O8?#2x
r={H=rvx_|!O>'|vi?S4sd(3ZkyY^s5rw><BF +/~!^z7k],7xC{9O'#FO?
Q@[l?\[Uh:S'=z:B0e1cgmAyy=ax*]0'c=[7o)Yz&eGhW{}rWTLyU%I
_+o9~?)N\yC*M~#7F\P.p8s>@nf:,O$H3p&&&K/$w]tE55YN.J/Sw)).-8
Bys=~12vW[NvvYi3,=lQ.]TiAw8~Z@m b%e7$/bm3s!3\c*PM@KdQ@~il.fHdq
cE9sk.p*,><bB@BBQ
2/8qW>Iy
'),&Nh'~z}~HJJ;`(cccgg>_e@aU3Fpd7[egF@)/ }J
pj5`u(k8IsB}<71VNc3P^7[d ((0' !3rH<;7ry]vcW',nA#1/{.qxK*kHY9Z0hhY\#ptZW
j
"1A>A]<_=u5j@VJMMt>,
k?pap7,6U 1 3P8G
L|H:(0[OW6r"8&O,K,{]XZ@,?e:.pq;P O(%@tMV]-/i68u.9,-
|GSNxpG#8\^$M++i#_n<5/EqMuri.EU}kk`6`U+/K.iQe?(/@i1:u9oW`'1}q:7>,[/54TZB+8I

)sg|%E:h(*r3uI.pzOd`Z].Bo7xqmE;94+Vxluko'f]',
G`#OG$D767*U#9sggq7+oL6;dU2PVSwt-NJwPFP^rJ1]`9h .$j*IHH8bbbd "iqy._u>:v#rt+c
+@IDAT6U;r"R!CH}=E,Xfv!OcN:ywGzY<}v3glMJx/ebW_}%i=;W ._tH{iQ6#]|2"t;G\6rnsyp7y eUrQ~S-
==ey<,wE6lw-I&>vWEZ>PQ
/J w^R%eMF
}FA0A5D <$TZL~={`\
:f@IaZ984pB yPZpu<vAu30#>30@\p.G]wu3.SRRA|x)#_x@FY_r-x[gan+|#Jz >.~3&.vAmW(O[Xp7r'|z
w@EchK96`e-y'%~r
O>x:(X$W3[Ji !];||:_i/D}\6YDpMC0o \ho\{M<KyRo-q)wEHKOCcNA;~?p?6c"=;vc'0G}Sr8OG1q^%
#mEbxUhQ|apnv/}J~O]Hcyq
e_W^bc?g)uhB4@
d0!:KOUK,F*yaYgSS`!m.{uo 3`@d [oJ^z-"Ao?{d1P.%>
.7qo%k"|Aosf[7&}>[lG=Q|"O 7X7xRV(&?<8.+pB/Nagrioqhkkwi\Harae!I9\-',3'_{w/t8o}-y|OQ!H=5L&\:]w m5X=ro
.jeB1>O|~k7|ci>(sC:m26aR>i#}{oiS6C&DhwxQ'kHU>;I=OHM:Do.8hs\M(=%et'~s_Wh
V6#ps.$aG
U)AzKh3@r;\^yoe5-<7;` ZfN}3K
Vf 0^f!d~3p#K>05
s`N!
Bl}{aao^Eb.D`
oZN`
{7^Mq-!a^~^ifq-wQ5PWfcQBiw #\M09goZj\D)'J("C_Dt=PPli<p
1G8R5\OxUg ;,*%xNH%q'19v_Y7;R];A4rMK0dBr]S
4Hs/3
xPjws5#q/Q/Awek/L<(?e&\p&CA#p-n|^-}xF\ve^+k&X\/x|MyhgVU,Cpr,9^;n#`Kut8FE:]%Zob(-SG_~qE
$Y:""(3k`f4?#ObE1`#T1]^VB5-\+z[4H{,SVf).RuYx!h!2 p2L-uN8D9@A0$}"CC {&MA

U)):C`=B6S.5f_tGc)-W0H,y ._r2SbWNve('8"r=mM/Ns/Bn6A7<f9Fr},>1
9'oM(?{4ps|(w(\z_}>J!?"/e
16Po>Y#}1Gq\KYq)*|QY#tyX9|PJ.=z'XCaY#PnlL8O?
..V@ :`":S\u$$h
nA_Tb7ad'VRRXFu,=Z6 XNiG$1?m]3 %T5`0BpBxg-=#k*=00;M.W_]fF/poAO?7KF},AA7,``nJR*`Q.>Nh
]~#Haa"`Ga).N<y?od:ip=@bf~C .(<y[\7~;O?r"(|Qv!{C%co,,b}&XPm>J}y5 Ms-
,:sPZ?c/`!9"|p!P/x!rr#v37e;@HpV(X(.vv0F^/"z,%X^GGd8 (zQZ\MQR".zXphy
6?A6?3J>xY^\IV-zlF#p#p<VirUrRZ#XZWR{C5~3h{
.72<IgA
N8d8xVIf?4+PI9C@MAA?P<|#0pu\Yj"sMZvPB!}G v.<;_0G9FGL\C[!H
~6qcio/Lp t
-/e '('V fyKoc~Z}oxf]Uu8i:3R-\"^bB{c,,]^
Hu{,q'@ix+ cKcYGu'M)|to]`BoF<mi&ZkXC\Z
k,oP][Dr}OKZ#,S&^,jiZ75&a5*!5f,2xgs_B3|n)B%;x)3U#kD]&3eOfd`o?J8!UZKDT'pwfRC^Ghw#'@r=-@h~|cOm
K,N`'=pruwE0D)uyCqqor"xbCUh^"}rPM0XPg2-B>W
D${kbbIB]o\&o>o<k
K&u,a#ushFq-H--u'v4TVDNE>A8P70#F}cMH6G/9vq
~_;P`bz'7\
Nb#p=BC 4qfYqq7M]#b+*.Fhu"23!Iaq5Op$}@YhO,1
}uX{RUN)/64r9%VYrO]?
+6g?>Kp:M"URr;sU\@%f7d=/",85
[2w:[10@~@yX+,u+,yN<`(.4Qy'q3D|I[*$~j&ic:{6:p5_$rsO+7~*-Oik7]M]X X&f \ buAw

]SW(N89BP5K.
D`g*f sqpx|X:2Mz,h.IZp=}hSviOxUzgW7vkc8LWe6&@u}IY:z?;lJ^=K96
aP@PRXc$y`muRF!WStx_vov)'q3R5>AG5k@}3SClL._/8fo#B:GRXQ2T& Jt;}Kq%.k$}4U[Wu<MfYDm
q'e!
w.v'B=5[b/X0B:u5c.`,f&L Ff{;FYbk\7SJ,])#nBj[!J(\\]9qI7(;()3eUupIc'+,{+phQ> Nq=k<?RUU7/XC~;vm);9Ko[
Da{^C`be01wQ^^#>G?;=pY]g^oJu-?RujNI{k`c
r0|~3A?.3p]y4Y&&!|3xP~xa(K`G*(&YBeYBqa'%\jh7=@7nH T+'5X#X~z5A>2xX
.Q,n}LmKoTdI~DY]__;QY.^(JL\P@X?D<Nx&p<O
@1fXeYE]q%Uauakihgp?6}=NKeneFD_tl8

,*(>.W]/uf(-W^0X~N O(n]2b]a-u'P',)1@@r`KM
tCV!?B<)951I7kxh@N[&w7.S4q~>?}yedKVe=e6<VW[}(-u!"an8}+@8O:Xj`@{aWn-e*CXU}7z-5m
en+)fS]_phMqiEe=\G@>vcH\ZR+*Z.;PZyCT+x+\67@y!FpnCe[MCzjr[u5r R/>f H"P?'D QWL1~/cc
hXR4!"_&PZq
vCp^R$_Vy<PQ)'#o9i+mDiSiS~G:Yf}rpl)7\9eCF\{+q7pvo_}|G/\9IzrgJ$C"s.?BergJsK.Oh'\DN_&H\po#MT&)F$e
@HR?w4W+Ls--v^yOpcaf',2\pZ\[J<fXqQxnv<b%T4Ae9kBn>Kf@?qAb>BLm_oe&3
`uE'moP68FYh'/^!4X#sXk+;.+;TpUsAaWXA(SO=AoRNjAGe6E#b c%}K'2PFSvB\#,SJI`y'tu?}
w,,|XX^{s4{$zQ_9Ow>}0+?mOz&w\~o+(Lp13-bWC\i'Of{>!!a":ge+u&.t2#P P[#,)Q*DIC<.Y@-K8p
NfF(bcG=e*E,EgevW7 wk>BB<euuq?a[| ,E*%kY*6m\{Br!;!h38#o{# VtL(RI{<
WNFsY?yoV@pL4Bc=u %aIn1w_&\FoOt3aA2@:VWuVWQz_Y4O/AY`RB:>'`EX~ @nX+_(C({(J!$,QTNbB
1@_/(m"v& (/Xs`G@qn{.o#PMq UUvpg|-]+0m^(2.F@A(fpgPFbf
A%`vbF)b]<*:Uk;%S|A8u>8xL'D9f!YO,5
[ff8G T'oF\@#OI4]YHv&-4+070DM.uqH?D:wyg?A8s]vH?O@g7uiy'
ooFesmE_b]wQ]YO2qO@'B
Z*TiommZ
Z-NpwH$c9ox$$! sa23]7s9{SqYMm28q+9*9s=x%i6QRW.,s;!
{XH6rY<qgOddSV]3|5=|
xBPSi%9?|^j./0GUab8<atE Yb*wF_nI#R#hq6:~==(FO_8.'+#u.P B@>If9+f6K,QVIy
<M~Z3`QySY[l+$+<0/Z7OjYuv*/u~V:s$2 (@Mf?6]\zx'l3uX'+e`Wyq_W99~0+l\:pCWJ=Ovy#<A- 4
jtZq|~/Lsa^gMO93O\e?zDa8=s-]g&VuR[rT'K/`?XeM3Rxfopyx6=(BuxZ;S(:mpdm9.I5oX&^_58?}\p^ya}]XJ3?
:p[/V7f8-|X4j|gw;#s~3,o~2Gy-}oq< Bn"GLo_@VxH+;AO;Nxqg?`(m<Mui!W#EX [N5$Z&3<gs%O]=-/?pz,
Is9\>y835s^<['l^Su|un3^f^nU&zuv3C8?p8-soO-K5zDef?\.yq:p[g<dnj=zp~|}
SV=8aG5afXCqj|]q[dMker5.j;qo/n|p;;m{?s<z,-|}p|5[2Tdm04lP^=weWkrXiRBQW2.=}YmWwfqzn;
+\??mj}e^*["-
O|6\9'*3HL^JY(j$CR:TWN,_xAPTZ[V:W_n!^[y1]m6u8?V=WWPK9:enj1QPWtw<{wqejp>j=K;aGm[eiJqnnT>A
}Ej9j\50#qQn*u/p]`>,]m5/ac|M'
.$Y5U^uK:!p@Vx{jWK]@|^v]sf)LRfUX2~tgeWfu2U>esc)2qL[lqy[mCO/Bn%`V'.$W(K;g_j>N!
B@{mgyfrrk]\dT=I)B@! =F6AcEZ,&U$B@! B%`3E1],B@! Boq{Hj! B@! @JB@! B@.")! B@!PmDp6B@! "
HJ>B@! B@T\
d,B@! b+B@! F@jC+! B@!`+"#B@! @JB@! B@.")! B@!PmDp6B@! " HJ>B@! B@T\
d,B@! b+B@! F@jC+! B@!`+"#B@! @3o?#! z4_[\%eH(\F@prtVLqEGJw h_dr3GPkW1g
]fZ3xr=LW$&#2ni$]A$ETCdFz:F-/8`_Wdql9}:4S
,'$e"neTkp
V$'=o9?0
m=qQ$]+"qsl,YZ4
b4<$s9h:-IPx'/%C`; H!D^xw`<'.7hU6U`9 M5N:303O
Mh3O{<arU&@Q8SYN=i55\H1Ns2swk7 e,Y'Inw
vh:~nQiMu2Z,3"%6>

rX'VQ{X-7uETSUU=]`(8FKQ|B@VZ~M,h4bL"6KqI(a-oi_y37&b<&m;}~=:O;rRnVl][Le$
z22^Ke{NL\OY
:`33P2tF=Ecz@itv3at;6*fzw S2:t=Iq9)bS#NsV3YQz;63PH>+%X(GU;P1Ty}%x$F&{JJb,nY*"G-
$+;hE5S{%ri^j;E[4l^hAj;o oODr]<[$1Dk@&FC!u]<IliQVuB@ .z8gW?$m7w#-#{We&dQgec|9m7,
?).2<o`f< [bHAW2vv%Ac@f`h:]#p5Zl_<<!rs(l[W }btS3tYe3Nmc"aW2o :5MqdV$90GOCaP~p"y'vc=:Ikl
Xh^ik*d6PJGql;!MC.a0=zZw-_WYkg]Ohpm'^.ZP7+0j9CuGu:
F%ZbOi\c63a&\Mk~Z7'aC{q&M5eyuI0*"pZ7.=0WW8AkW !F #q*N5kwnVCiqbJ{unM\a`:{&&*:
0$azH{?j
/B@"?mh4)zZ]|~2hras&4 G74~F|8/]SV:},mj$! kbMEpth\v|8@?;gbBh7he@x8bIa(Y~Nht\nOyL3a``wX,H@S
xdG}Q!?bI[`Kcoq
s8LBu;qt,\~ZOtEhH8`oq6.=(>~W`OW@+LM{WhrYN($lL8qYdjq%.1|;@cfcJ[bX\\qx.=wUI$xF+iW1{7\@ONEHl
ceg7ll:rhQea:fvgQR8|7q!,X'%@[v5c$]rAh1d9?`te`C:geB@E6=iS{(e|q7~l&g?= A
1XCLR:a ,'wnLSe5Z ! n198\=izw<571fc"<9+\f-w s[iMj('ZE&=idIKI8u"1W}CMfr7}
^GpNuSDzhH=)0D!ot$p~huxfeGH<n h
o@LZ.|/G!I$_:O*jSg3nt%Iu~
'Q
vBf%Xv#a8&=k@9R9Xs~G0{[8k&siG9;HPB8!#|Q`3qJ"&?zPw`h`t3#]e`/bO#~
3ZM:7L=' tmG*WN%}pZLC3D2w [Cv//! &`ll{F'R2f"VIOD+NAuq9)
.EaxjX/WoG>u#;CB@Tl}[x+~
<sZ(((cEAVP\kPchhVkSRuF$(ihHgW4Kv[])N{zxY>Oh#>/>]B!87ITtjHZJwL%R(5#"2
:\Iax.]_FM4#aU
\VM]i! m.
7V@ngCNtD4t48"K'[y)#Kc'i)
+``6(^&rt^3y mR|v@CF^cI-^NhV
/hn\iOGe88y]f8*r
H,uh
l-:V&%6Qapd:>ND]:"^~QUB@..T>$pQ n,s/GIsxjhO|I~-{bNZB~8M >w/3drS'l!?A4i7:
MVERjy=famhO[+U2m*3PE\B;?aMHB:L+4-VlyHr2+#tD<8,Dk=]J[DCt-
~qmxBY{jxf=Zep|Mh~(N|*:3Aq{Gx:+JlMZ<W@."s{{8K
Kul4<]6n~pI$T
UM[E+Gx;=_Wgr7$tG9Z[w.<|k)Z#rr,|m'{TB@T'rF[/("!d[iBNZ=GqOlj[B@4?u{<}#HM#1qDlK|nQhRvb!B
j,{5pr Dj9 h5VN;Qcv?i$2)%)DO>b3W\1r$u=ozhl]qS$\4o>h84|4i.YNu3T':7v)3e{:nZ-
aMEy=~8[&J,I@PrnT=BA^7Oc?<P
;YTE$tw#CI092Y>;l$_cUhb@MQv9c?KjB@T0$(r5#iKXkr?XZs[v?>~h$V<V/|nTyP^+hI&B@j$`Zot{7a#;ja>+w
hibvOBRr-9/zvYv#JLyUYP[xeah9B4Ey!"*q1QO;;EZ ^9c]>'{$9Z^^6\F]`r
s/(zJ/*fy4okJAyvB4k&mD@IDATK"p>Wy)Rm?oA.a?$pl{ZO8K.rc<-
1WP{g;q
WcjG}'`"A8>,pGLG"m6|H"!}"iCM{Cbk9TO63O;s~mp t9"-8t6R
_"lqIJT[-*igx|",|ax(&XFm29jNc#Kibh!-4fs0GrEFJJ6-;9#?O&q'2cv|~ 9_*P8?8QQwH,:4"M1Xr
v3#OS4n&1er.VhRB;Z6T'G~&1vsrP42f<ihQ;L}p<B#}=G#}O>)~^ YMIiZdFPIs c-
gIsQ:cP^{X#d$A2LKcF(C`0vjM"#j7i6B#:o3Y[`Xvs I+0`e'J"V5]oTJ5@8yh:
!zZa:sX4XiA
o#[Wazr(xv6}jqTx][M}~:|h848$ff,"vL|TgLY#%bRc;}je-
|:e4^;'"b_#u}(Qe?wFV^V6B@TVsMy^frO5k)>da'3=>Lz0z(.CG7Z)g(pvu'24sF*ALgP$zz*fA9UnIu4i7|A@mJ+4'
4@Oox"oNEYdL+<=g6zs{1w*3%.WLv|Lp:F9F8%p|jL\iR -q`(}~.Rt|xiITfq`dEx;Q{ksp*R>]Tv'\fDNkb1Q"<#z/"Vf
E B+lVO0S')4Ar{-zxfa>S]L,>/B6xGx\t j#xQ8[|Y _i[vi\l9yt
@5&P`&ds_nD?B@?/hxsl[<4Q4jJd&>5eia39|7Yx6OeA52b9r
R^Myx0CR*N"x[><UJ*p:~jJt4#J1r7"A>ii/qNy'EL=Me`1/()X%p{

nKmq;r/A))0u!,m3N&4..$rPSX
u!A!p Fp!)36cd_4Gb0WdT<=@+O(iVh7Xzl$Y8! L'yW&%~:&:D e^!jeEeRRJ-
4y(8Wt)WV\5;Vet-qo,-HsGY0RJ@(ZyB@)\AhV'3?Qs\zl<0woj\h[rB =_CabwO4L!(Ev%o/?31O0: .7ea7b"8U!
@h5Vm;wVRa"9)$`;Ofa'#b19Bi`UVWl-$&FB@!pQ B@{mElj{uFo\<Ygl ,=pX,8liB@! U#`{A$6EeKHGNjrC
$(:B@! >hd,z|ly?B@! B@@ .eS%A! B@!P_hD! B@;K@;[JB@! B
Dp4I"B@! % - ! B@!P"T$B@! BB@! \M! B@!pg rgyKiB@! B@T.U&IB@! Dp4! B@! @@*@$B@!
B@Y"YRB@! U KI! B@! ,\,o)M! B@*
$B@! w.w&B@! @RhD! B@;K@[R
vKB@! Bl'|0,
&rtvDM7pDf ?Og"Gf-3W! B@!CFhQZddAzu@B TN_CMOWcGQP! B@! BZ?&l NMdfQ ,gEo.B@! BBfAF*%K-
|ujeLB@! BBfzY\c>|>r9
/9y%e#'|_T-LE! B@*uUY{9rBx+X~z=:5cjnrs)$1+E! B@<.[XdJ:#v5 nPr h-Ph~7?v&s"2B@! {l#MhP%5(*i *)YZ
rH! B@:.qIa9gs(KYf1{X3B@! B@6\-2xf|aFow&CSAtS(,}R! B@!%`~"SnMxvKAzn>~I_|bIy#A!
B@!PmzXdRZ}Xb.%^}(%hPo
+Z%+q\! B@! l'%pGG~xpm98}QbF>1HQ+7u[!L&j]]! lCZl{8! -i#eFu
.\K^m"0_&D/oYf6;OAYrFdeexNB@p,xzzvWSAASB@*~ska7<5=<
IpqSfz.,w&=?A[jpvvF```[+B@i4VBKzz:AX B@!`;AyW-g.~d6my{y
Yy.,e!"NNN]]V3B q<q`-/%Lv$uoV! .Sqqqm^JCzV.Mh^-[cxtAnB@TDL|K[M_UE !
pzAzn?,XhQ_]}^Wun 5B@j-J^[%! B@!P55.B@!p\u@~:JB@ TG+KB@12]! JWB@! w!\"UB@! B$\JoB@!
B@Dp/TI! B@Dp)C ! B@!p.(R%! B@! J$&B@!]H@HB@! (I@<B@! w!\"UB@! B$\JoB@! B@waJBz
MB{8;;A+{18jJH"! &4tNNnO7wggWnV|uf &6Y|,3s8[CD<6i,z_<eAF7}GA^zM\M!P@vjN;p9$&r}78t .^Xv3
8w;g."PvJ5f]sBalR%SwE?g<fBO
@y1K "O\{f,x!t
o/UfW;yv^MJ4xV)x\O;*v&2 LUr
3g}|5vKk.B0b70t(5CF=0p-x+#5q iT_(@dT lPP?}W~1bFaiy;1~`20t}0T*
Ic*RUN/[/~u6u8sRS*< .fb*zIw9.99zQ}c{{-4)xx&x9r|6;`jp/<7?zw! V/?zC0g$8ZQ1Mde<[dWfzjs]W<xPA};ao
>[E?<8fxnh.^g^~U5>ZK&*we6~La03xu[G[R:haWd2`Y{)Ec4tQ^_r*#;7>p DdfI#D,Oo?ie)& ?_G
\=|PDektptrsq<q
p ;QSN ?'#W <=X@&I)0Q=]<PTfgf!d*TX ?yp'6'5Y|<J]W'E K1s5j w@jRrziQVxZwSfdW-|HLH@nNnS/$v8SE"
BcAanI}n`DKHC
gKwv8C$H+knv2o3BuBZiW&#O59!&>&;q5v!<Hs'G{GxzWr-:87T'|'[
CmrsJ=#Q'6/-X<v5z7Of2KIH1n>T"{IKN!2r4gQ{F7=hp(Knn

H_R4,Y:4@Ec^t]:SKXmi
ZGW8PAj]{w5vUOFwbUJ7nJ{c i@ux A^W<4]D$+Fg^rRm0e8vAn?6a=4#NNh-$ZTX:=a={ ,,y
D)gO@JGAKfcIGp9)KH(]-
QE3eK<P]~}1L8uY_"mE_td]`bS1}n:~|u=zR{x/d&7gHq4t8`Ud>Pt17k1bm/O(yvtCH^xhtdqh?=wG;,.D'B@TL9<;V
2^Gzp:L:M<|fUg_O[h<v^)]13qQ<2aO\F}]+?p?_^Tnr>v_MzcEBGhjhI!"*q1 9~"p[cPhM~rBxxP{cV"J
G/+Bo.TLLIxn0/PthE1cr1dDb H!S=g\<}@m{qh>x)/?y#AJc1qo4E5V<Z70=+9W?)Z
)B?U}06ny!2G,x>?^|ql _3QOZr.&7EsriObu_o^4,:n1'NBC~Q>&'&9!]CcX8L&-@,[+ZsA0k~l[?Xt:h|x5
xt?K7_8EG}gIp5+dnguH:%50Z7x*V:}Nj.dJ* g5p mG{m~Zyb!R{I|Zu)#Px]OLQniIWOFZ97uj2+-
4'&<)F^{!?,[6nv!@9/q'of~!5#B[?gQ` a,tjhgS9W-C}4fu[oV0g-!"FC:AmxL}yw30uT?~Bl~.v,
8s1jk&R\2KAKli U{9uN=QN]0)G?SM-/-iSba>t5=h.9$${57szk
vW,UyDVlc08cX7hRFz$"_XhwGpa%:+vo7MCM{{;E@f&rI[;{W3:{`O`ehWSGnVY@z=nUR&!LcFG<`w1`[q`?0
ub,\Gs~n-`"go-LBqcSp4o
g<.-02O9e7h@0b':
ag`E`xVZ>>M3GYA9v\MB@ y 'gr7?/<*};6|p>|nj 5d"W:u}uQbBQ:}EU@5N-u&8+S'b
prJ,[>BEkOxX7w>5 |p*?z9i(ga/w'*URI$Ca? c6w! n>X34!c;pvxx-.M_jR1f5g1n9~:EdJX]fyb`e#{#s(7tM_FL:kC

[~^vB5id>ki/ ecR |C8w!=<l"GLb
mo_{/q`.GGN|;1vk.~x@2QJwqMXeLt~\-'O*qb?q%!!-vc!0gfcL}0ctEYD'%4Zcyhdx'O|Bz_4-fLqch?!]7hn\:^
6/}[ER+LBOBS [.
#"ND(l\z-|f,)A 75v(\y
u)\' o}n?L|Lwm:_'
Cn&B4m@={AfzLc#+BG&?<}.oTqk.|Zn6<ty7Zi[5-LF=WZLZ?O
bbN65"1;wxj:Vlk]r^Ws &-G>G\$l{K(|2UhckJBr~}Kt"FNz;.[
lO' "27E|r8_
O {@`E
|cr+Gn(rhk5Wa&$\7%#iAa]'z}iIM~'i3Td(0cmEh,=0e#hZeHhJJ+V^u(B{v@^J$n7-
}jaJdN_l$wVL;70Q3z9!>.!L~2n+Mz=K@8p\-a5iz5p1~Ksr$m_Yc|=x4:p%@Jf]/PdYz |{0LH$'*m9voW6;
o?-SG&m?>4C`U~zk5JjJUiW-nr*#
G@^#gGmo1^[GO[h}"iosv8vL83}zBSq)Y5C[&g.F>!PE
?CXFoY__'ugK3mCry#S6snKu]u.Y\&eCsrSj&{W91j"7<t-]A~mwL./FF8:9/?qV:g'j>lY1M
S=H`L 0uL4Z8g -;*Zwa}8O$
cdNCuK,YhamK&,[a5GXhQXvLi73ih20<?TQQPLd2?G
R)lD$&tK0H+Md\@k^WzGkQT ^gAc&a45n{6.+o ~AkkI:UfmV]K/KG"SlPdYMs~b+LCE-u+.V
-
J.=J87M2%9\>=EDiR;YR^>Xz6,[\#a_7o>f^E9,p:Y#Zub@1ws0O7E2m^6+o<Jw'd?Ww}zxb:{7GmH.]n0P'0P9XB"
E,x[$e249/>qEp1/=e:bhn;$7_[:x{%$8ca>h^6N.ULO87h$;,fb}@ nL&h_G7V&5j{ad %\?z)7j0l^I'h!2 'DZk
Kt }:Q)H*bS"o2b~CZ>>9MssQHE~p7)z}=&qa$&s0t]RS*_xp]RhJrH:l<mii^';3B$R/6:yijGtSb9ct?K[Zp6I*y
S){x+Y1?Q<.aXLFmgISxL9J! n}'MDirZxnOg#Ji(3SJbZ>E
0"zve*-{4&G&{K}``Q<MZgwmFbqG_1E8L@[=~3dBB?V~ZIg1t}K9jz62;>YMk"E
Z8,Qd/T]yU3_{ar'taL1__5.Yvo5esOWmQ8~5^z$8wGn@.Pdc;Wf"Q(|+z0[Pihbrf/~X8E3Dq%~Ph?tehg>AT\zflm4
ZKYyAQ1/#p-134o
WD:Rg9|QdPQ:cu-J(7Z5l#NB&r<}5~y542c";
i5+V"BNm^%zKQa6SGvG49BduS4g}h@/__H2`+q[
vfmHHl{luA#\R#m]"\.*'SEi @% %SwI\,l=/._79HRO&yl>U0"*+4z|~>kvJ$Ur2DC/3&Cn@Qhx%bSBr<4B>8N8By
3iwhSXF_E/W^vpg}1vQ42Y8Dw>|~Z,,eG)4l+eTGqMAkgiU(H%"ZwG)i/SIpn lb&cpK)<=mH@+o~\Yq}\8{@#{!
L\0qxrh6zc_cg9uU'/#UeV0d4

&j|}qL8M~7K>GI[bgq%_Vs?zhzx'qo?m[;@\R`r?KtCGuo'c`xq`0=`)e3M12kVh0cP!a`r-l
v/: ;y/ahX|6 Ql^M{
;i.jZp+kho_{<Z^8{}7OIs/cJ7v8sm"! H@?]B^Jv?4h2j4C=&U^!$l"_]v@=8Il
w
j_C5wMkH#XYy.~k%hQ 4(p/L
? jvBFlI1j$k2O/g.e2V#)5=KWawQoCqwww)-ZCqww#gv.Ke'3s%vk]?{sMJQ
x\?-[zB3Fj'ePR<rCV~}YGZxts(HcX\I}j~)KPE'0o`}B=gwiGhuH@L171"CQg1"c#(*G_i V<)6MN)(
CNi1{9j,zwl+8!xM\(LqeEWx-^y@#f]p+-v@&F3/lN^OTjKfJSQRYfDvSyylJi5`Hx<{X'7h#l
I
o~
=M{6yCW8eG^x>cw04wD.C1HovCeae/7zw3fMBhX4<_,S75]'B;Y%'{F{i*g, J74j4WC6-y:!{Etj
4w.(CJy4]yO^qh/~d>c'b0nJtgA`N6x!'Qik4S:oshM"np[[oR3HP`
1FL<%,]y5!7^nM(3y@xli4 Dy41p$SC"'S]`<};ZNdR0y$Esfv_I!ar:Y) 20 (OA1foz(]
">? !kA|=Y?omydyy?ptf)1/#mdIg0[<xc!xs8UAY{c1c&!g9t/\ ?|?r0#y13Zj8|ypH.CG\crh"iIS)VA]$KN]pudqq9q+
J~nB{tFE<=SGwh*YIA`h'Hq[a{d/TkjoNMyvTR
*68JU?!jy;;jRVy{,j,&\ sj49L8V96MdHNh;[H'amb|F0,eXD$x%]2),()%b#yXS2%_c uvu3 \v6io$K?#hok#UI]
"%=y<'=HM-x E

$C^,@"&<Z2fK@'X?q%8,p5216,I37Y3S7A-
"FLwh$HsfLRta'h2*=.s%Km[R!HMQBIvMQaFXYP%HD`P$m`*.D(`SF|sh zh@ITl`NzN6vTZ]
K]3UW[{Es3GKkVJFG?G}6>M$Shfms783ZjA.FwviQBEcciBv,l_!ms3bNs_V6!iL"E3$&? H2%
Sg!p)&JD{Kso=`MEP@J9dDN 6Fa.=V~002TR*~[S/Db1@IDATY&I0B6<-^5$8$mtJf%R/VC054
Ki8{JV}%!I9?_Q`&$0ft|*q'$|D$s(/IVv_"%) `bjN39IIuE!J:8,R nvMYWJIj}2Jr"B&+4p_Rt*BO]8Sb#+
u
lt,zh3-=$RKYvI{7 NB{tbud(?H/GK9yEBFIc(P$^R?R:{{UOC(P @@ XrW/g>m>9a(A0.,dF^8fg"]+%;/
BGzaO1*ZRnaN)RLQF>m>)UI
(P @C W/h2kJc}=IiII0%~aZEuUo>0!9 2fI(P @@ Lm
y2gUJ~|]ES:aCzTC8xsJ%@&_KV(P @@?=I/J$A3.Iu6=[13Z870.H@@
(P @ge\Xr9[<~Gs1+P @@
(Co^rMY,dSf_wJm
(P @@
}!=HlOzm$
(P @@
B C76tvz(P @@
(Ac\uR:@@
(P @@?q @@
(P @@C@a\TJF
(P @@
)(?y]
(P @@

R2*P @@
(P OA@a\! bxvswbov'DGF *:_ZlT"DDD4Ymj\?/A5HS-
P'S='FFXX?6_:"qRH><{aOChHkMsCh?'n=%5=bxm8TrY#-
h\;/?|9_.X89*/[}SmUE{p\tY\wp~h\u0cLl8x{Y]Bb}cur;smI\"o\&c^>D.P~@m\\ `~plcX8ezj>^BXwFqx5B#u:,gJ
B?]`,Cl<?EPXa
xxTxt$v@
aA2(1 ,)'"!.vy`'qpx#+^_DHj$/Ea/n<>eN66csDREW#bBCQPs$Czb4TU,/,Ax?8/x;=35/+_NETsqPj% 44&07CsX-
,U`o=}JT&&0wx28a4A"ix!,n0,4041

9\V"^T``o}H&8##os1eJ)7Hc=ILNu${FtHX8L`bz~IP|-HaBL5sdD(`ec*pH o
"Kw>
P9-aa.y
L=G47kK-"yY#"<L%'M &[4J,Ja4 3T`[sDa4xq|q1>j=Mc
$1`

b\>R1&VVLSFNRtZXnW~%"&*:. w QhpuE>}& Y%Hb|,HZX8|1\oD&vuR{OcS0N-Fw,E
oh"BCBLafK,*&= 8"F82WGDp\IdXv/*hK3J4 &aU^~5$[QJ\'M\H/[whUK="2
F&(^0UDl]K^\q?K^01OZG 4
0D,[^qC

`8R0hrOKI10'`|3=@i)seKDDQ.<:S~_X|b|B#/XCMI?RE#Ok2\DGro1jlj*"mVL7Gb/
-=[k^Vc L7f
f1<zKw`U>1_byq*,C,#N
Yw@S@+XOx}"c0%eJ{!H?mB %CLqr:t!(_8TKm
!E1` (ocEMQT)^

ocAWA>2}r&?< 'q']bl0@~:O$Xw;&M gi#!?d.]GdZ#a`b&CR9sfNs[qtMI?)>of- 60"ObIUA2DzJ)Ck<a2s
X3q"Wu+.>-DQ<Xz=^4D&1KsfS-K129
1&ss4^'jjMo_`BS,\^MMFZ6&DbxKG\P&PT=Qh9Yk{9kO!gv':rpnzxh_p'H|VGL2_?=x`4(S&<w5A>2R%c(ZUq/L
vU=\;y0~rOZoIy>b}2$J2}A>_&N;p?L?*cd{z3BSUGbi2 $Da1kLu
Fbv5c1rGoL@4^LS
&v]{IMwMgwKc~l
6@r=/nE(L]7o\C#Iqu6AR}{vXzvx2m/C*:gjQr.De
.C]H\nbdmLO_JB"aI!?qE$ 8=C:6+1n<csQ:JL9tQ/7A0b@ry|.G
MQeghYJ*];/!+w5 .3Rn,-j#s,^kW5h3E,c3
{';^\[P^
X} qe&Vp"8Mmi:a
K.2J=! {gl2D +Xoyr/HDE0iT|q>.Ko +`/-P
-_gQ85GPL$rrl9tBF|2>:Y[<'-/
yq>#L P3k"lE+7pJ8y-aznaX
XK }F="|=b=TzB~W cdii
1n=j1:+G|nIs%BU\c@!>0h://8cVm;#8pfC~ >["(Y9IXdpeRs3eNg7 c'F8cL@ <H]xoi]7qpC0dPoz^+%IdI=2[\5[ps+0|"
'D@?PKh!'=;j$ E1o<=^>053Rw.WM&5>}k@/htmV/ :hrACF}=wX5?C.qtTy{sf{h(&NQ5/& /)K^ivoG'c=3X2?
B~C07<V"Rm5g}>'>\"1Wp!n6sb}XgBl '+G}h;#UCdwKOsah?
x7r>r1qwN0zLMp6c
@@b:i6&M;HFIGxF+OG>RK6@I4AW0?MV@qd}m(>u=nFhF
h
[2uV}6F`q'(5#|C#Ok!a*BQ8_{CA$z!q".'U3bK'&a7>N[e<9[68 &[agR(e ~3((>|{ja#HT8w*v~FD`QxE!>2s6P5z4
9`8~AkWG8o3;sIF3Dq/>Hu"M&ra6qb8x{}<OX01\wF9pW(2=0_op%``oB>Cw.uH8pj?)X Qy1dYL%zNp=P
0tp4(cw.Cn"oe~Zdvf=okP^[L8 ms&Qs 5v"+FcRBn p#w]/oDs
wzI#pzYrgwGsn &JDN
,MvR0a-U@=yRLi5u7Nr/50fkdjMaxu[*h~t=zc.{3m|{:ypl?
zPtVh"v]x]{aW0a4J7.kl
K\D0n4Q^?i:8b.
bQpxc8S<|.-JKeJe=cB+d(@#^]?YT4Il(ZFzB92 (6+^cG]3+HHx=yqa8e-
:b1A?~
F,>Gai;/<nOuG'qvnkrHvxyq/Tt`QJS;7K
Zc!{L2es j~3??.mY_N?G$\
q}`J9qAd(Qs;cNdTD]F]QZ$-V4KIR"&NQ`-9"9$Xba<BM`x"qa[B*
/bt;:,9
j)bX
T3D$#^s)GJ9e$C*{7>e1@C]J}q':\p~8y"N r)`~4B]D\:fz'rIit{hbiB>k&r9Rgqx1h4 K"uEZ1yC>0[o
Q;=_aBrU E&dBgR&V`
Z?qbtjZZ*8
M{=SBDq!"EVGbCyzwh,jj{F{Swmf4MD46lZ2sA>
/+S{zn?Dsi7<mB<~CW#DI#>}G@-oxv%T|a`hw9*TG[<*g)C>3c;c?=DXx>oEZ).]|jpEd]op7:U7p
FE{6%krL_P0
"`p2-r@"ya4Oeg8~4jlvX#K%R'F)7M&; K453m:klNJ,$w<_j6.<c&l\' P*aut9[WXU[* ;gb-[45
S}h[7=?^~+{r cyj_H+Vf#IK/W.K6No.F4'^_~J(_6Gjauh2QP6~lLX>{&\>>3V-VNe\ K
{u[F5i!>:p?26o(X<\la+*U,j
{LoPHz<iTIk(|"70nI=/!F,;
dmLG{AIj~FP<V`Q6T| A<`!
xa$D'4MJeEF&@mCy^{Mz'L[B#MIn ,e"kW>N^w!X"a&,jMxD+W&RX$qHoa
iQ=NG:23\l7|#Q8]4<B.3Q<U1QgKB%(%}r/[^CM-

xl.wK>;lq+Q=!(.z/eT@}Zb$ht6~fR>+K"w@Ho?}g7Smh`/puM\$5Bx8
bW7n_mMg(!`N<suJ"C^Y:(|uK35}E|>51eE:]:Wb xt4>`2Lh~} YQ<(< "Yli3z!>Qxh
<|$=/vz7mO@(mw"vu<2ftI]="R Ab"qp,\fJo7(Y,VLAtT^C'Q.LsMI7($=DT/"h{|rR "~Z2G]:0O4h}(NFurhI-
d,D~qo%'Nq)LL,Q(&KVl/x/9Wk3Ne2
Mf|{m{/Y)B^"6BHSLhfsp?iz@.r4w8k.il)IOT{x*hX\I=*/G8NjShS!Ga$2>.td*$0Ct7W;?b(C$
u\!K$[!q-zB(}yZOx9%`{t4-fZRp(qv6yAfZDv0|BJQIo@i#J5!{|!k8Rpb
+H[wP7=0rhl>C?x_)N>Ik0ga!jl)ynq21$g+VntArwdjY~"~hs[2; 9uvjEdE]~sHWEi(+q)sr(#@M.T+r^e`tK3[8:Wg9b
K6of$hW#5,P8^H!vHgeE7HCqGI35`'L}<8p?c@  o[:B`mGGgaxlAFyqn swLCGJ=ktIISg,18u0wBYw(0}0
^aL3cdti*Q]<,6H<}[[[Z-.1-r4$mwnS?oqDzp]'^1
+n$C3!>,o)/n
#[V>gO})'[;K:i-fkT<;Af}JpBh^Ud#!M|I`"zPF-
WTEDeDZFIx$;^~bZ\NP`ap=_}&giNdV\S[V^zwN<yVk0*TmjWC.+9^!aMgiLZ)lMR+TB3Gg8iS E-JVtx.<}-
L;qjHBrMSNGC1m
P(;}dTc~uowz0A" |Bnk5u!w!-;dWj\jGd%^z4yK;q?atZHxs06IG"+YL_h7YqLgA,) }N;IL!I9x%8)f}=<~C1VYP&xx
h+CH3eJat8#6}I$N\dqM+jVL_?Nj
$8g8G:;5QA *Vl /EbYi8Dj5S**=tI` "R ." dKH7!  ag&0&H.%[QE;T-QBi^X9b5h|4M!`8Z-
C$S42W5$xG)E>IIO%OagkKaXwo{7!b!3*o[_$tJ[;tAK>Yyd>)1Fo
`%MFYy-aib%Xao\C+Bj?Te*2R TQ+-Zc.\-Bn8}T/pMj%f
akdU{)(@R'yK{,0cGzMD{@[L.w ^uahi
3[-A]PkJurh__]64VzDG5Y:f#IjGnm1>RR@ ol$$3`yd1aY5BD&dD4V_o<zH2,} (,#,)7T,3mK\P7'v4{i[ssP;-Z;R2u

x%k!/i:?F-RM{L82w6hs8fFJB*3YPr@ C>w4$MD38D2AiC9{nn+~d32%LWP$ER22
S#)JU6h++W5~i ~;_8sYEkJLj#"B"&>muw<}M&q;eeS_dCc6y#e~Tm/pK0b$X:Ye{<C$%Rlw3hzy7gRmf-'IT,
Sy/_N4Qe+h.\lRj!{f@?kJ+"4y,|rc:$x|Snz|
T&.2/n>F:[f#CMHau++SC%<#&T`Pd!lY:LM=V?{0ej+`Xe[ZP"*K&[PLAN?\:`~R0B05
_n$*SC4gfI/*mA(xC!&JV%cZ.}n=Z^>8jjc5xPG0O=db2 OI>Vn2Wo
SGT7WE[60}N4!4Sfj~347Km7(sQ.@GNVRR/T*+dzWN .p)P;f4klkdiNj3^<>]vWX 7@9t$CZ||r
/QQ!U=a:0v,{GP-I)+i_-C'2vV!m+0{^IMPBh%:$o6.3xr|opq~mK`w/bf)^3[Yx.go`Y4+J.-_;7i]/!$j'!Q|=w&>o
5PI]x\5GA*;!1t6O_{nzj7'];X
GN_`XwH7H&=svn|md/\9}MuX!t9Q*OFr#7M\RkZm1PNu!<'3l/!X{xX4w=3$CbLwiM<y;WCZx&i_rF;{MD`X'/+
v-sqx>yHuBjK
x_Gx&Nh\TX3b>JU.JeKcpc<}xuND%bpA~6cFBJQR53G;\gn_q8szHTIU{+WbTby|^>#p3X2elA!*vxsn>7LYc~2e
(0?=_;\q 8vg/p
Ks0M4:![M
~l:L
g7cu S:]>z6n?xb8w4Z<pqH
ivis
svC<-wNzx2AJ z"?{{7PTj=x_/~$au}T4
y=\YzDU8{p','MgwQy\O^<>a3x$+cXf=#OOi&iFMkCogu
Kaz9!~(TD%rdJ&x>O0{$M
|dI8m}l}Ir1>lZ=YKl!+C4rEH-2IHIef=F<u|||l6;Jvs
0x]<mJA 'f{obg"B[gIj;<{%eDG~ZB)U2]
oz1(]4'{Gi~~GJ\jz6TrAgR*A`/|#Q`6mfE8{tJ8#Uio:x1hTy&yh\\h%p!jugqhBI7IN?O^!kgCiE3x)5RN6ne:Hs(S8&sR
;_~9Yma/dQ`Ym#}SR-YZ%d:)`o5v0}-T8:h&vuN"z1?GrCLQTn`).9_x:Db>1e<:#L+kVDvZ}\<e~bc3T#c,['0y`)`j
5$VE8st)`h%j/
Vk^-
2DK<%{W|j>RRHH\26cIHg8Rn[jE'O.(SN%"wp,ZP)(in%6OOYWRK,Y6a}5DSo[VD1U`\;NTk>,6B<m,S#R0wK
%4Sf

n]p3Ca<q]&GfMI2
x(PZ6efTPAH[EQ8}
^.2=c!.vSFE>rqVu94
"cf@G>LZtZt!Ng[C}0nZzIR2Q-"[R2+q~s-SGaEqp),3A5pjUWi,h]RYUzebR+J]d1)J}
vA~S8A&i~D"p@\36z9X+KgPj3HTle0o
p{1/X.\5"L=3{WE0o~Jx}_y3lX3^3Ky!&B(CsFTkfXQRT(Y88c~ZO^
-P~XNuO18l)+P.s;fL4]6rp:g3ha,^O_CiqY"R9Kx50S<CWm?
+##A8$3F"e-D hS8j%c89;u9'r3v0@fu1kITA%iT-`L#-
!+LBiNOb;S1VG)De.`lBAUCo//G8.{l^tU%?sJfzg9t%,LUvB+7-m<fq0.N0SmNI
VnZ[.i"0S=[;3&z+U&?'768#o_$Vm+<~&svbR3}YD2G?I1asVk\}Y%5[
[Ei IALiI^}RF=1?
+uN])]P3XF\1n(G'=[o4h<5**w>KF\O/#TAcd?L&cMaeEg+$E17AHUh>(R7$s:FBB"RD5'6]
QC~NEsakAW!I<I8+$ xn>E$ZO5t/Hh"OM
J!7#%uv")4VIuqC7r=0Le(#QS"qC]+!/BVQ6rq,"QjIOt$9#mMzDp&c~C ;1jE#+(c3oik;1ZC~9"8Q8w4Z;#%
'UD]1Ba1YU$0dBx@62S#o=@IDAT)Z]?h,X?T8D44!}2OUxO)><36):~EF^hWKkI; dEKv=KWGaX1d4(NS1w
Ah'*Y,xbY

<`=4W`Q5L_I:Q0Y8(z~pri122j +QaR9>^1Bm~$pnoiS! 0'~F?BK:hk(%!Bh#*,xmJ{(q|t m0
671X{/M(VSK|`s
49'8
s+,l8Rd";"0v>[gRc(%Bk8^)},hhKy;&h iTb^ saB*I4c=Q31jel5$`df`}aV?t*?yr Aqfif9qx<h6wA=.^ws899I{:&
k}io|\+"dDipx:/LdOn2BuQBw5PE-q+WH5@:cytZ.kgT]a&K('=I@FLHPOjI01-
jcDK(2D%5PK}59]U75YH8j'5QW.U?}MA*aN1-x!u!+uRJVBk<LvUk8Z]?<"<n2 `dkS<63z6"s#iQ'A Gwr
I}M1$}?('
=sqnT=bS&KgdzxO*XJj!D:]%0* tg3ufHz4Xx:IN8tqG;0k\%GRC~$'O6&9y-
Gi%w+~LO}ZT.+_!DHzQC"rAASeZsT\KICr[/\?uF.V\3) Ojl.M"6')>kiIRsZb"
M8{Wyp+T;XL616Kw?'HiGrEe`:lm\Mg{(|s8OtuX4>(:TCAtFP[/S}TK2UX>z$ntVI
>o0cH1dk>U\yCd(ult|r!:/_E~e2_6aZ#cRgkFMIR^JSL^\3"K(<h|UxbsfdM/(@&Sq7<U
0ssYrJA!g\|9nD3YO"2e_>7q!#bGR93'Gnsa<U?H9J$a%)Z0ClBZf_b>/T3
(yar`Q K<Gk6D[;KsX3vk40O
;w@LnO;r@@
(P @@Ooq##03SV9Q]3FQ.ovK6SgV(P @@
(!m7$2
a4JKo>CQH@@
(P @@Oo!)_(\ E ^x<'o")(P @@
6`Z[SW
;/;t+_S[
~6Dhi@Z=<t3S4GXOPRgKx-A-a*Iui*jR_|aP9?/Gh3o=1[qU)
(P @_
&b>A.=-1go<)~Ns
G{RXz/>wwbV5_r<w"n}
?(5~
lufZzt(j-Z`i<6GMysLY}k<*6lB(B
 {a+""E{-!'~gfMh/;;w;g5o+-nC*aye,^HRsWXda:%!
R]0eKB{Du}m[4Xa;#,yyH0K@,.rhk1DnXF!'M"]Yx8ko6DYyy0e$VGvD/}4KFwO@nN!rn3?D-
P`>|/3"Q\<F5;%R(C=J%|h$@$P X
r.Ln:7idof:dU!5-
iW*Qq74+M-O

!X
|KG[qPYd5[P"s>f|wy9P=7'92T>B^s0
l?Dza
,.wA=r)ce5{'>MrU>/X!_X$OcP:l
PiT0}"<,RY}%ucKGZfukMT(wM:_Vr(9;O+{Yxe=*2E~UV];Z\q*S<e?j*W3XN?>3r2)VSjm+28x^cKfZj
r|s}\Ch|2D+3/Fz.
UF\o=fMG\9'3i%gS]"7QnU`A)-/2dbUH 4/P:nu-?*(%^
eap_/}-Wi~.oSn
f25?}}'1ru!bcs:"7\l;9=Awhme??z6AJt8MAn#ye#&>
;5kb`[j(gNyCqx-8UhOba~jHO/Ek@)$1ni|3k5n
q<^rjGcwhqmT.E,Eyl;~5{/Z0gA+:M^?cVc*o?Z8EVqqv=HKL!>z +bmg-zJ6'qWqa-FD0g!CWl-
'E"N=E6/DA9,zZ6WM;c#IlgLND*?}_-\CO]H#^v3Q4)?kO?#0o2s62&vn5e(yq_*Fd!x`KHcw08y2lo2y}Z'p4e
hYojIvQkQk.DJz^v4ni9oz
oBDwNCLu/-u?'01yT2*F}}pUX|:$ec   B@yAa x\WW}{OWO'=SsTl8,{36cH+h5;V'Lh~]crxgF>SOax
#.BH_SepMx7?$Ft<hwd5Q3`qp6}iXbirC^^m0uSH!A%`bTlIOX86.,a4-tY;qw+_q6:F3F%cNxmD,N&mw
W6Q~2"?G3D_0~3dsr+,@=Qf7Fg60mDGt$n
);?";zT$SC~BJz^o8ky+<^ $j
Rs7_v#z;c^M 5!\0XW?;oMs~-N)RG-zZDoKp7`KY6cw
=4.hO^Gl~%0v8Z3c371xGy.X[7Ae{T5vs{pkzD?\=w^|c:{xw.]3BYl_3wurbwl'\Qs*m0{1q{|t(,---;O"
G_wxD;p(1JRU<8w~x
FT
qp0pw-Wq{tE2d,^}>s$
RO7;'hYlfYOx#= Es=kMkXZ>O_W:*| 4
+v\q5"ioK-ogNDX/IV=g@Q :cD1{^QhM40%G;:aepmZK8^0bTX7 q8]n
K]~V.^anNLso^`mc@nX]b7ZM'J~?!EsZZB
=j42cj"F)O)]l;e7|umg7!pCwh,:hqE+E`"V~xk['4TMPh&=`n/sC;7/VCVz-~e%n^$=
V4C=/2o?Gkq4#?vC?WZi[Oo?C4o];5=?W_4!`~q
<$< >B'o/<9R;zf]G
8Y^6?D%Pi9oHHf`O_a-yj[qG%3O/nndBFw<^lbB+-*'S6e3>}?YU<?_4oR*@zHaM2*f`4vL"s)lgRZT"-
G:wYAm=]2KN[?N\3OYunbA=zKI
5eMUBm}uRX<<h {p+A'ArH<[QT(7)1
m{}%\%I,8omb
+L*HRf?C=X!;_uf1EI7#cqXcp>:0&(ZEkGTH A^~Z$V;ekQ#.ty5E0d&2@'0(-;n/s$>3-iDLTzim8
^>]T./O6tEQSX@eNK,_?{
~;Ir2,MGa@U"{ #QVDKF; @
#POJn6m[VQQ3q/I*Y=TuW%F-i
(]<@2V#Z!lFGZ&VK3(n,,$Z@\|1;f^+O;-NqJ~O^ %(DiP8|VQMe(8jGxjv"0~Za##MtY.j>C:pbsC
0"K+D9E {[>/v0EP;fYT|;|yvN
[t`
({e>R_2/,3"E7
(-_DAdOZ+
:QA
0i]fr#H_~T~=Q(.\0u@@n41   @3U56&H[*D O6VtkkN;Q@,,>yp,
zKr3_uaSiW\"C=yPJ~92R`$8=\b}$@2Q.&21@36l*/2RQ\lU3 $$ .5KrW_^$
'\~M{5
o.jR;anR77>X:DTzOy<b*DJ/x7mLk=k8r_9
]<#[32E2E`xX|%nhpz2*wJ,r(H gn~+6ge-*>P;.\|x5lgS_wAHDc"*<7QeCK_qys;Vu:j";'# jJ@/#X*
R {!7W"q*IP\eweGH_2g{vB7O%Ybjv-^WG_~f0iKx~0F

9a}3!Fm5'?~-euu5E\1zXL}5<CE)Iu
).Bgg(l*O$'%wwzmlUe#.[wah~q/~=:QfIcS.`t7(c,+MV1/vju8,2i'<EY[-' _g1Cz9nb%o&-&vj+
}JF%cpxs0Q2S9,OG04&BBSa3bPX
6_|.V>#/o;<J}1icqbh
R;3."b0.-C1;o
:S7e7zDKI~3KMjC;5
lo>v,zneUx\{$J8][zw?6Qj}\9;' jK@I<_iS7$s\Z`
 em#jLDtx+W3eR(>{6HHc'eyU` X,a,KFF_;K+x5{US/n[#&OZk%A/{DD-1o~5-ViQnTWDd9b}`\{>^
y` 0>xlD[<`7F~$fNJ]G\?Edh0uRnrG@(^MulCzxZ&~ImD*eRsDHt]&JN8_ HK]F<<6DWi.M^QU?
oJSmQOV&{f@s,
v{q;qIdl/cZ7BaW,'eX>^qxe.^/.5RmI%."2=N++!-lOH-
]sF}o>=]Mz=o)RM1_W<R~dUi7~
xiS;w_`)Xxxn3D#>L|@gxwlxXg;K}{z2)CK<Q<E3&xU Q!HH
7,]^e,sem\"eU^0p)!YmA][7Yd$@$@$F
> LV$SzO6kA9N_B1W&M7rQEU@B/'{_w5(%$ctU(,uD_vMBY
7Wbj)d"2%^OuUYP6
u"/3/~R|]g
:q29WArE_*Lfax3+J6$*tmHeg)NoY9t-Q/'TKSPbj_J/O^r,%C,
bS%xxDQ^BTo2i6Y,7x*Uw*)B)qJ9QAYec3@U_l)7
L;,Z2y>@uXh*_PF^2E%}M#_)%KdwLW}C=h3T324W1xS zSm]9'e%}!44TKTU   +X 3ioWZgU6#g
^ec(o)Kb.Eo|K?TX^^:+ o@]Mc[^e;Jg((,QiQ/kr m&QJ,h{:-H)eQD)wJW@|JK~K]m7 _>Fb4t8sz-R<+_]~~+6kn b5"
TT\jj$@$@$@$@$PPqFEQIHHHHRSk&   jDK5,J$@$@$@$@$@5Z,7   T#T\QeQT   gIHHHHR*
@M%@<M$@$@$@$@jTYHHHHj**.5Yn   FT$@$@$@$@$PS Pq5r   @5"@UE%   JpA
nEzG]W$&nlpJ$@$@$@$@$@+.fzCtGZn6>My1
hY!!3&^g2HHHHH:EX1g;Lr D-1<}<\A/]'OOK 5   TRX{.'^96
' 37A>%"$kY#WJSUe#R{)2@b2@$@
,8rGg?5i:rryFqa*(-vF^SZReBQbqx%h?$p^HldyU;yO]\JMwZvs(FmG1s.$<@guG5G{ E^/P2_w5=;R>>\+a{:Y;4
a:, @E!iq:iRR$ EG-xzp[lZFCY%S99p~\J|sm/4
LEBj(wObbeGicGOdtz~^_CE/#R<eQkl'6 ;#
C@|exuY^<z>-p*+otF%*NGNzIwhn:7Y9D?#jXKByHE=Hj$=WMjWB_%P*G'+\q[TG-{0^\6+*?S)\YF-
5%R]]I&IEkqH* =`^+$U/)Hj~bA|FMRTLB
#0]6>1|*IJq)B$PVkLr2i[5(@$P3 V^
TWMf^*?jvL/pE[1i`7]
>^hoB
q_2jD$@z6EEJC@i VQ>f|*X ?ceu=M%upL
~Z8n@Y$@$@$@$@$@LqQ=OO=^X($9AcEbXZ}sdJHHHHH@).*SQ^D]aSkq,9MKCrq !.be
#U\TqXO&;adY!NW`C
IHHHHH,YUlHK/PXR* R?   T$pauQ^Rs0~g{+'^-)V:(R$@$@$@$@$@gpuI^-cDyqf{l~*-
g~q)ZsYUfs~>RsduK$@$@$@U\Tdr,\%Y%Pi8U7CNO=?EV~V+$S'psH?V"@isHH!P
&`%S"R#pEL&k0(s6'*Qa+d[*|WkG
UIGB:1HH xUZ]i"3^=Z) 7o/EU:LDJ(.*#N@r ABQ1FPhFJJdek^p:)y0AY'-
yYf[fIXdR`vHJADkJ'hK>HO419S]VYrb\
u,WLA$@.A /=73WS(kkY[Y='?Vg
y9h&$NGcYsV4{FI'P!_2$#Ld3)>6]p8-Z&z$>JHyq8x2FQr=1_,q)~GtzxIOlPe*06^ZzB|dE6}IZ+vln5bhEMi&.O
X08]HHBEnGGiIq~4
RlWhrdeduKKA5 \t6E j$"91Ewxicu4Ao2&ZKWlQ=a4n0Q}.DVzm0 #}}v;WDV=%i[:I
U/_OVS57iY#oIcce; *..T:X"=rwjbciVLxM<q_N=;r3[AH&~1
x8<[FEC<1<E,In00Civ~RVND_|F|u5CLU^\^yRE_4X/Z&^zp)

!A9>[K,x^h(O#,Gu0y><6=F6kVnPQi pA9PKCphvkL-:G`~-aln;7<Rcto" --.i7l["-
aX[&}axl|?n:gk/?MWo@HDSLKSaXs-HCfn<_r+BB?Q$foqONAdalxb.h!i1\G+
ExscNs:oQ/>0"&\xo:zS)vALF7pq{wGwh5'
o;,S~a48\/Bxm-1ya0
ddQU&Ea.>Yq}[Daz&K0gqQ\H\@[~#?n.RV\;2]6F@Y>)1~,@#'VoY1$!=OC
qS(NiU+;EbkqG~nD%N]1`Mb:3 \X:IKaP:^G$z^PY"1cU36cN:h+O$[2R]'.U_47=;aKfI.ZDmWXt(.n-
!N2J>??G{<z,dXODMX!+ahnCcP'SZ67AkGLVC~<$
"PAB/^([s_zK?FGh.ng{n+NM/>i+aa,@2CQ@bc"+@DAXwCbV^Zf5OdnfT9C)@Nf]:i#V3U:wyp)EC!R?0Oz'ly
an9Z=bE
._?0%K\$eF5
dQeOdnX;DN*UTP+"?M
xkzp[pM~u=rEI6< "[cYX8>Z~2{
aV<gQ#2xO/F2/7:Rd~PJ:I\xU!:CwVZ*n]DBs~]LwQvEC%q:XILw$vD@j{*{UVV
se@I,i<=q*% 2ro=
0dwumJIPc%AS'{FM()g-q,f`FGbPwYIL~[zm |-O/l,O.=G$@$}1 F,28s(.yE6
AP_$<?Kok@O-iq*GNWaD<@PAd""@t9f}?X6D,2o3/]6sT&o7~ww:~JV<3|MVjd?nI/2mHSh2DIy[Glz"-
LQj:A+B0m"`r`
Y]%zCBb2$f nhP(`qdY]FFuK1xg|je&<zC0gfa0,& JC>9?XG|I-debLDB/VI2e71`Rli\=k~q &
nWdl5w59;K~68,XUZYJ,g
jHKxrlyH^2)J>/9Hs+2\g]CR+Y KYQE[SEb|\QHg`k;(}h"x
2}mu2oF4GC R6e^ZI~6j zbx`AJ}IDAT&at@[
L;,1AY7/kwcvy4e,<^#~3yU]p])Faw%*kQC:xqXN[a' ~>fr$@$*~zN*D']3dDxtr:fe&m.z$&}BWTRb}L9)xt"4-
&GZ(4Y%hP}#aZ5Uy[}A|.Cpw4\FDl@o<hl>yfm$$Mf1Zch!~fd|>6YV25e=m#a*]`h8><fG#>2&
+7q;|oFV"IG7MD>qr&Y|?4l>J#2oIt^\jV]:C<s+$@$"j1)M"nZDxkn|_e28%m}'l(>jd?Vr0us*DE6[=Ox|<KYs<B"1<
7. m
v"=21! ~_[F
?z5t(,<PK6 *+*6t*c%qxAM)DmnV1L}P
l,(eO9;&rrCS=8HH k{BpQVc`]v7Se KFd.if%G{9f]1XH~J)ixoL,AY|4 o\S:YM
`XAXY|`A
0\?*\FF
B.H@{R$! =6hTa \<S0gs8/H 2G *&@2; '@2G *&@2; 'p&TP3JD|%
REqX63'3rHHHHH*A(.h&6XZ6a5;q_U/bUZN%g g~(*!~v!n!JSZCE :6N <"@UI@Q:ZKKU
!N@_<OyBWEv(_B|wGS0e~&<+'Q&9m-/%xpTP#R3.K #_|m.[
F$p#2M<_=O$PYUD{~fP)kt*C-]cNlY+wgPZ^=hPoWr \`QZ]1"P$oAeJNzTqE(g$e dm$p|k&5e)=gp>3rRt^IiI|
HCoEAZu}i=_\j/2ybF.f~]#aS7C-_E:?EI:AFa;elHkC8?A5] ly
FGyd}P GJ[Y2\=_T-LEU:E&C#oVbLhMO
mZMLB_4)FS;>+u7
`X8\yyjYu/MZ[*E&MYx +$koiM&5RU"+,sVnwuKJ[`yDiPwome;a-\B&ucxpV%rIN< #2
3Ev`ZP W]krP.u:=+#B $;`Ln.%Kji$nRViQnRf.Q\T`LCq5;&ncnQU\zFdFR-=C/DN,HfPp3hndU^jJ?lU*uSE
%A-SF2$PUN=T#@$@$pIf0 Kt2Q6u^X
<P[(imR&6
"saee1uoV/ 87{pHLtDrE1ty{]'2.(<O.a(&%lx@$@$@$@$@$ OBJbR,ugHr&R?>chgSV<
(Gj\3IeEWZ#ZEP@V'c TqQ+751d:oy.yT=x6,^w P*FY&:|6z2[v,DZVc3>p
kJ(N$@$@$@$@$P T"scJ6|rlA<1hJ>IyXw<O,TbU2 U/n`j-u5^6|ow#;<hgJbNQ$@$@$@$@$@@+.*W
D9olq0N$@$@$@$@$@.pP)'v-=gHH.,"m/^`[]MHH.V+/GSA6LHKKUwf pRRR/Ba_$@$@
RmK_pq,.B];?~0f$@5z#++r!Fb@??(jeIj0Tf\{Nn05:l(HH ZXhR]WRHH,U!$HyrYpww4E5A6FT
%KHvV=B].l/mK$P8j J.8rHCpSq*rR^u
HH pIT\@?xU|9(8G6>tF$@$P)6]@++ jB0W)& @1.[HHHHH VBZyLL$@$@$@$@&.VX 8*(#;)/81VY

y*V%& jG]Q` y:gIHHHHR @#@9KL$@$@$@$@jWeHHHHj*.5Yb vNJNJb p
KRq1Lp3!$@$@$@$@$@;4I88 "@ 8#@
..UHHHHH.0HHHHHPqq0$@$@$@$@$@PqqFq$@$@$@$@$@.EKU! pF3*# p)T\\:(3T\Qa
KRAaHHHHH
HHHHH\
C$@$@$@$@gTG@$@$@$@$RTuP g88 "@ 8#@
..UHHHHH.0HHHHHPqq0$@$@$@$@$@PqqFq$@$@$@$@$@.EKU! pF3*# p)T\\:(3T\Qa
KRAaHHHHH
HHHHH\
C$@$@$@$@gTG@$@$@$@$RTuP g88 "@ 8#@
..UHHHHH.0HHHHHPqq0$@$@$@$@$@?.{IENDB`
PNG

IHDR-N<sRGB@IDATx`TU Pz
"X7mWw-Zu]ZV**CB*!:sd `dfox{k)),]s!>pt ,xa~Zw' @ff&@l7 tr-: bHHHHHHHl%@l=7 tr-:
bHHHHHHHl%@l=7 tr-: bHHHHHHHl%@l=7 tr-: bHHHHHHHl%@l=w
i*~3?:a[XTzq
CYa J^%%6s2s9d
c,nzkuO*%h8N-XSzpC=eR(-*CR7 8'9P-/B-uf_ha?7d"W~6T5Q-pELd?4UcB,[KlByc`CR&j,X{EKLo-
CGOazo^Kvm9BLxs vTLLk>}gN#1tK}
v9]IHHNEg#ybA
v`u%xogARqfg_g;sO,\7gOO[7M=Npu?gIgQc7Q^pwv8v#;^g":;tJa8t)Wzk 7Ln>r.~$N(>
)bT[Nv t.9l
 zp@`oq){sFl4y>KfM7
8{#&..<$gerDymRR.iBKJtOvTO=+7_*y!
U6j xui`\ub)OW"*
_n=aF+u X%}qrcaM}(.' 6Ar
6%$"Q^fk
2=|]
;vR)z%4Q0u`xS-}HCr$0q29X;
{^W O6jt"aI{#,ExeF{e\T Mp{<VB#bKAnBZ#,k{SRD,GU_.*(k{Z\,bzu{%rs
ClzG_lErup2PWup!T#CvBy5^8o`~q(.]q9?7Lr@{vIg2
aDqL4Y+`~nR1r~fszj$Ag LR1wj^y\xO~i>! *Yj,DzVctNFNe="q~.u{Lf\0.xX!wpvw}qno:fHtpBt-pc&KHL
!#qY~7>>z@Z*x/FqmX=bw"54n[0cD? [JX</jFy~`;HHHHF<u8*d,;z]i/
e{-
E0=Pfnf.xU;"B_}eTZS$<<'xqdTS+V`|Pwbun5ePY_{!9v Pnz#l3*\a0Ip@V~.pV:+rq\{bEr:&yLj~gp(?}IbY
5UJ5k3}"MNO
^7U/UIH8+M:9RaRX%mzPJca{Tkj7+0X68/^*7WtE6:&o:D(iH[<9[o7Jex`.Lzs5Tt++,=+R(u$LL&9qq 0
BNw/=2)Tu+K,:j6uDGi8KmXS]v#LlL2,5c\C;;a~J7nn*)+^
-8`E"~8frIyd\ab23;M/'nEDr=1~8_\2I9?n)erc[5q\{gRQ'E sK`n#.@;>o-8a">2|V$b3&ws'ow
]P[WV.B{P
P,>_B(_/ 03~M)Oy&pgo=zs&<m4wjY: 9=*c#QpP&YUrN.dv`3~yx}dA$<>c<~=s~MC| (O5/_vSv
#%,Xs0"GFb?ab^5wjbLvRs21sTYa$Rlo:vhY.gaR?|-entG3C&OT {0Hdupu}k@xjO&JWO0l`<8*sa#7'-
dhI;OE>>4e,70kX\5MV#
'gJhl:naKAI?\VUxXcMNs[&d
lQQxqfejQ
uxyNt;}7c ?# np8XD{. b1 QoGO,chZ|-wO^f'B'1

C|p{cJcG,c?6gm7n;r,zzb}02S
(n%extI<.Wf
2_}3;&
@]#bLCq.&sD4[sqnx(sktgX*@_,ywv,KKR/4[$.I[qm
+K}uC#&a={`C
1d-IAXyyUU; Rz`vrgj<"|?#"bjs*&gf4[7bh1|H0Ywa'q">Lvb][2nQn5cmW<BxkfS0U?L^:B,;:A/PA(
uZoK$@$@$@m$x[~GMj&4%fNal!+>qxc[LPC1?>W#$O% XUCgwOLs":$!NM+P8le{Ey"1G-
Fwk.6=+D\D#WZ{} ='c.,4yzbWu+O|@'b(hu QA2,"!{rK>m1Qon@S(&$)r^xly\,"*8yZgW/bZ?_|=n95O
Za]7aLJxwQ".rWWcAq-zxTQ*JP%VdBCX|wQ*]Sjym,\7&B?
I`etmYVy==bV[2DrQ4
ZQ(e%P'lE`
vkC\p#X~&VgLz
;.chEXTR!MJZZ-Butm#c4$@|X<3NDK<TH|~rs/2nnWbd)Gu-zr!AVG6[9m2BYERF{|{B X]sX`/iQ!/|d Pa-
CTKRI(j>& h+R|`q*EgZ"s2?
><*
}=n<]'$X!+t
D<0+d@~#B7!1?*Xa2qE(sKr1_)}OCL:rdWzU`A!0Q'uu35HL[f..H6^Yej&1n)$*[N05Tb=l"ZZ,I&cWZi
"x?7J\Gkkca^[}
2z!%# $kC<M_Vt eL8eYYYu
:Io$H|Zq+YZF\R*G&j_,h!5S'Iq<_rP>n mvo&fd52{A;t*iM'MIuy *H,FdqqAQ+&w9A
MP xA@*E2eyk L9bQ%K&{q7P-bp~[mvp>*t?%NF"@B-K.RZ\s#1.Pcjy2wRm-gC[~LR./S&s>)+r!b]?FT7nb"
w*2tRU)O3F7xT2iqC"pj>i1UX4O[':qk5?L>s3N1m:nl
XC{P&M^;a^`/kq$ytao$]yuFRmR~.<XfOh89R4*p4|M_/K^H,%1K{AJZ+e;A}U5I'$'eO{-
%Cv}^s+Z)#gUXsg=a+X7 &m OVie]/)#q@5]B3V|5]0myE:<*45eW?jQg62LbkQL< n:/N6&lo4
zN=q|jwZ[woM,i"]wk^nETcpIcV[#Zr>N[^ @k?hS]ox4p-K`ahOG+x1LV$O'(z-
xW]2VL,yQj't|iCZ$"`0O$P(O<@VV&~NViAFZM>,STT$GVbOP%. k2r [Cp4FQAmEE|=_\5Jwh
o=`}"^'IYJv
=j>o&>ObaT%Sw{aSY!b!62/Br{F!255O+w5TfI)bX6/?2$-X?o7OlU"j*Q]RtW9$yn5aXx>7P"
]972l3e3rDXHR12]%a/AlJ2al]jZh"L9j;X,;kHAMFV)7
sD#(1MSK<.+u}ATZF~))92]1a`;B)zmr[w/<*Gee_5%e;
Iyw\y'.&z>*FZ`ZmI\Kr~u{64YIHHH@vK~fCN|NYd`kGqX}<m\^2:Q1~pSB<frd"4m5VYtZ/_r&6eY -K1jZ\
6vA\71z`4vX _/Kc<q//FbX6
n#Mk4j^HJ<rdsw>i'f
/daIppotPsCUneOtJ K&K "
hTyHneyE)OdaZQ)K
Y=D
Yexq{"do*Oj:%_2e)i>J\gv5Q)K`9]2nUa>bQ-,e+&E,@a]RQs/GG:dCc=t^H+?wiP2A7>d4.R]yo;7;X w}
~b1mpax~R#Zo8g8fWpEI[?Io,@ OUY!qEe`xlu6H=>ejS\.^2xqH#U2K[\wn?aiu`! b^ *ZY5R>Xao`zd|.r
s8'e/MmU-pXAS3Y(df7YB4=Gov$q/$?,-w@,"VGp]oi_ 3 @[8ZJ
petQVbY1Wdp7o['V'vm!}{KcL@O$ez'5LFh@T{xbL4ee!6T|0ZKuLr$UcFJ<9f#+65%2
[P>]<5!b*])pa! LW\^!A*]P.Zi.uNrHc<)3!uLIo`.)P?/2]$AVQ1yr$%KM?2a}R6
+UblbXyJ5
+-os}Mb Lh#9G8WXgeQ 9Qa+$db ah/"+/%
FwAIY)r6fKU)2HDw
 x 6IIdSyb^$P'q%^"R+mZ&"u `[UI{(Kl,Fl<LHLA3=(M,e!LcYN6mm
21 B#{10mDU"HF#d;d<+O48KAX85C\(CU&lh-
XW2wePq)6k7U)(Q/=T/ns/4mNG|k
0T~#<Gj9.9>WGx,k17eQk EQ]K& `,

[abL
EX\;I8tg*k)]sLKu%V(6h\vX,}+uj<'_i}T',Vq!0,`Bdr&5ES\KK`0xk& 8AN)Z
trY}l!;lg6Lu!F>mjZB<t;ovD)#? Q6iSpW L`TWnX"5vigT Mt!i,DL\}e?o0BToqr~%@tt8N@8K|/R$\^M5
U:ho`s`qVFvV8&L
'H[3'=C,0 =##bG*VN5Z]5HHHH'iD]VJNlB]@"#ZO8Q1Dc"-w&vH,IbU/& Y8%
  8t
B}x>D8L3Z:?J@: picb*JWw6)[vKy0v]</ @l}>'[THu!t~pE;0:9n_}I"Ww
=,HHHHHH:=<@6]+/`nlrhGt
jln"
xel'leHHHHHHHliap:~X!8;MWJWN4meCHHHHHHH$fnni/^UEoYb@/0+tM}%JWD\i)`#(/1zX1qYw(VW9ZvB2
ND$@$@$@$@$@$@@D(t
^F0neU"Uj%`8dT>r$jmkF34.#ZI*L4
."\{D1LX#RtH^IR
AJ@x{>U,*AQIuu;BjUY[
b@pRAtg _/RO3Yg 8p_t}_(Nuw^&=nVn~IW`L(;b
oLxL89CHx-
[a&QF6t
nAClP.'9""|l^~jjCa9o4>j
iX)Do9,
W)DHJ@K
gX0H*"QUM
*rw"bRAP
$@$@$@$@$@$@$p,Z4#HxS}os"Y4/l<YvITXzEz(OnDDq}z`So`$"]4k_?3/KV7<
Zz7C3{#qFonAX7*\1Kfj#ogMc.bC$rs zm-Dq7}WMm->@Di1iRL2Kv- o\spx,
<(bZ\$<ib}q8iMukH .*E \;qB|!bg{(dTJL9uEu-}xxsm"4w/1- 8:\XL7w@H>,O q$v]Bl[^%OB!mPS/8l!|zb_=zv
FW""J4OqqFEEqaQk5[$#@_lk.F]NTCXZ,#FDu{g!
@8K_e_DIny[c62%_&Q#@[#%hIyb/q#BRj}Z*x`KbP"llKqC4a
wq(n")h:u[@fmX t"m3+8*y/V!"ZhOAS.AXB?-Th`#
`4"J#GbfDUE}f`.vuiN-%u1Lm*HV)Wx.blR-CPoxz""Hka
-D$@$@$@$@$@$@@Pb7Tx
Wt FV4td'K-
5UQ,n~+gQ/BAYumS~kBXyHSfUQL*r14^.)k""7]=#J#9HHHHHHH3pB<<ByjAB]8/=,EX)tdP[hu?NfO}Pf_X`4v
BDpTum$gSq%SKbZeKGIRF|pJbet$DCL5F& D:T(U1jj'&U.^7O~
G{
nY,yq(dtcK MjiR"AV7o,X#<WOiDzl,'yOn,tiOBGXV$ZKRyHHHHHHH
:BF5Msc4+*CL5U%B_qx/40BYu@V^_&}*'XLjk'<CL@BcW `:A5{]^mc#j"K$K[& ,:DD(^eEb
^=a
YMZbN<"~=G[YE YNTS$rDbM+hKBW*(HRkt1>pUH jKF<jdaw 9 {*X^{d}R%j>.~s9x~
(`o`|bQ^|-
n![X<psI'^c23+4HVK(%VqSbfo/*HDp@}Dm%NZb?qrIK2)yb^/w#X|5Ho<b,bHHHHHHZ!juW\&VGZPKI[>udx\
6rXA1]m!_/tAFr *.d_

Kds8,XXxgF=_Rf)qQFPP2Bz{qM6YT?.Vq
lb1]:u.^]Ba+| o[|' B,- @&\%ABUVAWNRV~krFc`Wd!,`$<Lm8F{fb=b,#W$p4@UM-
*Z!cZCs!bL\1@w8qe31;d\/C!Lr1$Oz^>XoFFffhTK0*o~\9
eu?QA)Q=:4QKC'{IHHHHHH8ZJ
pekVnx>*G1sH H[&Vw'/47lFx05$_I%ljKc
_Y6[V

qaDsi0bA
-v)DhmP!"O 2e{H< OdMomiV`}).0*ZH
"J{JjP"Qmn^Wg.lKcZ0 v233fmKQkRV
\T8_/LKILh)+tZ4okU\KI~axv8JzdJ?uZKtkcU1DhjcW(Fh.X[$@$@$@$@$@$@$p vWW@'LvJ
BpOCdNlV>>s+j8}o`|~yU8iKO
<HHHHHH@p'[[)5R8_]RHHHHHHH p.mRtZQ'=Q;wG @Ep,tYX)Rk>f
Xiw 3@E]Co-YPJY)DBLvBXWS,HHHHHHHhh!fEJH`feiSWo^.z"1 iKmtSom^V
/*3KHZ{2!t9<HHHHHHN3m-t Rh
(,)QVVt<j'E=4C T+5Q=(O;2% Am4w>\`.zD-FVX\ hGm-"-@Wy8q?a" 8 MPjD$@$@$@$@$@$@$@L
Df$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$@LEf$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$@
LEf$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$@LEf$@$@$@$@$@$@$@'GqQ$@$@$@$@$@$@$
@LEf?bJ$@$@$@$@$@$n$D@8M!
8GGGC|:}<xzjO$@$@$@$@$@$@`X@'www8R8O&{@$@$@$@$@$pVPY= "g{dHHHHHH#`?c
3C$@$@$@$@$@$@$pfhqfGHHHHHH8-S Aq 8P8N);D$@$@$@$@$@$@ggyd/HHHHHHH#@;
OI7zQ5?,QW8f v!jE:D`~d@r _n"j.b!$@$@$@$@$@$@$@?-Drso/;/G28h2_/}QHHHHHHH#1A>x C5kDZt
1";.3=okcEf}=& @.nos,,kq +OX^^9v{Fvi9VL\iibIHHHHHH#H{O.ac_d[q17%(gyE|}Ypq}F{:@IpP
4Gm@$@$@$@$@$@$@$sh_BV <9+7/N`Sl o\4j :|x6UAAg8aXh\(^0?IMcxEf+FEW6a" 8Ot2by11M\Bzv
6ncI=u8\9n(55jRKS
>xpQ?#J7O|Wt,rl}c;G$]UK:l]%]6u1llc@IDATf4*Lyf#Z/$@$@$@$@$@$@$ph_BLc
m:=VW't9"vfk7a`.!lF;awpqAfb<;Q#s#]1bQB>,@
?WWawfj
bky"*lCyH)<(cU1>5Q}E
Wj4WOR8W,eTvbHHHHHHHhwk!P,jV2Ar#X<X>p3t
[R&"
~_^VK%V_Yi(J29
t3|,{i?%X9M;A:J%/~Z<qr<BX~eip~/@MUx[qxJ{_bY|&\:7{6o>KF& 8
iujJ$,|}ptlO$E=j}P/.*tZ:D{')YQ6oOr@#cD 8O7NoE*.X7t5W][7?"\ml:|],Kw[<,Mb!kbA+nq?B@d>}\,0;<<zTb"
3@[Z;`kabdH3_U\=bX<N;d-$CXW
"U5(>."UNpzio'
"H,OOyNyj$-~7n(akP"8R/5?k+1pxh:<:F0;%sG(RkOe)~/r}(TP_~qZX4
C(J--6%T{G!4P_.+v86@0VNZ*]T34X0m`tDrNsE
"L~>h%PNp,
~{?
rlJ 2wGq
9t >a,.>z=Kj`+ja7eSSF\)X"E$@$@$@$@$@$p4 tizaxsZ\2j*j)!AbZ4AqcMWN@aIU8?iE-(^dYInX,
4$cogO~ZY2u2Q'K'Vx(/GYi>^o]hZ}Y,J<Q'un7P+Zr(.&#3N$<HHHHHHN+ZL|.K>u<t(O_v)V`_2Luo,ZR^7DDp
Q!+'8~Np|n~569y$h a{HAxp0r2 .N&g[`t+Z:
[5k$1Y2?5%T+;6%E
e^!QIiw ,HHHHHH:/-"S_,g_ v'R]$B\H^sG|;3:B|"~|AqCqhc)~J,ZgO09%A8,BWcz2xo'>onRMkWE[]X/7
q8? [#ByE@A4"q6HHHHHHHl"k)),]s!>Of q*&

b3 6w 5B.i:o8y4zEkw
I`{XDHuHkQBkC5g{:;=S."
#!Jb\RK#Jc)0o[2UxPDD@fF6ru`ky%.}iH/APp=j'6"g~="P
iCIVD||<:FP"\WojS+@/+J<bfohBO|>ytZg]U(WMjdly.,4<|{aq_$]WLsl-{IHHHHH-=Pr,i:7!'81}",EH%^-
ONm@SAAE%<*(dvf{kP~$ 3@JGbsGh <%,Y+.E%(*/ar' "6I] B2RfkWWD$@$@$@$@$@$@$@GxVu8
L$@$@$@$@$@$@$@$p"$f! 8(ZzHHHHHHHP8HB$@$@$@$@$@$@$p oHHH
/b^7"SV__os!pLz4xbj;k;3Ag3ggIH4`u-r39-slS@EJxyyA$tTdlT#::vicI'MzNN3N@"8k ==8

t\]]m&555&ct<{&K'U+www7lCGB@||<n\tE;~Jms'~w}{r#,/t"o>tM'###ie`d\q9s&y 3c\
HgIeHHL'74sG}SN5>;c&:Q;v,v]'b6l0Mi!!!JG8s{w>*|'fcxrhMkI/_,[S
o?I@=}lb$L4]v?<YGkUC"33*`(V;{($@8_rq; t":
0|zl&:1e,_jPlL:BBB1>hj%|pZbx%{Jwthggdwc_*BK]T@=5ssAvo
RvQ?u@vC/F
#E$p'n2Q|mN7:a:t0j(FPi1cPW9A'~'/ME"OM}Y+[^pI=Bt1l3O{iSy\}k'7nMvU2j)WK'^S?%ZF$`O_L$@$@@@+Z
Rzisz>jy
7Kz{=.TxxqNMM9s\17l-yOu8YMouqyj6:m4|2Oomwo 220H?1~xLRA6oltk
O?4nF*1 FEkdHH]'-rMz|Nq
?
c.tB>33_^`#|Z,<h|k1
ue766clk&]~z5YzZ
V$4?t_'[D^omRTTd
4Ykm;yMmKr=^jkcsr_0`yiS9o>>y1lVI6conj~=Vvf;MRtN}KIakiGt$Zvhm|5n)ml:lqe'-K_VV/\T_{-
WhmcI0pt~bMz}h~=Fh^-9#& N@E'8 l Dxn2<v9&/|=9xv&vsQ]wew
nt]z#|#ZN|tb&zLKh>};va'Soyb3r}asK5:5ex{{7Oj*1=ZVQEKgt}MM,*
B]ayu:
,+h)yaa(ce&c1m NKVk.@erP$&&w193kLb++iY{1/`iIikU\KT%%%XbY$??[nbESk=_
FnWkzn=Zu[fkzmu=vKumS}~k`cGJm2P&}5cV]ZukYXUZZj&_
`:lAIWMak~6UkjmS&=]1o&B9R'dfwSw_rzt@Cu)WOmMnmng"|(Zts CMn?
^ttun@L4M\EZ]o45FN/^l&'7oqoxZlpj,<\{@7:?kB- [o~\~&Wou+:z|Q-_-F~mSNd |kl7%].3h[m)
7`krk:gm['0jXTi}eHQ~5_uB&$nmt"{)NXTm:k~/NnZGEy)'vj<'|S} KKTWcS]hr.
`cF-O[>]uUZO]U@xFy2Q5@E^3
]6imsW/<fcD'O=nsN~a8kV^K:M[cN
kyK*=5kVD_+)jUKo=/=^y1yTrS@tR3o^Wb}]3n{9#4%=F-xTlcAWm?8?|a+Suli__ImK<^]UNRrtR7|nnk t
-:i`#HH#5
m<VKs631.t@wVng]z v!?5&Fc1?3154I4j7PPAA{}>w|<wa]3w3w3gf@+\FwF#y?Q<:tR1 *<!GF^tQYIyB
NP(b/.a+VDk5g?\(+Ol:x1l@<"<8F|2<QQGPXQdir^5-M=:}E~aQdA+g6q0n)'6F& VBIy8['|rJB
X0<2&W_}zhPbL|az7J ')'
zCv^RQE2JHw)rq#NxGQ'(ZU~)1xqm(c`q
CfAk|~F0V)^>s.6b??B0(d`ou?rc]F<c~{&6p"'RO@<N n;u8'V!93|&%T(-au?Xik1cs|+Iea<:>l<CQ2UFP\2K?A;S
HYZ#(#L`4k=t_{HysNQFQfU \KxKg2xk>@\]tEiD8\S.aqQ6q7GyLO
qH<pRC}L2%)|be 2UNY 2Q.x2rbBNQ4BN7IDEb!U<^Nwz<@a@`
A"

0 x?pQ)iysG
kaiI2c
q%D*_~H1\7(v`#,(/mY9hJrb,q0k(Ku$MEvZ0ki
r{B`O{B)LBf0$O~B7s4$(Frr-P.mJ}rFjro|,'~s_~Q;'pssN G@-4Y).*sf9r,NEe%yV|#ty:n2ZN((htq2:AGtD1
~Fh1zU8t* //hXpc9E(6of$WkeEe2sE9)v[4o@kc`e]P`RRCSX!{t32
2r!=`DCN.D:8r=e[5awUNLNaA2s-u1?du|5
AQZI'y7q04bhA.K/)`x&q?dB~m+SvP@o|1.=z"-y7?Zo'~o!p'sCP_BwL[~6d<ta`&b5
pFyB|-vy m8L+85yg'nj{N 87:.gmZUnZ=MChI_5uWni
t8tb`^|M=((lBwF]K5NEQiFA8E(V>g#6G)QQ8~I|JW,w8b0B)Aa.0PXP@B[@NHPT(?(<`=F)oyJ<$VN)!
 rb`,bEDB[!>-|r
+(a|6",Q
m,5c$FB:fs9:Z0&93 naJ<eIM^5y5o]q=uh?[=p9@71`x e2KZS2a&"vS,L!/7x0~b3uEh ~:K;]$N |u*5I4t,Vu_\V
9U'gpN.n!:vU=TVb=#rRQbG($8E:(W('QRBt1rAFC2tzciwtP

bg`J
eBy,pr8N>| ;y`ta4:K^xxp/jO!3HZ<'0PGQp;>s>Vj92!vx)PWzu;C:
Uh"OeJ,(7 wdF6daOf:<k 0-2;gm)\H5%0IY(x e8`mFKo0p
FZdnX6LM&>%gd1<tBA||{Mx>IyxpNP2sZ6fkV9QiSf}M"w'/El/kmO<]%6MiLK"o7QzP2f}@(W:,o -D
>+bGCT-<\OcPBP`btQ`<w\qfF66p#V) p~H?b\~V|!)1XTNJ;rbX=K8S>4fSLgayQ8=c4*AdS
_^Lb==u-+yp-cv!7_0ace+c\3"/!?i)x@UU8ESO^+c-x0Hpq,NxD0va(qgcl^@9YS6<)oJ<^:[jhae+%Qj78v.rN
8'P[[z@6Fh1k-2%dcyis5Q]YnCNGw:U:\u$Wp<#=:tUQSE5]#e(tCuhBP
COWn^{,YXk`".[d`O(8QPph#(0|)<I:P9bGv,p
#|0.`aPmh.: ~*@!O]ruqFTC{qmMno90Y>^axT+s{7K<~[cF
gCkjGlL[+o1N +^1Ppw{Dp3I3;*2nmU9'8Zf]B 8 8'PCE}pxV]31'DuTW<'pF11R CZ:xtVR;R\k"4Hg
s>Od`ACG,PWF/#n
^Pb)';QG)UqczN/D}=?/},'6O~%B[M9v;fC1`|yb'5H)!l-}Yio P@(yXOC]b6lEnU':W]yj:u /QV^;Y2r_^D-
iit<='@'5z6bb}Yn*6O)9nul|,RbA=$u):$Ab'{nyZ\rx_7u2+E h.>r-;:a*46
W^UE'.qypmekyK2baFDq>e=
>q#VXD.K.$_Y{A`Kf@QKxv6S_]uQ?y-FNLeyj;ub|dY/PL`1MEQ
P6XxqBf"l[uN\2c_UGFKYkx$^;SPx>/;fK8TqEwE4`
/
<W)-a1
}(/PuVs:uw'j@-JJ]K=v*OnX}GZF}
?i^-^g?;6wNjXi<CWc8=TwBam*EGtO5W 9'8rZu^_8+>:iSzc=>
NR[0c?t1]C N^!5YU{l3C`?wx\+*y
G}4RQXe3z'lANrr|8(Wau1zmc~C.0r[>o:8n!?"L=4";'[(s0~Yc22
>}7quDHVUH'H~Vu_q<ed=>ce:L`z'LM(k<L/_O_FY='G`^+~
]kUm3,QV!O8{P}:- vX3q{z]c7r4QU/-E=-s+]lKwT7!PjFG;#;\0eg&{/T9d8<tT`/]3:i mK<>=2p(u=
c^^QbdeC9hseNTbyqMFD_GJ[amJYF6lPU]W`#'F>O2qG}M<)+J(.Sn=2T`u`x_j`[#Ex_o*']uXC=D"l%)v7bS7_
kr``T6.[mAB@k8/8'P'TCv*NO}8WM;um1*nJ;~BW0m:=[GDZj5iedW\b/{H$k:k;:"CfM?'WSzF:4Q
-od;$6\:t8FqN a%6%Azxzoit^?)YuFzoGev1A;FDJ5rF0NL:|<< J)*Kz,fI!~WY9)'Sb-N>(_/{%{Nz
q'6] %SNw}TCyc$^;y
u{m(r]%655p2:|5Z/y\lrq@3!1YO6w8Ozrqrx?+b\`=>}Dl>~qETNa"4opp[|y!A}.cheh=1[y#3pX3!qyypNn95-
L/O_KyF
]ou>L=^9W?8]}[6-H_]~lKw'o8O]e)7_kxJMYnFuiaB5~n=;c**r=):QSO"h{M*{N 8L;n[#;uv6m/)jnd4L 9tlQoGys
~9sf4.#0FpmF`~_E
vL{" t,z-J
J

((:|z6QF{h-12BH17.LamX(\;n(JLe\+9E9k(O?-krQu*@Nd*gu^rRv`zd
Qh^_uUQ<ckeInV@=Q)~h; m3uvbV?F-d` -^P?#3O5<hgo>jJ@DFnT]xLi.5o'cc"xGeI^y(
q1ZP':33~\sM$wh6'O1bD$#r^x A{F{`wd'
ua{E]n5\\koFdKJDf@0Z/%t~j{Z\%sgjjn'Q#A.3>aX'l{6L$gM_HxS.#O?Zo|@{<$eZ^M26-
ej"Mzj:&hI#u9j}3=rP{MSvN n%)NeVh:B~"%E?q4 G)`I:\R7
yPsN2/:=PQ62A'`F$k
*PSNh>
&J3gv!i5+G~>~C-'OX2RYC^j4(+YCu!'(igx!'QfGNEi<J3e@0`aaKFY8KMMb`m6JY)0(Ei1
Lh_GPMcF$\'aPcmf&|d(=Hz
]^r2{GUm0<Q7xcT^zEoH6G[z~8.Ia\/k(_ 7XNm
wd%r!0es;?QWsQG\v
< nf!n65Esttb(0ImfQHdjy;~itxg.G{
Yx

\wj8S
Nvzlu2LJ/yw.;M)=*5#ES-}t,17Z|N?9'D[EQyEE85e<^(SN:tQ
/# /JJeaG_"9#(30sQPn`QE <"O |,U=(w}wp1QpjQ'rB)VXYS;p%PF^Pbo1eEq8["yNc#?@=Q[(q@gve<_~yd-o
ahL[5o]N$m
BhOq~!
i1[8{b
#iC"
8(8{7j>l@(_`?N 9kZ?uP+5sVl3v0+`5m=D'N;_nalyjz~tM.==ny
!.#,!U6,kI:y_5msv[?>FP~sm=)'@tW~vtaurGgttp03u;A1QeFQcN<F<8AH;>}3S
0*]Ai,p|<4(+7 #y{nQQs<P;3i,(V3J!-\1L ^
wRSL^r1(0%
ySf$X@bubBcSSx~*w!F?<AH?@!]3ZxR`>;2j+O)La NS6)'VP^J=~gt=/i(/3%%K~|(Q"#/q8lsQ6hE:2OU_~
8#`m'<'nE:{'^0u)vnd.
9n75^X,MvNies|d
YMOWuCO,o~y~9+SI#yE'J\F/7_lTn'^1R~-70s#[3mKV3`kn[EvO=W0:z_i_[:!zC97:g^0:(Pyd;k Kz
{eir
23"u(kCHBbefG@IDAT%C FM^|*0)=VrVLG@Cz!ez#2eY8G?E%VZ4|78G}ak|U<iGH'uaT7lPM-Bk^:`N(_c-
|c0<i/<1PqqH++"?i>>|&Mq?G))"rzWt1PFGYr6/>kKTVG= /Km[GfdpQ. /~%sX#<5UH;8A^U'6|&HBqN
p[CYj'Zp^^-QwCNNk,]\3Evg7fL_J;p~/][`OE/ahDf*^n'-Y6k@:C#n\oOPbB}aMtgzuml]7H],WK/R^$E% V:' MI.:a
ty/N`tLf:AG^Aanz$BQU<#?/9/0$dq{ePQywBCY!i@D#Mxw%OPss}[J(Vc
iqmhw{:6(|oObv6X'w:
<47XpsSosVx'{`H?NV<qyU&iqpu@-*%^F}gSG`uI.1\v0y,teNe{dC*]u9dvV)-fXdD7}fd7X2mNe6
3$'d,GCm6x]xCTTfZx /#N 8{tByjC[YG`NG{qP<xtp(LrJ}@yTW!ISXFy>|W,<wo:P9s?e@{m{vNn!9/mUf7
6:ie0wTPK7lQ F5L[:k\Ic?[U[Ykigl@S"KCJ;L5sj]j%(-x 65mzfeeg)v,o7 lnfhX% @zV 8'pN
8wCjhadm*)=J++Lv{Dq=V3yS]8g.nym0+\>i.p<b@8hce7 F}SN 8'pN 8M hQstCXv;`FpUalj^6> MY:$VuL?pN
8'pNF'pN 8'pN xpN 8'pN 8'@#F:W%.pN 8'pN 8'7Zx;pN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN
8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 87ZxpN 8'pN
8'$7ZjqpN 8'pN 87ZxpN 8'pN 8'$7ZjqpN 8'pN 8/hQZ&KJHHP`_Z*q'pN@Bb*9!:K
1}$W&*Q-eTctj_I I._IpN $zYZ4=4'[)QPiIj*\TOhpN J-
T7U(+56^cMWyEtz7uEs=];2k%&$PLc;5C\|QmT\lC"oX9K6WJO:"'jL_Ri.=G#tm[[s0^]+16/ 8'<$dX`h=^
bOZ~jHZzPo?GP%_KHg>zkz-[Vh5?lF3&]J4pN@6wEl=v4c75zy\Nzt4k{]g;ez.pN 8@%e{z-zm$hz{juzulH.~Ai i
$$$FcYY)3H+!,2S|
q-F
--O^c3`=[|%&%G^YJj1*-vCAT({iIqEZ\o[HrNdNhX~#]~Uqk[Yby?E8OMHf-
rF(W{oy&[*f48Z&gPZRNumUV],Q4xU*;'@y=4+2XC^5=o
KV6unpN uuj]5yZrSf4G?n6mW:fpn([WgjA*JP}<
 XV_jI*Mn}rx&j"mMi6-l)6wjv}wSO9^YR~\5k(XM]
}>9WJw<(/D9kSn=FWWM9jLA6C@y>xEC7I,7kZ\3>9+Ktqxl5MsJ_2M0V)vd<%{qPe][1t"a:Z`;45J7C`Bq}U5G.
ZA
iH+a6Et$nsT/H-0 GhbZ}lyewI#GMvnKutzNVP/[C^]hu3omR}4y&|4K[?Y;7i%*OP!G!},O#U33N
8'P;*/j2m[XS{DzQ7kP53PlJKMl5i]aIHE?Y;#E@Q
(_Og(1HY@G&VmjKB1P);WoQZd[LrzUd
So)y; Q,3?{Ui
eJet?-0RYdoC^ZwhBSbSyLR
4@)3AG~<O{UV
5S#OMk`J--ew 5HOg/W4B`KTP\n~kzG3T` %h[ck$)U[[beS5g?k~<Ko=/8ccj^2V{YziJHf>y5lYP/>N%*D}l
</iUe6-[RVZlTLTSO<p^`:O+C/3MY}Dg.QDi9qr4U|Z]S6acJ-
k?TAiz?jZ4ozI4]/=w')Tc}pUyPnHzLc&#\6qg?,2d,^^S[6

oyK2s=wWN}bh]]Y:))IrU>kEtS>a7oA`zY:fyG`kpN D^l??-
OkKtCTncRvMMuX#VfjwdzB)g}CFl!WV(L+H2b.&{xh@\qhgf'|jn#Go&M~]N>4l+JdWDvEqLVSYVK2jfE

Q[)k},y
ToIS5G;3*KLSiZe)hkhUQKW{6ef96WML%d&]rMyyT#zK^1oR}giTMtD.$W?:kC4WhJ-
:r:cD?[cVWkt?j_ct43xJKO3yzof"yQwQU)7=E%i<bRP|[kfE{%J+[Wc-f7'=5k1;ECw`0@j6mS/O
4gE-i#S.Fs5`0CNY~~7urOl'g.\+(X8wfc|:IVO5PK:R{I(+[bD9B<i7K'/o$i6}l#pN 8C#Z-
l{EA@a6\#z\k={hslUIWwTqaASBHkI=7W]#gpBq64Co=#K#EzobfRg~]A_hInnB5VC:-
]FJ>VmViQ{odYyZE8D}Rh0e=7iggdEmM&h98;]V$]27Xe)jGjtcd}dCipziZTs,3AolSQfRS7K')S8Z{LgUd:?76-
%SwwCzA}TF(~TeTPU'./vO=4Q~~AajT3'[}La
P[c6(HN 8'Pjha5*Sl;DsjZF~a$[Wp,7E2CWxkLEo
~)7~9:'%9M>w
}ziH[7'Wfl`ZPYc>[Q[4jShy7nTbFFq]VL-R7W
eu>tOo4cFzzzLs|Mg=r^K;S5k7m5t9
FMb"pN ${'kMxorhH#gob+xBr{oy^_Bj}q rZnvbs0
}jMTX3A'X"<P^{h&|0Q}[#@{)6!#2:3:>DGc 9e\J{/9\H6b~2S(1;d~.f)9p<>:MleX=67HOyKxl.ED
[TVBciV(PvSci,C3"4t},gaZj@f`az
eSQNS*dmZ*TR;NS-Yi hd'49X>kTC-W&sN 8$PF5^)t18r}j:{MFxPbe#/]?FJt<[?_D3
e)GO3CTmjY5|}5Sqa+/s4IRr_:J[S~.I(4i5OMv`L]z)4ZXo|8E%V)NPzxUyW6.7u?ao.7LpN 8&L2cIiL.MUQ[3`#
f@D,D;[` +E;@PuAi*<MMoA^|$W8 QVb^-;:Sjj`0%?%#W/L
0 bd"v.}HW_:ukMHa)(zg*/O*klQE lDy"$o>ZjOTEMcV8Yy&_X+T!EHR5o
slW>nh@mrhO6xr&V7Td=Fq6rX0V%N 8'PCtlSFeamK}:\YwDzbq_'U*fQxAQ6ES42R5s2}lo\o<4i]XSW-
3kI|Gi^,T5{cvf`s3hY[5?Q<![^u:SAKFhy]:Gum_?{N+j9yh4g>]|uSk[e7ZT"pN
"xi4h@WM&Nk#xb$&PT>26g^y[l{BPftKEsGmZB3S;HkfMkUc5
fsfyV3x$/x&SEW=iD4/v
eI7,u;ux6CLMKE7vd<X *JZ=f4[Rno<PM;Sj`bl~yCy)h
ds:?;?cZrz:bi`zzn{ESi
5}x3yLp:z5\XSi'w+ZP5(8'j@{Zl`3HBQ:`eWfz=6
W%JXt-SIIJL'Lm[_*V
[PBVMU='
%YW|Nsh]m6dq(FyGo|dGM54]c5>vVVij@:=kX5G7<~.i%Vmuc|BHOtmU4>O 8'`LFMolmy-6x9%ElQEE&6
i}F!Wv#aS;~d`uc.y }^fNc eb\]r+wQ5'~QylmSVh]:oKACQ.j]OlKms#O]EClS=k LaG,.r//OC,Z
>Mm/]cp[fD}.<o^|ma})Ia{N29va!,{OQj3LD=v=(QgPWss%UfzKn^;mF{iU9+!TC~=}:-XP Wa6(8'j@-
&;HClkfXcXb2lOo\/$"2L>WhF-
g{DjL]]{;[pQvf&h//?~|n_VzemLPQvs4D[;}e.|T2Vim;|={i<}:,x'gPnr(TVzZ$J&znM?YpM-
xQyBN 8'v(33;)&g67s]ak.l,n dwmZeYmKe-4s)8)11LoE?eeg('}m]OQf7QFr>{bHj/(,)g_R%c'~UuQh<4@i)Gg
V:c^82=V\yT[Q$f&
dSOLFui KlpmJi]fX?4ohX8`y~paZ1/bfu(H,EEKM0C
d,I93PmmJs~j;<pl}1J]n'y.Q}"Zt?v$%Qyz
=N'hogEhbieSl~`ko$;&K][#6bVrjcniW5npN ,cg."#OFOiTmC=G=0*AdQxG#]lco"PPn
Oafn9^xe]*jZh/hok)~c"^3Z,XF=
"3udT{p"=~PLkX_E;GC5k:qPW{#~l"P5<jag'pC4$37nnJK,'$iVhn,u!ZV,k!+;'72z0$=Q+6nS7U)t[HmiCd*q'{Cr
`
o-wNrV{V3"ohCBYeeFFi8f%1*,<2T@V3D<,NZ?[m!7y"-S+-JcX9:
L="sXU"U?bQc1^%n48^lSI`w'5kylXWr~rx0Sefx^WWznt}h;ub.=iS4zzt(
hu;u)!bj[sz5:Nf1Gf`C2,:,MK5zu&eO_3Bd<Di[X/E2|%/2>dMW9S.o<RYbpN
{Wx>VcEysxQ:WY:{;81*2b%%{:WZF343^TYHKd1R*mUBypN 8/@-
Lrul"]gfk,D~Zmk?\v+*/|i{YF{v7^O*Ii=1c7]Iw+syQo{T\>"J*RY!<qomUj`gq,fE'(Yz?6BRjSKG_3uiz[0WW>Hc3

mFjy{Q8N=ct/6}W;Gdb{W5>4 8'pBHfmc$<Q'E O/9OymNp_5-jrd2CKmv=1lZV wxd0BSG `_k[ah
;dkk4O/4GZoIt4:2YDsk5Zy;;A$EL;wlQCmibA7efa&LZ.otqaB>]!+Mcg/:6y-
Acj4(.vK@Y],_`m|erN 8'"l[K_ps57ojh3\4O{.=[=sNP0I9CXe%"OZS3{x qp)@#(at<{>n!j]dc#
#2E/9`(rxN 8'pN |m -j|zj,qozwi:yvoOh*{{
H8U+>|!,v'qsXVJU4!?pN 8'pNB!A/<K/Om3lTsH7Jg;:QRkHO 8'pN 8'"PFXx
bkfl(At@=8'pN 8'pN @-b3)4pN 8'pN 8"pN 8'pN =n{u9'pN 8'pFpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN
8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN 8:IuZ\('pN 8'pNpN 8'pN
8:I K*!LxpN 8'pN 88c(-J*JJ$$)~dfDvRMEf8{99X 8'pN 8'!PFD3@hB[4VJJn]1rzkny],\^HVJJL;5v%uN 8'pN 8'5k()g-
zQfee63w ^%JcU:5*BIfZ;.pN 8'pN a[V`Stm`A2U,b##drxj0)#A`R*yaEE%f1i-5(OR4EfG
9'pN 8'>8yv;^sm[uSx|G'u]uu\nj-Q-!Do[SPzj)8Pf-MhtI#QOPepN 8'pN a5nx#l-
?<ggf[iTh^
}:Vv-t>"NlphSCjx`^
u5y-lI|203ujyfarwX,46REs53ugz-Ht_;9vN 8'pN 8'5&PF37cCyIdeT"Mk%qD}jN_R%Gv~!@A;T[w7_>6Q
]XyF^mdejpZnfJvlTQ\md041%f(Qf'}ba}:WZ~f,CSqnV-kn]Yw|Xr ,<pN 8'pN 8#5k0E{&{8(dfj)8V}A+ RKJ7Rvh<)e
&j}]
KZd(^?S=a9bOx_/
P;(/N.im+m =BH'f/r1$MZJ|>~wof{o4:| lGd"*-V#b-:9I1iUIBQa#i|82pqHC
%aT"]vQX. 0&`L 0&+L (uy -aj(`[`<'#H/{!B!S
a!(2T.iEXXXRyW<m>]IJ)>\0zo`ZNmcLz[^us9m%!1U5>50a0:4
kRNV1=Pve&+`L 0&`L oEH=!#s{O[o1mEC>!<6"kuJ

rfD?|y[O7<IY[=q;FrU.X)p8&FXVF4oq^eNK)pq,K>w
33
Zd2L 0&`L 0&(ZLN4bQ!_kvg8a3
O.#=wXJIiHc"N'$,Yr$ SGN=vBD
>a[!LF,aiix$6U =6SQS?)UE>,8/gL 0&`L 0&(ZZv
&.?#za3
)iB#ao?[EVgO
!IE-PCK:H"gI
iV!#b#TH(B!dQs 0&`L 0&Lh/<G2}\aL%J8Mp.83V@" 2K8U,ArB
!,'S4!s&T3P~xzT*]8IOFF\ }|8^w]bNol}?{ 4+KcIfFB3-x^`L 0& vY'0Y`L 3^(ZON#L[4-6aXw:-
VxTkCdi4!Dq=pU::H2cC?~]vsDU[PCT7McB?wArch
>t
EHQiF7\SCF_1jHHH-4g;Hb.H-1a`L |1B_LrQdP!iQ~U 4Jp,<5@%7FZ+JYAd*Sh/fC ;i48BADb
%p0SVIPkP.VOHB /JC28#HR{-DBtJM^M-PZE.ip~801)igd"S\'KC'/z.{!]<B`M@;U$L 0&cM|<JNc.T*`kk+M&)-
vr`LxLL_mkWXX|}}QBF/&-V6#!$%&-HJ$0R[$0> Jd
t2#B@#Gdoqf4aBgJ#QefA?bWbR'H 0&@q'@IDAT(899[1B"**JVI$&`E@9!!!RcERA
ZE (":*i]{ hyx"_;H~!">b' ~dbXdb/`L 0&`oo/MEqie 0&@qvvv0J(dY@ZQQ| BL8`L& G]f-VM7w 0&dmbm,Bxi`L
0@*/`L { B35/&UT?b"g`!{3Vp`LB"C(~X
n `L 0&`Lpc`L 0&`L 0b&Bb3&`L 0&`#BqTL 0&`L 0&@1`E1`L 0&`L 0`Eq*&`L 0&`L sL 0&`L 0&@p8`L 0&`L
0&PXhQ9{&`L 0&`LpXhQ8n 0&`L 0&(f,(f=`L 0&`L 0&P8,(7N`L 0&`L 3e13&@>
z@/D|)B8`L :t:-232&^FBBi2* $#>>)ySCC"C! Y+qlX^v8x+0&@&-"p/id$JKG@Os)y3;~Bn fW8}:tmwo| Gj>
au,0b<"S+aQ:%&O\F-
^F
Taep%Tj.+fh(:uh+|ot3:Tdr%QfL{SmS_'xX:$XtH#Em`&ABij%`w0qT7)2q/d"hAC/+L&vAaGs*t]ML^a[K.L?~W`Q
2jiEp_R82(kZD3&4zZxFDD4MrvJ.BLL g3N~3bA`p.0}p`/D 5 fM%[")]vY=qY<|hyzg+_/UY!|xB0`,aKiXJV#{sl
M_4+^24+%F'F-{<d<Z5[@Oa?kW
+0$x2

lE&Y&@ R9G
KB!G4$ii*) h
88$-91*

Ke4HL+<_M`ozZxGOMIu@M
dJ-ZjsmPDFOMab'RC {jR SJt)-|VF d8Mp2q3(m!hL[V\=HH%P|jN=5EVjq*q/M>BB>Jdpp&..mC*9
)tuNw=aL^VVFU!#3p8}5;7 E|\|cjfNNL@ZK]64iHXD7,IC@:/1&4<k|4.UnV#2UZ4<
MkW ?i,b<6Y_cgcISOK0
S
iSqF4uBFt)Qp]V4k4^4o{Pn&c=-c446)9TbZRRaZiIHIUS*V/8>`/A g DNSj>{4ssF}*4;oM3O| 0&HOE}{tEfm0W&
:uVh w<HI.!ozB$/{wCfMy"mo~{O|ScYZFazh3VH
Zl-Zk<nF|oBum0|#WDe4k"T gmkBV;h<58~!I;~q^=;Ev!]K}Zc? "M6s.8GePv-_9SNun.kpV<+1h9Zlm:":16q|2'zvjf-
Zm;pN\\:<Iq_{]GgM2S{`>~.;1&u
:N%X&dPA){3$D=g`h_ODS/IJ?3'1LC|9do>DqF0nYD
+nsGLO7~%]@83'Qn>C
&V/A`!b;EU!w9!|yL74PSCuGKG1R{|w;}J$D<+izL=uBs171
L}
cQz9d2Bi!(:7@Cq;<
i|."QJlmi'c[{Y'DoVwChLu,tp`D gINg`kcS3)E&^U7*6O|warvEvmszm6$51`bj5V3dn 6vDI9.>Lx4K 4Frj*Hj
Yyix%V[w>lyVh2=#78{QkKTI~1z0\|UMs<}|Nl+T<

E,T
CTV16:&a8}l-o
-^#VBBBcTfh?:WB&5.@Tk
J~q}s55ebMZ2*6C[wg=s
xb2 4
{LGXuGW<yx1@@h&zvY#:vSF@#{#3&^+p_-'YP9cPLq$rch-*p8}8fK`NX:}\|z9T[Bc{e
\:up*92yyRU;Ya~pQ=Vv`X{:]BY#<
kQm\nEoP]@wL{0VfdWSiA_mWZwgI]*uA/^FZ?A&o[.[TFA)i4:nH'?&.RuePYl,\:iFEXlP#?r3on-
k!\G\,]NL(:xJTmqu5I2w]$ q$>fL (|AO/n
boQ#BSqV?M>1Vo')~rJ>M:7ATG<g[1qX)wmxsRX_ (iV[WKcF[EW rNT-]j`:ow%EDsi-
m`8r"k7VRc31Q?jZ+vKTWq^Aib'Iadc'X;9GYei,j4jEKqHITJ3R0`Xl6r&Sbh$sO?~.futo^*
sGbJ<9CGpPC~ u'`_
/ [}w^WwFk1cL <@m1O],_ 8h4FjM}#1z\LBBS1^pA?&!4i\
oO[G`s:!t?szVQ|3?J93?E3R9r86o<Gh#YmJ;_p%MH$GK}NmXPbH%s.`o_ C0}xNMP:A_!'/n$]&DZn?{Ne -
TO`iq"76;Cj3jM%LDsP!H(NP1wp<570`S%8u$PdFL lx7; #V`pRo(6rVL 0%@}|M1g$F~z$=pXXg-
PT9G'{5781.NRY5\4<'Q{7L@N=^"9IGh=JxFBEkwn=%Y
zBHM
lIuK'3@JWv*
0h@/*#>s-DI Oq77hZ#B|HchF#l}m+'& -kbeW9CX<wU~; x NTR04E3nHzmJsiKZ{2{&)m89GJPk|90& //
6q,[? }cKk CWl5}RtNERFY#:%{U^]YI>Ae E]rhMrfgIYn,d8|p6>sHCL5s%aDNM^Ot CvPed$$
adoKOaUfN/FvyKdqT d@xD 6Ki5
MIH"Pz;tjOt#
I2+7iAfi|fDvI$CC"R7@(:1LY!:<pzbhj)xEW7q @$6U"9<`K]hylZ!GJe(rH]:!
(C9:kH/sK kVc-rI>I4!01LE
dpi&AF,@8c(}iK)g'b;IB{C'!G^=Q--L1WFJl]+-<#:Ym^>X=|R G(udb~|cR43$/%N}@iIb=6Vea[CI56 >?}a>9b,
rjj'M#uO>$AT %#cL 4MY;o%O[a&<iXt`1&T^wc-8v:rr*9VblT+-uG`!ddyr{QI3 r}S~RH.
gO[0{H\M'm

yiZ H[OJwvi_;~s##5jVpQ~JL0Cjl k=qSiWHJ 4Dg#Z-7|qM@9*G!m3H`[q
kl4!j2w|)&Pr}Au"^RgwI\lo>$X3sukDWsxmX~c+I]'8AH.QF+}HTuy^N~q[s52&nbg
YRdJo92K&Mh74)LM+F%2gr#;{3 X=
AsriLs@tQXvHUG*K;U%) fh:F_PZ>p>a-p
+t_.v++R)/Nn`ZP?)/='vnsW$#z,iw?IU`Y&s%9W*(.6wc'7-;b@=}+U
:8LfJ[hL 0""PrM\3L~7e"E0gWm7{n44Ap|l<3rFjM5\ $poMvnX*teZ\K6j
*Zbq|2>JV!SJ[jPtaN#E
|"ChJh4q" H;$r 89<-o}]f0.a=zyc[+0/=dF+Q?!wNbC~lQN\(T (DOm3wHw(k66u?l=Gp/SaHglJeQCkU[i?a,
G0n'Uz^,~+AU`0E
!g9"54^^ 0S9Z8<"[9h45f1mn?F3Od5-`Gk
9PB.K~2u7Vbd7'QOPdje\s"%Mj|2Sm:EKpi>[6^SJ %PmVF 6_A."V{f}e<K0"2lGRtxfr-H<ag}%*IN!Nd140>Cw;-
5pYYCV||O`L"Z!go!Ihu`mK?(AB7scysb##'&|?>EFTn>9S\mB(AT=JWx,g'
0XsZe \5MaZ$ #R$:4gY!c*pZLa2%?@>:sAPnJ_?M-q/$Z5CFr+w28 >a?6o[R4d23 \AKJ+yij:q "i4GID4N?KUdQr
t,Tl>LI0t>Gwa+ENu4VLlQ21]#XPp!YI@-]HM _|{-z:[t)nFZa$Q
/c-H[MXDBr:xrFe?AG:*p+aJWa_hXZ1*
AB"Af5;?)dHZm
zm3
&-jg?pQ.5l#4&lc
{2w&%A
mJ<VvaI qWR<}!{XT~r.,ZYWqm!/w 16/>;dZo
stbZ
L;=qaFz)(2#_ 4^I&4Ydh&d*Dg%F=miJ48"?rv"P:I<z(S"<.AA2:>}A!@k/B:8)WT.7 %
9PShAn&RZhZi~+.%3ir3URN ?W#%CA2ez[/+,"KD`F&"ZCooLj!shGZ Pr<j&z
}R/4#SEkC[Bb=(NEgNb|6cASt53M/#1F{m[U$vI},yud":.mfh*|$YKELh_o@JCa%Ke(y0&P.x}m||t?M{7CZb4]hbfI
y<s[tmPX1GH=ceL`hT{wALRu!5'` >7p9(-l-L,Qt4 [ZyiM%GF]^M<~5K }
:<mp>DJRKDC3x^:7z
DMC"'OpS[L&9&45Z5cahSo"MY4.c?w<dqzn5+9N{'&Mt
U<o-< OaT7Qw0I_KK7
w=0kHg]u_#+"Cj}kT
2&^mMV*TF>yU<sLDQn
%!dBGND2~PTQ`!U:-P3um:.Y1dB1ehhtLcuGV`T%aU6TrO8x
@N]U:{CgH;F$
p7oY6!aW!o99_+jZFwKD^GM79qbW9b,D%Q>S'#Hkt%na!Xd2*Uh<}^(Q1`hfxsLWzE\3st$E(QS0~R*<d0+[\~9
&\h!L@KdIiX9~
K`A61iU\~
_Fa\jxMJAK~JSAt`bZ,O;TA{"o@tl5ikatV 0FaZg,\'NP]
]gaN4,'?mXj59"3CZmSYQ
{&vw>%nTD[|w\k|;=TO%Sl/FAa_|;#2mBt&4w1epn\;.W9Jwwf)kl]$[ERYKe,Z?';}=J*mo_?;|f8"$n9
eBD ++'B|
Qs#{a"P ,Y_oK
T%<;7cqmw|6;C2qgE[i,~&y7z
PBI9>4}I3UO=5Fl0(Ysm!
WU+(b_&@ pAmZq7D>1a4r7#~a8wz[0K
hl:y G/a9?4[qr2w 24rS{HA=e:t$dSI3&9d"
+rsf G^};7p5CNM
1G$M??7.AgL7=_H0b%k6c8GuwFZu"T}~XUXy&|.l4,)Mm3]Ca2,O#{Xy?=%MmaX0wcmE'A%V|QiKv/g?
ps/AP,7P:3[;SZ?xQ7obc0y F}_m\fV(L\f'$I0-
3/u:v3{`Xq/Kg}7|4(w9,q%q*Gvac/jZQiQwyoQ7>c/XC@QQQ(SuKdQ

#gZ_5PM|i+oTW\`o3L[i$n]%\
lWh8/y*B:yOHmhNKA9Q_kI0H
?v ~F%Ceeu~+s~/'gOrDDx(d%\J*[P`7UCS]\vR@'
+:"^2gvgUhmK(")8^YRI~)Bu5
mP[p^1AG*=Tm35= [_D0U.V
/Ntl2r/?&T2 _prrMv
*FMR8TPzFE;hh
U.vs1W'!0(dcCh`RI;n.RIm`miUN&yd~gFim-3Pdo8S^R^f}"$0i9#>2

*
U*"Eu2\KRGr>//"}<hw1[!62&F|D~i Mc%i$"619|MG1nLGxp ibF'gbl*%4g
mF~NlaN!*,;Kb1%iFg_4?YIiUroHIIAxx8O;eH___
e99'T/]b&Ff2+Lto UcaKK99UG.T;>b;HHAQD;n=uu4Wt17<i^w&%LVK5^L-
\Q2{^pq+,w0G9%dy'L&ePrYd}7q@l8o-]C:"W<<G\u1B de
+J84!J{}VRP'72X9!|ZZ4}^$X9}80&^^L3XfbrhVr21)zv2ZPQo^y2N(]/R*]&K)l(Q,r$Gmnk1
S3X^>T;ih92n$<}\]u.Ug/"q0I p`EEhbCg1SKIB<!aI&Mgq8JU_R:I`!eR#$o)BMhFz
6[7oyR8G)X4|Zm`l+1ju}uh= 0&G "= "lG#OCYlW\#&PVhA
by>}5K6GI+DlMR4(imEm3S?Kj[5AuoRkbIP!RHc8`FBq);'Gr(+m\:3&*PR?!I{*1ZXaSC-L
0&%"csI6:9\M&PtgM/#g4.11Y!H<=43-MK_Js6=%YE:K{[hKB*;w$hS,!gSlF=ka,(]2'kx} bUHji6OU|ST\ui
0&@mqadZ\NFq@!#iZTWN%ET2pya f,NCh/%4O!v5SA'ZUL#=:Q(p aGlr9E&@wlJ[SXHM
v`L#`L_@uhv5+c]/&F wrGFLK tMH6+8&8xt]VQ<G{60L 0&`L 0&""p]GdrmR!u
OPXn,aB\Mcac&`L 0&`L 3Bca&4)e$LN:s^i 0&`L 0&7gn aJ/K!%>`L 0&`L 7sBWxIp+`L 0&`L 0|`L 0&`L 0&`+wKBL
0&`L 0& B~`L 0&`L 0W-^bL 0&`L 0&Xh`L 0&`L 0&J`+y[RL j$%@aVTn!:h43%CeA-
gUb;g4h=zh]i^c:j)6h~|(TD,'sLo$KOJ}tE.=0{He]zm_O]

o1>'dW_2-.oMQR[\:~,yr?fNqV{0u%J`/J@ ?>{>4q|v\E&ba^Fo`H'E;1}H)h8w~l.-
EK/%}<ErYi?E\u|x5td|;8~Xh_=0{hy"UN_|@e
/%d.0<W%bqc_R{eLQn]$TRo/"%j+jML2
J
;wR<'>}_~X)yZ|(<]K$Gwa6mNI7!-c4J-sHN(tX<|NwD0vAO%LM'>P%Iub:% Ojln>U,m;2M
>|iyP _<M0/'Tn)o&A*U:f&2m21IhrTSB;]J%e(r
"wTMBt&Ps
tCIeZ }LaBK2"SSQg2{ #5QRT*
O@Q&&N*LV
G]pTTzz(8%d*DF&t<Rz%7Wh5UNY!Xp
K&2s"=PYPudR[<](bn+Eu3HO<EN?_wtHNLF1c-!=&0+O"[+-abwLoB#T
&#:,KL~}~3k"iqCAa)_]z`Kr)j59.b\W
*/yaMwJ Z/4U-gM#KmE~A{hR0Oc5=xqP0G&z9
z{Au)^<oJS]q5f%p?k!jQ,/LYg2:zo0^xep@inh#[}Vb%~M~FWHOaV9ttqgG&4H&/[DXyaib'zG>~:\?=0{$Sg=EjZ
Uq'^%}p'SFfs>SF6dfTxno!&TM !Nu.Cv}l7N\]c.HxUWC(-
6NM+J;6owsk'2={OUkMz4IIyKWCNZ8p?N[0g<wcxJm/U
#&MB*1A>u'?!48IS&e)ju^Uu;OY`H+q:%Cl,
MqbaaC'[Yygx6]PCEM3XI<G+]wg:1]!7.`s[(kogcqC{Kuj,vY-trzjb9dpXb
ECC,$S@IDATbIFM1exT)DqtplDD|=Y-N
[z
+p{9Sx5ks^oJO5WDRE\viZ:QY3av<@s[W>g
ptR;cbH@.0~h/XgM|.UJxvb'7e[`AfNk^5KbDL0`(2kb}+n(Yma=LU1Xi(:&A9p.F.]#D3`gGGu&`?`^
]n@~cA)]EFM-d1,]Qw&QU]iQvPl M7p8N-jHBak%J|NXEid=zQ P%&Nj%

c'4}3<=] WXd)nSM-s"k|44DV!*`.R?oFY])C;q;o9^z|?:PgM8y"1=r!1 df1~Z#Z8FTb
,PIKL?6&rD>uQL{=J iB,_`n `j6kq; &<{gw^w0P&{bA8Ol;}V_]7>D~d?&6lq>gPfB#VQXi7#`[
&O&KC\f}x4ht]mW;4=7Jz{D()qc&a2VGjKlUq%xKrIeut8K&OwCL6PCg{v/mdpAi'AM0"[6{7
)p)[MFlC}o71yG$4
V0zxkh\=~1N0[sfvV 0J5 6@lX
go=&9\]KeV;}G2-'E%C -
VnKs0Bob>R{N,Z#NE8E\&a`Z^Ppz6lT\*` h]/:xNC vT7Pz^YcI~_}4:WTp
^+~U`L*De*VE:uaiaj'zl5pmV/^b%HV|9>X 5Z P$>1Sfzq\RCMWN;`Z@ed|G4okL3oGmN69]g0dhtiR|97$y2M
nUbaLz?"6w?i6i"Qu>Z0#_>5kwAJAh,shEt?PnsSHPk
DV>e7#Ksv@$li9lW
P'WEQvAj1j0hT?ZU<H]ESe?O{_[g1a[3qq7u.<NySl_R`XAf,
v^=h~5DeAVLi(16fL+f=z<;)uOJ~XwM;}d5[4y$>['eL&J91nFcqh]wfnt6ul"u|))1|
H`AC (EJ@q+Z\[qM sw7W<w3ggnJA^}5>N1~
]0}liFaIl.^={dM3t
*Yc9=J\Q&H%aCi4%@.eUWEJi7+7][_#
f8$TFGge~^slM{ap93!"s};+R@<'"#9)IIHU)]\ a4,5
g|QAx&]9 S&OE~3%s(3w1s0v?f$nPYoac\uVf.>OIu$q<EQ1t|1n,Nz%|ykE}PvVL^?~/[z4AZ[Ld(BCQ"<>[~p
7KOWi6~9^E@OORYeFR
=Et!y14KG`9_M50C`H!kR//`a\)twT
1i|5Cq3)]&>RQ/c4>ABP+N]?EOyx?xQhn%h%`P
Ud.Jh*7oGQ07ThZdkg Z}f3C&^0/b70P4/W+h78Wg+=x|h,xu:}+/[9,b_aQVSg1i,IC9BabKu,|@? S(L\1JSc*]c#n};~
<P<}:.WS1pzjFPuE4xxRY;a0PCHS4EB'%!C]+14wPd&hS4CuqAvU|0Ekw[tkw+4\<M98x*Z`8s[^^1!#Gez
G),f^"#hY@/0/qb`g<aFN_Sz=la8~3,9v:NjFp\tP,;|4D
uy^
KR_/ljKkrzk{`G,l$#m
H_EF=<q#^Je4QkZ/AOi\4<jPh=C]uj(pjCHm:pZH
1'LK+=7a`S6A)<^
WhK
{0n
y|>!9>Tv&>VZBF/E5&(Uk!vWDgjn86k& b^g/B|U|h75S]J7cE~`WoV5HjFG-9tQ0N&pG]9[
@Vt#8yd#Ce\z.MDYS:n/B"8D7m|6gm]FUqxb_x[]LL@
Kap/iDcbE-:)"hTqWsQ<[8pd-
[vm4~Ub<TO|csg0;qvJmLMbV}l9[u6n[7yE?vfSzYuDo\>$XXsiQX0uTWx'xlomk;9il`G|t-`l~QzhDe!UK
hpk`8g MRibpWt1!#kQ$10,j4
vq3{vnnXt1Gk/wn'_@}GnBF ::y'vDa5^UMk<ymB)l0gruRhmrc0m3;9.cg?`R?+M+dF
w|]Ocw]iQf{V>=M>\Ce7{xPV3O^1KdfF4?LZ6Kx)]W|6szn9j(B%-t
[W^U5~)WV
Fv mXWzCS!\n$h Fn\F4n Umwt|Oqlg
=O!-Jb].W-.!^Qgf*u2@?!0:
d>\v:W%Xt61WqVn*HQtH,]hmLg1N=*MHW&~IWAS]Yat$\4l6FU*[r(9|&eiQ2:nJ(Wk{2!t&
[b%($} s1{c3qA b1*b3FASO
#uP4$ o<zfU:R}S]S\QC[:WlZnu>`hq~&6Y`vI6Xh&h(so3UU2g]kY}kx9eVm{v{Xy2vvTBL)wH,6kh[J5&m2~?
'1<xC:D.-<jBI6;@:%y?r#1-9IHR:QYaQ ,>$fa8GJvGO89st>-4N,nA+
oSQG>a(:^(tN~
qU,)aw!0~5@\`
pket75qyI9fU"B69/ `Kp7gMQcF
MnE%mnrn3Yd4R'1zeKF'=.\y!p,>%\p]2{~?-ZUqA 1X>KA_M7d]TJ)CZzuc\.jtBPT}6.O"sjY%|]+S,x5AJ!hz"7]Df
D'dR/IH[g),juKU

hD+3'"NCxFI!|kWhj7X(s5nEOY(j6:N
Gg"(y^=Zp-<])IbP<fKr}493x0\
sBK=hvj_gF5ka #<:
?{Tw."##5C%P'xbJw%$:H

M-'QgM.~*ai<\<&xnK%l@p76}
ZUAxfu).BgN
6O1.Qqu$1&]XB~K2Zg?!GhVnf[h\TL
Es:wH'xu6u!f,C[AZf%o\5J,A
U{N <Z67[<a261Bry%uLL8F%ULi8q8h)AmGifsAQN4A5O)Z8=j>Oer(9H\$HA(]*-
2cwq1CvR<b]$^{|y+,_=r&%!a:9r@+e`3KsE]{C@]k18~/x,2/W91aeKa<$"<jK/)*Ex4Z=dImi1tFjsaHa!PTy5oTh-
c)~=PT) SeUml=u{hsXrD0FVU*::I$g%ic?Hw,vVS/J3'(=WmCI'Nfml[5$-4cT6
(IV"WZTtMTws:,`lCJV$(B_~:^'eQ/"mplVQU6c`

47ZV-J0^ 1g*~8z-[yjNePGOVA4`+/qc7&3ikmz}2CO2M`7#i$ighy4
JL8`g?F]JQ[b EYO\|\=q^1X>c4/laUBq:0;|(u
mY+
!* @))pKV$k0oQF#4{DSU
_LU%('b6R{j9$A
Rw44@m($[GKX/W+hQrSS
+5YjK.qRN=yS%GK6b
!(Onow05:::[jRBi=G+%vQ:t++WlmlYmt~T85\#?%",yT_;DLQk}N<oC!P%yTthj6jI!cKc]BAG]hc,fO=]1.;s5}04YL
Mp+CcXYi@,1qDlG+pr++THK~kmHw>@*q[xZHx{ob
q4BHU8h(cuE&JK"~4 Pl.0Q3'"G/'+SV;>G:t5@+5|
1l
s9#}XQgE1j4.GkT$cx/,4p?3~&SUV;!FH];>+OTB)
uT{eO|"7/]k0
j AY,Tp#A{VdvKt:OOS>22d
a
O|6U&Fk$edAI#CQLY
>|SdzLk.E09i8q(*{9hznc,gNk-8gnjcv'
C1ogUehOl`i[HUS .z36D[7'(?\~u-;Y[HLMJC^~dckh9`.~-e; "
.DH5YW=jV;# Y4m=!'^i|Kh?Q(GW!aheE2.b(BXPcN=6`ux17)1c`>AFD T.!uCPdk9X
1FkH:YDqc{k*sO^D<z4pd&O-'zx..(kx>]}Y%.SPe?0"<BC/5AM_:_ PP<N]htE{Bf:#@QRE2q7xImtWM-
44.*iwbiN6&]&}f=c)\ .;H5]OvB^|~PXhC 0zfqgQ!;vTXg
,}$ qQ>Ud{m,_<i%I3v<@~cd,T@~?aA>z/YfR"NSv?8Nyr"C/$m~-x:[U<FC'p%!P-t9Mh<XphW<|DR`!KW-
KXbxNO8Km8R kQP-}f-T ABkOp6<
0g*/ \O4]9R^a've'</IO*R~YOvol]vr6Z()_=S.P>/_Zd.
C"7UKGxE?\L-
_lEJ8TmHJcYvPBcQTSqrli!}b,Q"~7]bcpv}>Ew=65#%_K7zUSjD{7l2A?K>}pa3B ,
F AAg[Z-|^xpKSX\:_a|vMocpu7VORcuEqq,]9Qm0l.UVj)Tu/ "#"ml?z /BfA1x6_d2EAoVQ-!|qy^~a
+ErRZ8u6o'pgSZmjfSV[m3Uh{ \<t!&iLKrQzsn=<p8/bG C4n?|l=@DX(n\<WK8X'2*Uzwf}GGWNb-
*igX/bchKr,Bw!
p+u}X<;UtD`8o/c<Z!!&B|# <~Y0QIA+U|{>z(F'S|HwDCP~{Tv7"4Wsy7ibJKb^g.>?
Z_<ybX<~FC||1cs9[6WE6w_6k]w[1}v)2@HPNrk ~c"?=_:w1qJ)FR/6wAF)XqeDk,?=CxXnzI0q,<f`\p*.\xhT[7|q_g
&5SxV3 5A-!O?Dh2z
_!8<
GGA`x4kB;y"Z%_y/8y3m<wF.(u Qi=Z>TKFt

.H[Ab?tHK[ear!Or*itt p1>_x>qIsUR?GVSB-D]Bb:6PbV}Xu\@C
=(%9I0b4<}{Y,_ar?"a9ww=E`F&|}q!&]fNN(V14>?
7*uwAYFhs=?(wsnV8#-T?5aI%A-[CD_Y8w.|X)E?Quy3qk18u)En]$!k {G3tZ
-Y5T}blW}fqO!d
)E WfFFRb[q6\h623uhOk+bhoS($(.
f-Qu22LWUc
]SO%]hbYZ;6t261jh
z"#&<6c\uMBGaqlhK[mYAgjZnWivf
I6=qi7!Z+xF]y+,Ipw+N<*.kcb`|t448kb$f~z%yoiU5VZt#jt
lgw[23l[~f#[*LM^_Ci`q'W(Pk00}Tx(
m;|` fly6*lVX(P2SBESHhFoQ*,|w#5VX40lhGFm{]#
Np'].iT8_syZ]=T9s%kj
NP|5Wos`_`rBa[O\3Vm[BYU7W+grn?Dy}A(s*vUfJz
L+{vo|*VgPf~5F3`[\@~34*4e)~BhGtt!19s
3:Q3Z*/0hl9xUF6UmtH@RR
p4c>O+x&2|Ium0cNW&9{yod*1ok 2yB6l&CV],
CK
(MA|B-Z\n,xH.UujfI&^1~R>n6E)6d>Fn&ShvsC9YJu(#!/
f'M43}E5-a1wZ|&41bQKC0ZVo\!MO&I5e/]9B(H=\KtSZT4g-cdx06,:h-Vp{>)2,gz{B`0Ps$&RSK}G
GuKzg6zqalya,Q
6kA<vHhS|!'CaQqf1gdkJqS*[~<!PSMToTm-mx:+L;l\5?`(X|@xDU|d!
2X{R)R| xu/F`~2|d:8jTT2-P93&gG/
f&2QRajc&&cr{<=,]q3e6 _KF_[PDK|&B|OZ3K2d5[HP#
VcT"JBE4<x@jBqD2zCg01p"9E7J ig$*UJ
+[f$i)QN X0UEuE2U(JimlfI|E>4})&7& Cmk6=)V)hkM@ddC
i^hMVRnL$We;tH[ZXHJ4PX*`0fzs2Q^Oo3~(zyR*)(R$&/TZIf2D
!xV(H3#32*st{KKK.EQ2hAG1mFD
$&&I>!9< ML!"s
}3S[fBp|6yFCRc"PJ#"@8F`1a8DD8LpQF&TuA=4"39g,O^L%X<",,2SI`lX8-c:eYnT[~GEf`
F}61v1gRGDF8*vDTxo%G%\<IK=0E$e]4Ncx>C|f(62O>Al,LCYT6*uE4PE|"@?y\|dmk/|\BBZ9jOWE6pX
*Z="S@@}h>n;9V=NNNd
Pb
INDI8#
SanN3A!ivS~F:2r-@R*w&J2aD&e$SA Id7<%+1}dj8uD`$s7a^8(316vk^,x/C~Y9<
\ICSUe@$5c#OV\ka88$uaN%%"1L->~Vjkp/JKbM"]D5nNML!=r.~A}ELI1_:>/"}
{];[mVl]sTS[ZP)M"es=M"a\i8:,}{"PHXrNH$sO6f=54@Tc?^y#9KCtAWH3K}>EQ_1hN!S
_WSO~&CKgC5G}%%ya(<'&#>'Wcc'FWY2b\=o<rTTW6"$`usbSXgvtby|P|EJ`MC.0$:Cx9|8^]ODI#@oV^EQHV
kEpQ/.:kR:]ObE(NXrn^UBeXX-DNQJ*2r?ewX7'EO7[@d
BQ1c)fmc
zdCE1eI]2vx Xp60w:t**pO~\~"~3 ZDs1wo4 C]>56_Hp\=EiS&>@lY;`pkt ([L_7LEcSH62,oa+c./S@@J^GOaU W,
*50~ry)nE2]B('2@.U_N2>D =W%-~78O`!6>Cpk9>}PcCybPf`c) -hz;GB!&EP)%]/SmTvg2Ed
7/nmo}UY,w_tdllA+5uB!"M`hr9

Zo_1yFv:[-dtUl~~~C.eKp-+"GFh/lR|E
89e/~G\BQUA< GCTGPX@Q.?
bXN ?,Y?G\20G#?vW=[Z*(>dm[]?po,`$Su7p&?:QjG!w_# (5,% f(";/gggtkU&Z%\6`Og@hl.:q)QUaEGn?FxB$}3`
H>)_2d
) S@L2d

) S_G?C"D= Ng!
'gUE

1FjNj`K]AE>HK
i`5
HTDe<:N*:51Rgf`:]QQe5`Ea-P[,#YF jX-^>O@GL2d
) S@L2d
[Q/Z
}6 _."EOs60g7~0{aP!kR,kBi{h2DX$!r,Mao HG!l)k\6$KT$e\}4
AXdna!*BDZh|1dX,1c&QX) S@L2d
)
S@L>BhT|s7kH&A+dsGcw#ekr/u7t;ZwL},?mmQw:N21n@F+531lr)l?G8@IDATmwi2Z7c?%p5*mblQ"f=1;/#FW
H)BL>xV6q>E:AX^
2d
) S@L2d
) SEq.xI%}SocKk< hksS,I"fN_suZH@)h_nB@>UV_2* fD{BVZ!}4t>:>
X5=Li1gWws]Q}B,=;FB,~[;9t~+cHY`G\FL2d
)P
L_b1O&\l2)\})otQn6*eg!V{oI)fi^IH29CjEqm!==
9:@UhVo##aSnzNHB ZIs.-CiXsR0jq3T*ky<7/?8SWZ]* S7"02$S@L2d
)@=r%>Fg3S;<y`"Myc8.$x3xGJ2/^FQ,?jiwoiE{Fm1/X
/m[L(r7R(z^mZtpF>(!XHfH`@zc8)ySnBLZHF'"*:[
JW=VDX3!@2r67`fp 96hm3sM2%GdKQ(_Mb^$Ee;}0oa,=<F)S@L2d
)P0t
2RkiIxL#ttr\6/+aNsDHIJ`m V#MMim !K:.Y L-P;!,/~#vC:><gT,h`+l_
VXMG$d(G\|43,,#-*aO oaB
TbN,{zy9%g]9s4zm:Pl(4~I)6>HVmJw- !c4.gb4.YIVdd 4B
S#0CTx:,>onrJX0+OA0 w<9L2d
) SO_PTS7;V.Da? -RLJ(]=@
-d4-*!B@aUBnB]>(,9u-0j1)@ESGWa$tQJc}o/F,
ZaY[VSQPelK(?E*+#`t9C<zqI.!q2,WLy$-
oUUQj9(r?e?Uwk/Gp3RY]X$S<bzMF*3T"[G_L]|E}gd(_M>gbU"YVp~MF<<)#Y! &UjW}9{
-]
4&)PEXX(N1.2V&zTElPO][b|P2nScxg_DK:I/!TqA4\ym0fqX&.vA/X
60S;TOI:)X04ON1/) S@L2d
9s0~{60A.Eap
w?HFw.6&Ve0pp4U.S7O}20|(r,F%*!9kNX8{ZdD`e%},Zt
cR-#.33`j+LZ-|Vf=|UaXq<J[`WJL9>iw&69A=]Kwq9^s:@xG8oS`fmq7 ,%eJnpZ
{F
6x)\}SwB`KE G{kX{WIK{ux;b/ew4NL7&~&Jb'-P_{"reKaw
cOpvRe1
#;+k7`A(ki>#8t
)D1|p_$e?Pl{EhviLA/q:}Bs4<4{j=a
Xo`1^N*##P>tWuG6.]O"0%{0p`y#^=n|q5jTMUqFi&~O]B[F[O{PVC<8WF1S#_[Q>&uHLe2
TXU<cX;;+/Mx%;q#$GY`Apf`Rdw}BB#OLEbY8U+xO .['|~iCXt@|r
N.C~==O/C*((:!) S@L2d
YshOl;W=O`B1G
Vlx`,Y%k6~}xM

q
;is`V!Y5p%|:fIs4 _xly}MJC
2gIP*VvfU~n #fp2!KrxaFnBSPbYL$
Bzw3p
$=
/nV{-Hy!Z{#A.+'/N`}s5.xVl &&\A`W$)&
oNgo0cIlQPvs!qcF#<jv^vuY
@]7} lUlgNQ~^]y>1.[9G^=E:Nx6K+<8{Py>6XZg%Dq?}dKaOp2mQ*U+k5g5rn<tf*d 37zJb8p9-ka_,LinXj+GleW
w*Pb3!7cfCP9f-9"?t`XqvW=[uX$-p QYXKWA.)+$\E4K<Wxz
T,k#APx8^?KjcZF?"rsQcc8M4t.,\R&|"{rBoquruprdDFC]0E6$S@L2d
)P 3;vjmcIC3JWMNc'tE&%N{)wMV@@tda-p:7Q[pVWxdM,]/" v7O#QB#OB1V:\hSlAUZD
GxIk<==+f#4m=LKpoch+,taB"\ZnW+:T.7F]}8`a06RK@ol
{
XQsx]v4K8"[
WHWp:k-36U
QyMI_ByH,&p+J,4Qe/%:Z:U
KCQ(1vw@k ,>@5fz->n?;~9~IAY
\;{tn76[FukCz#5%=@}UqoWw_*u*{@IkMIn 43L+;;)lIX0mt5aLk\3{{)i)#uo*)gV,-irV\?$,5)|
(
R{vE*i6|0Se2E^k
}
$I)ff
*KDRUJ+_6XTyE)4 ~
r~BKzl[%@KU=18[
Z2d
) S@L9)
K@24M1TLTXPnC9U+K}CXeD]YaZZr<2u3Z+,|~YmQBb`/>|HBNqVX4xE@'&hZB)%c#C!U7>HQ%]0%>F?!t)
!NN(R#YcVEG#+$"A[b@<
uubO4Yw>!m-LU!ERRC?zG{Z$'!J$\
BUj]+%&MHK{{$#;u`ko"P:%Lm;t_8YU,up?=t A3ad@H/k*Qak.h<x 8CW5u~ Z7]c>2W-s"Hm#s
8Ug\dB'\9ma#<8o@mXX|^1^_JIkJAH1|mqO$aJ1u2d
) S@L<ojIK<g/5ME\6q-|,%KVtySd1Z<:2NFjm2
czQd% N6 KBP`qYQ3L@)'uvV\F8+X d(KQX'V5 %!
_l}tfKLC61KYKuZIGcPhG20@y}|Y
nP`&Ik
+B^\,TrU4RP[_:k}xz5B;S\;<Tojep5ptmGO
eJ9oMVwb8wG"WPd)]b~A}cJ
pNKRV,pUB6FjBQcDUdRT%g) S@L2d
)wL4M_`f@6<MdIF-FNTF,+*_824f,:i%Aw?M0q9)'lvD^!]qPP1GBD*$8TdkR
.6uj>DX(BFHP(I!g&!.:\[qw'.#]Q_IZHiR-)PEK&h{Nr~g=K(.as+|vb3g_JWrAw_|^t
hjj=;Za1)/?"{1s1.&s.~,ICzY=BNLQ|l)qS.^j+WbL]{gBj5){}*>vm=xHhYu*(-oe!H
t)uymz7WPyO?{.tm9_]*0nx][T:???h\7c'hz$#8{zd$Dy_7Qv~B$l({w)Mf%jzvU
=6a]kls~22h5j%%I}u[K7j{N6o^mo*7jKujJe7o+vh}6za$%dCImO/!'jKGhgnoQio)3UZcHG}=dfj
ve]w+`7AlG:C{O:lk*WF]hfd9/w>{dl.}[sWo=J0Jg:FwKP_o+l6:@g:Uf={@9^z$50^<N<B{SO,Fon;W/||pmK<
~z}>$ ~8iN~)^5xErW^:_w//}[7di
B7O@~a4H1WzU;oS}3>bWm2ki7mP0[p0teWFUvBvP~.y
<\vR
q)fOn=+Rk$qD;Vp/5)y{##>i#7Bd.GwKp]6cJR67CGB3D#O'g/m8}e[9x::kX1u:lo#vHsmh|JY&ED'g?-

"m:aKNb[cN2hK ILj""z$&y}q<2~8$NtO:E9rtc[
nmu_\q4i@t?k)\o_qN]}UQq~~rAX]{mpm~/\m[J[/qN7odY/d?)"N-
toPLRSaSOgj~X6&^]frd#C-]@W=:GlVO~)$@`,d[]f?'MrQ:D>}UL?U6F@ ndum6_QQjTj9RQ?#V&(.+l
gmRJ}|"C/ZeE*HJIvqls?t]%Hqd?nE:w
lZOR+hLl2cUe]#NL6a4U52lNJ2J
VR3rF&)*!!a+%LFZMpj)%>W,R
\2[<#gn
{gIR-acKKJUZ$n6YiqeM@]X\9h
q{E*wju7^sTZ&tijj[i=H(U`+ zTn^WnP-fJoNBB^l"(q JM1jmxLAqblB;x>J
Ua\ZjnXGMdc|~+Z%g\UDdRRmvv{Q6+n[e'{P>|i=",6,o)`_`EC@:
(s@C]3u]JEjN* @ hu Zt2 t]-]JEjN* @ hu Zt2 t]-]JEjN* @ hu Zt2 t]-]JEjN* @ hu
"hmrprC=zB@@@)pujG~i--RUWJ/H@@@h/p`u1a~tM1Pm/U.>C(7h[n@wAB@@@
t~zX\~U(.&6?n_==+~tehTTR@CWFnKDX`=-%*E%@NhsB;qEtDc tnzX[vqr_wPq#`6\tjrCS_(cO:pe:mWD
~
YV@[^]? LhAOr'%'hE
on\MmmM_2Q}SDsYb&4Z?/g1~myyg/:~3u#5GxSdADCGU+]lVK3]T'<%.Vx}J*PU.(9T{W?0%rGT]['WlU\|"H>]
Fui#(22HQyz];z;_*(0 C}=r7YS-hZ~H;&?jw~N1um

x=zL.6XEW}#KNu]{UZ7=BrCf\}?*oa=,bu1?g{jYie[m.o3r2$8(kNS7;8{P
N@tQ#?BO~y"o, @eM~*tX]0qP*FwZosW&z9i{uZ}h"+}GTAw?j7 i|9ZWc:
>@3Ft8,X&$Co0LX=EJ['zc6{+nXo.RCGizM]-QLNe$i;v[`'/U-
Y_?T[%Un=0N=HUzq&x]]%RGqRm(R5Z=G>FNM0<gr?@w_o+d(3"$@
J*_U3|Leptzb{N4zh;v_8{k3kgye%PrtO="k0uJmQMuMaW!g/st
$k/^VN@jHXRS, Xt^Q=[AG)3(~T=;uw9zE/zy&
}d!JlozMau-K),* SXz>w^_^}PU))):0W'nib~4CzhN31&K .*Bq=H @@Ybo"p)vBB govoUV!ucpvTT3z)(Ev
w:)Stj]q-XAkDs=QE?}aNy?}U(h7?.i'X>u
z}:^Y_r,g>QJN9Km;>b|)uJJLPj|7%*"RR/jo>&+fhz4'[0Gz7v?h<A
_07Zu\63
u[!G p(!qmCZ:7hay_YN:O5nb/-ZJ]v`)9!NgwZY>hv>Du
+[a#Fh03J2wv9*{CjRyG~e-Sc=+lH_BBvEYsHpY[.9Kv6hL9ch3iy
Yzu&m)7
{&w~eY/*})>6;^mCN

t+H,Z !A Ac74?R g$v|8t 9-Egy=]`1l0}eQ-=_&|!x Vxq|7EOdfj6.F6
U[MPVUc\QaFXl-
z`U{m}),!f^wX[gS,bwAWHlCeUk/Ru+@@@8D:7h*mA-t;jd/--
W~Y9Qfx>jG/VhQMq,EFQvidW*mR6_EBDm8Hdj)JUmm+[TVjHz,j;-;'Rwg:laA\Yh(-
&RO<@@@@8Tw_gZ9,btq1nFEOY4`a
w
t;2=:>zE
!>rnM8LIk
"rmM@tz.\8l(=mEy4r}LM4\lJ
k6a>2\._M(fnV#{e2mC*c4b}8' 8p]l~]pw/16Gw>QunY6)Ll+mNE*w=x?UoZy
%Phe))6qyFz%gj#+a,z$nQm\EYo=&Ney[`k${y*x
z*&a*,/WVCV57&\Uz&J)Qut\X}~q@@@@KK
ts.uC"z0'w1G]_X2.:Ta'bcbl.:U!ueHB}$;n:U5/fA6i~jmL
!}3>RElP;wj(AMZQ]gV{ll/#UE?N/7fm~kyERl.w_SHSW7uU}#n/PQSW8Y$%J];w:Y}R"RD&U~c[uu4p^6S>@Y

s3>zsX@W>9_;MaG: N!i0v?}&++K:E"qO=+5:u(r-`Yil/90uhy~i-0svleimL;J+-
WY[}xA7eCMw.J[]KgUzW+{Cql\4\+4ipS;RNK:;i@@@@J&ao\>6 wxF.eBB BqWYA>vuYy{JX~ptt%^-PbJ[>I]
!^ M+zhdClp(
@@qT;a[~E|p0rnMQZBz%D)(tsaYo(!{wi~9vZ
I[n5grAw?6t]_$[Ww{(wn,W<Vb! pUev"/UJeKCP|-=p_8I`
"vCr}v}u WW69O7wVigy|Zo.;]=U4y HFksn]c
:w/-U^8U#+}Z8AW?7,_GpYkW!l"oi}~IjCCm.7jkUmP_Uc=0PM5d5%lVVr&ul*Tn_tNv5X0Z6
qkY2vWS^6t@FDyc`"M.i:N,^Qn?e-+Y>sn^g}1'+[w8KTXxm}srR78d
voSC{eTd!zuf }R>WS^]a0-\P.
zq
dJJl0
9H9}-`qzI7VHc5wX_Z5O'NU/l//<Jjf/MN>Z'LQKK.Fi#z;Dvkh8=d*j2Y'6EW"WcqH]9k$GVVF_ CcI:
@cy~1D}n2c"Tj+l)S(|W95~:
Zn[=2
P`;EFk:c0`3:=\37E-zlnFk}&F;nD
jSo8**EGn<~F{8cK?9uN[5kb(hK_=:]^3^Z`'mkh|zu5sprLwb~LYC2=1O?n]Ok>tLzORto4n=kvgk//^&
CUQj'vmiuy2]Z_?DNvDCUZ^8\UK>B9T@>:Wi^,}rJ&W/n{m]5skU: ~z{*m*-T
9|~~eoom+U^
}qKUwR%-5ap]7~"M_v}cuB:G+3>\7lY=ye Izp%?+o#JS
<6;(;.Gi=|]1&F][9V?;yZT2"Lzu4+O1q}pN4L#<wz_0E6d-_#V8c'6m+rtiw'K+ykktHL
P-"l%xvxSDi_h[&:AwQc(,W4u#Q%u(E3@_ %%w42!\%;3l+=~i:SYQz3,8__\uqUlhZK_9b>,-
J%;>n,PjdT2qF7cAj`zty2F*ZT~)Fo_> !@@x|acQ13ehS+,(]GM6bNh.Mt*;P }jruck-hasDP]vj|-vq?vEyq
i{uXNn)\xXekmY<ELTFj]E%&.i_=Z]NV*U6wU7F
j.AS@/vG7LxP}J5v`|xRyf<wjbp7SCn804}HVEYJU7l(M?T=~=t- hS
M8O S0NYv]*m!x&&u@IDATCvbe[i:i4*[%76&#XJ{OgZU#|zS3#M.xCWDwj-iwN=B%M9e
%):BXT/xN;l[A?~=x^rMj5cFPbFu6oF8 W{g;vt{U;~9}vxlz-]#j qc|jj]{9YUa_u
:cLE,[6ien-~O[-< >gUm2{KvhqCfBk5u5W1iMc{
lXE~Ro;SK35Q`S6O=mK|lJx};&`/G\I+"ir#X5OGm;8>VyHKh-Kh/^'>#GSi}kPckR}|@|-
6im#5zzS6],8#5bp_]81eUB/QBLRmHO6cea}dTel.9lJjKQ@/9 uZt& E
7G9cuMnk}N4
pu6OW3V=H%1I:PlMsWh8/A[w)0lz}.(,{*u@uK}I=~yO?rTg]ms 64mVBK6ha{Y{_]3FobnVab.<+O.&M$'Kw-
Lisq,QJl30@%y6NC-mm
W
ii@*zw
U6xp\p T=u Z7Vqw
Bz&6U&pBLgm4Loi}VmTca!$AW6H]R8t"lr{.[^Z_r'SA}+Owwf4?W.!izitR^<}6CE6'yC-
+rD?c.NK3%Gi+\uMj,&Lax#Vu)nH=;h'|R}ee5u$oQ.dOIgYv^*yymFs[U;a
Gy#'[k K[s& s6in#IXZZ?6=c;u\(Bl:_tRb=pvB8`\Y_xn1> vZo~=4|nEPry#.zN5haZ5I
w%wDhs!F]#_]<C/Cu]r'59Ni8PBdLXR:>91AQv.df_=]eUIjtf]ur!\IqzcWzoq6DT~]^N1M
;\G&t'Z|y~\("z4h *(v+^1Jh~oZ8z3zPw}?`>z[VqspD*|6Q]WPM>$",H7 @zU;l"pWaizIge/fcc]Sg}m)kklq)ghsIy:ub7s
#C)/<\F
nqvlj"y.x1|`?[v4M7e;9w*|NzLAY:vAdo0w!5 @dFIPw 6?DBsv
WmjbwTZJVh@RFdL2'nXQ;pv(_#[jA9*]/:4ds^n >ps0=sTZYk`w?.rM)LOScc }W*9Ekp~Zp;
bm8WlyHNn9YwEMo8z:R]; m(w?QWb7,z#WwxK;E@3h{&"|Z !cj[MP sagM1[MAw<m_nJ1ULi@vjUA
!$kCNzPT@@>Lu@@@LZ|@@@>L: |N7I'Mis@@@,pP.N[V2 @@@
Zt=*4o^DR@/>@
c`/ZXWQy
@@@$phhP%kGb`eZOH7g$I5u,)#$@x#\3Rp1y[]y|Lwcf| g!&Jx,`ELLRX]pv4uNKw_Yxv2

GuJWT>J Y
Qja+.xAB$hi?quO>~ko7Z;^n-=zyClQ.@#blQ2s~lph
<hC'(z@
MWF\t:4dyu!Y
eH^CzuX,]EZ?ngXUo`I&^P>5u3iI3>,'Vl=Nv6EB h7&w-V
$
UOUNqn=u/,p$hh{mHWOnUXY btvYUK%7F&|hh;cW.lKCzrSA{ZC 3;]u]U
>h}auT{DY!4*'gNm(t;swfMl]&'UQbtD4=)iNr^viuolc.X=6Ov]RbcZNS,JD-
vCzg|Vkejf89.VC/O'H=>v2]Pz(y7my?s{m42v}^<Cql6n{{omz>>k;`w
'l %ipkpN*mzcw?p-EGD(=1N qrz'Q?ua:J9Z28k@25i+A>wL?64$
NgU4Su+K5wNJkx&X>&d" +CgP~oRuIVwozwtv6j32`cf*NhKQ}
a3N
exq[^v3)sI}2TWZTY9zs[zwO)v*-K]sin*+*rzEDhjd+O8qym-Q2K'PU@ K!{~{EG[)^~YjckXavEn;W0 Ap#KI
6)glw;q;-cz,2-;_q6~Eg=']}wUhqVaRg+1%h>483I?8q^pov}'a*./WLB.]KtaJSu\nyqC+V
U~|mU-o<]SOGOZAYY[z|Wnu~v Y@To=GKs>*gNz+_jj$l~_G4j\
G]%Du#AMgI"-Sg?jS'E[&L QM%w#Y6A#!]v]szW4 `, Y [9uT@k4]x
tn*WtHnn1;[o{66v0g"#taT+MYJ(W>@^Xo?@kfkKu^I~@}V5%G
cys2};~`Sjklh]O/H]0{{KsO'sV&8{fY!W=sQ:5P0[zfs >HM_>nn>~NM[lh3ju;rpn>f~\v0e<M'kX.?=6WjkqeP}5!G
?)EEB@xE*UC(-(pKu-Sv4v}}Qk6)6T#$>#_jVdu}+V_QW~ulDM[Se;TjHaaTcF7p$&E)%Hj+wk}Y\KkJllnh
=hH9 S6$&1fVP
_-=25")B5:_mq}S#,=lwjK^eM_I
KV0EYFLGUTGjj[7[@k&j>_

t<abw]5U]Z]RoI va=w[Sq^o?y7rTc1zS&j$J?B6l'l6u{*'

PQ ej"}^1B?<HR
oIT ZnA/`c]/zWlbx;8M9knem~7n,xus>40()tYw~{vgVlU>>$\."cGw
KK.{FX[XCJN\1w +^ufs'z&/M*OwOE|&"7f{RE-
s8wxEj!oUNe":#]>nV_/RQM:%*>t;nffg*6urton=`w^5&xesk_Epo{eM~"5nU\{WCMhc}2w 80@|
%Y6N;_fW?<olZm-\/$utxrP^Ob.J}25q`1 QUy%^z"AhRd`=.<zia'Q%Y^X3@5[jG&+&[^V3+/W.*6}
z1V[whkhJXcC>4VYq5zq%^IT*\^Sj[l9UjhG Z-li'{7^]7_lm6cz^9,h-
1CTkg[8{R%kYn~&z@=>*15"H+*CrX\^%*wkcW%
(z>|~@oHGx}u\OM-]{&GZJW{zODZBv%ZnH{HgqDzQUlE?j(TmJnL-j
y:& =at.V Y@@@ -] OWuE(if{/p
ZS~i-Yn
<|fVRm-whVH
q;JRg:\cQT[B`on.f"!]eO@E5hM(Z}vL]7$Z.}k9]=-
V8MdTKj^Qn)@dpeuV:7Ha{JV]9H`R,8+lNg\x;;Xj/{x}6^6uuS_4&&~@Dv,u222eCJ%wlU4piw:
%***LEDD}&+bh"p=Uj+`Pw[zz&z*(585^;JT.;`juoy}v@zNM;l]Pj.{hgn[mgu|z:}\?%nf\7<ds446K
!m6
GknHKeMm\`zP,zxz&lhB[KO~G Xl;K]s 9%[qo.[CMEjuR?8LtMV?Fk:trm(K?^{i:n,?Q^9o 8pAW^^UR\ne
7!nz+}eVZlzG)F6pCSH "p`-i}+Z7{RVV8J@h,oJEA@"t9Pu ! p;?fwi>gNG}W6
My 'ty[IFB@ct{ZtF%yAz3}/ @0](wR IIIvGj_.h&"w}@+Vbt*:*t}iDZ|=>e"##RJuVb g/N\0rD
0}V-s'$@'&&]SuF\:$O[.ZJqI#Q@
ZcIvI 1=]N_.t Xt&B g.s\v%?7K=Ao}uF"hB'w;'t]y wT
`|gx@cAm:*E"! ptNC@@@6=-,Ni8
 @Wh
ZJ?aQT@@MA0%Y+sSJwS/-M

@@@N7.^ce" @ Vf z-6 BE J" p 8# hf* '@k3J @ Z<
DudP$@@:.t-G4(U[6&FQ{ oh566!n'J @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ
@`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht VE`# onJ @`Z8 ht
VE`# onJ @`Z8 ht VE`# onJ @`Z8{w]UyqIN1PT$TL A8,ojm[nzvY)^,zEuu2(pdNNN"/=Ng?w# -@h!
Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -
@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! Q{8 -@h! q_Y~%> p-R%&=!9IyA
KE' D
-,1UY:OWCJ]oU{UqW
.pB:M<SO]e^t}8Rm=di@@@OXL=}!U;$Kwz_Du"/c@@@@g>=_N(hn{U\)6U[D:kGM&jC_&@@@@CUH<k
u|\{m;Z4O]pa@h
Ru1g%"kA@pqX5@@@*-52,.=tp>M^nbT{S/\eCk_z;k;7@?9J-8fU~x}-U}=;{p%}|
)'sZVe|5Pct3s*~u7H~|9{TX^=[TVk^XtBmZ0q{$471nkoFP
5Xr6wl{;@@@/Z=|?}-x~,@XbEI7Y(K^O[TW[N{Zr\eOe+?GiuZ~aS]e
>]Sedfm/5k*+Ttq.?@|!JdO)C4zzW}=fTUc/m -|[HUMn7<R]$> a/BtUlBi~>];2+-
RYCTxW$v\}z]=>[{5uBmw]R:4m7/\9^v$~G
=s|Otqs`A@@@xE8k8]>f6dV
|Cw{=mvYj;x)*,,8 uW~97{f?-suwg5[Uk-;V?Xuwj4ogmas~[kstk Z&}Oh?^){WWhqz=ec @hhF!Mlea.{|v1k7x7Wqj
kEACcJ7ek-Y*..o^{S
q
;6Pc}7]{Hh;OUUQ?V
13G=jkP<Wio)qs`Lp^F]?wmZ @blZAO^l:dMg8*,$6RQf{Y&"5Una#=S3bU5u^&<[vpZU%/jGn3*v
+zDlhzgy' Z sZCu _{uon7jzheYF-7Y
3@Vj1hMmRW@aa->%<S(iFZF& q&N#joU+aW7;i\ot
M+ta'f[xNVL
S k d TZ~7/ETiye}6XyU5zUzzgu~_,@K+(W)&*wvs@Cqys`!5.7yN:hp=K_]]WKtu/-
7^zk$b~ez'ngVo>;jxFW=Vl@@@@kpz4'^Nk&::1?T610{}i]Jx@\q.0Z>Q\V}{?/Zf(kTu
[6y]?SjJKTk;+5g0Aj
JClAmK33|,R}iWYU^x
zgGxOM.lC;U -1~aCYI>BX{d{G'?oR'tCz6,$YJS#M,Z&8"`D-B/0Xb`Hl
S&zf
'u+;u.^vC*6a~NKn)6hX*:K{Jl? E6?)-e @j-uC597u*Ix!IDATyU^}6cla >y"b-H
"Y.p)c*AlkdP!
#q,s'$*%igaEYRb%,,j~;z , D=M=sZWWog{zG{+6{CpszU}itI+s[>Q^4\Z^;}cn# T=t[QP^}Z*Zyoo[*'*52-u"
8O WQp5]V|M\@@@x
~yX@@@@*-"4@@@~J[i BSYx>{Fva: mC+< C_{6,h% D@AA-VZkHKL $PTTb%"p%j@Z{`5}_ -
BmY)))rSY4=CiZm>-*WnEe
?MCowMno<z~|<Mg=r{}'z7g+JEw p-O<nmmn{^Ejjndgg7^u55V]]"C`#tx{P%.Pmk
B>5nwby(*V~ X *
\<oC=D)x?bKJIENDB`
/*
* Copyright Elasticsearch B.V. and/or licensed to Elasticsearch B.V. under one
* or more contributor license agreements. Licensed under the Elastic License;

```
* you may not use this file except in compliance with the Elastic License.
*/

import React from 'react';

import {
 EuiButtonEmpty,
 EuiButton,
 EuiFlexGroup,
 EuiFlexItem,
 EuiCard,
 EuiLink,
 EuiOverlayMask,
 EuiText,
 EuiModal,
 EuiModalFooter,
 EuiModalHeader,
 EuiModalBody,
 EuiModalHeaderTitle
} from '@elastic/eui';

import { ELASTIC_WEBSITE_URL, DOC_LINK_VERSION } from 'ui/documentation_links';
import { TelemetryOptIn } from '../../../components/telemetry_opt_in';
import { optInToTelemetry } from '../../../lib/telemetry';
const esBase = `${ELASTIC_WEBSITE_URL}guide/en/elasticsearch/reference/${DOC_LINK_VERSION}`;
const securityDocumentationLink = `${esBase}/security-settings.html`;


export class StartTrial extends React.PureComponent {
 constructor(props) {
   super(props);
   this.state = { showConfirmation: false };
 }
 componentWillMount() {
   this.props.loadTrialStatus();
 }
 startLicenseTrial = () => {
   const { startLicenseTrial } = this.props;
   if (this.telemetryOptIn.isOptingInToTelemetry()) {
     optInToTelemetry(true);
   }
   startLicenseTrial();
 }
 cancel = () => {
   this.setState({ showConfirmation: false });
 }
 acknowledgeModal() {
   const { showConfirmation } = this.state;
```

```
if (!showConfirmation) {
  return null;
}

return (
  <EuiOverlayMask>
    <EuiModal
      className="licManagement__modal"
      onClose={this.cancel}
    >
      <EuiModalHeader>
        <EuiModalHeaderTitle data-test-subj="confirmModalTitleText">
        Start your free 30-day trial
        </EuiModalHeaderTitle>
      </EuiModalHeader>
      <EuiModalBody>
        <EuiText data-test-subj="confirmModalBodyText">
          <div>
            <EuiText>
              <p>
            This trial is for the full set of{' '}
                <EuiLink
                  href="https://www.elastic.co/subscriptions/xpack"
                  target="_blank"
                >
              Platinum features
                </EuiLink> of the Elastic Stack. You'll get immediate access to:
              </p>
              <ul>
                <li>Machine learning</li>
                <li>Alerting</li>
                <li>Graph capabilities</li>
                <li>JDBC connectivity for SQL</li>
              </ul>
              <p>
            Security features, such as authentication (native, AD/LDAP, SAML,
            PKI), role-based access control, and auditing, require
            configuration. See the{' '}
                <EuiLink
                  href={securityDocumentationLink}
                  target="_blank"
                >
              documentation
                </EuiLink>{' '}
            for instructions.
              </p>
              <p>
            By starting this trial, you agree that it is subject to these{' '}
```

```
            <EuiLink
              href="https://elastic.co/legal/trial_license"
              target="_blank"
            >
          terms and conditions
            </EuiLink>.
          </p>
        </EuiText>
      </div>
    </EuiText>
  </EuiModalBody>

  <EuiModalFooter>
    <EuiFlexGroup justifyContent="spaceBetween" alignItems="center">
      <EuiFlexItem grow={false}>
        <TelemetryOptIn isStartTrial={true} ref={(ref) => {this.telemetryOptIn = ref; }}/>
      </EuiFlexItem>
      <EuiFlexItem grow={false} className="licManagement__ieFlex">
        <EuiFlexGroup responsive={false}>
          <EuiFlexItem grow={false} className="licManagement__ieFlex">
            <EuiButtonEmpty
              data-test-subj="confirmModalCancelButton"
              onClick={this.cancel}
              buttonRef={this.cancelRef}
            >
        Cancel
            </EuiButtonEmpty>
          </EuiFlexItem>
          <EuiFlexItem grow={false} className="licManagement__ieFlex">
            <EuiButton
              data-test-subj="confirmModalConfirmButton"
              onClick={this.startLicenseTrial}
              fill
              buttonRef={this.confirmRef}
              color="primary"
            >
      Start my trial
            </EuiButton>
          </EuiFlexItem>
        </EuiFlexGroup>
      </EuiFlexItem>
    </EuiFlexGroup>
  </EuiModalFooter>
 </EuiModal>
</EuiOverlayMask>
);
}
```

```
 render() {
   const { shouldShowStartTrial } = this.props;
   if (!shouldShowStartTrial) {
     return null;
   }
   const description = (
     <span>
       Experience what security, machine learning, and all our other{' '}
       <EuiLink
         href="https://www.elastic.co/subscriptions/xpack"
         target="_blank"
       >
         Platinum features
       </EuiLink>{' '}
       have to offer.
     </span>
   );

   const footer = (
     <EuiButton data-test-subj="startTrialButton" onClick={() => this.setState({ showConfirmation: true })}>
       Start trial
     </EuiButton>
   );
   return (
     <EuiFlexItem>
       {this.acknowledgeModal()}
       <EuiCard
         title="Start a 30-day trial"
         description={description}
         footer={footer}
       />
     </EuiFlexItem>
   );
 }
}
```

1.18 logstash 6.5.4

1.18.1 Available under license :

Source code in this repository is variously licensed under the Apache License Version 2.0, an Apache compatible license, or the Elastic License. Outside of the "x-pack" folder, source code in a given file is licensed under the Apache License Version 2.0, unless otherwise noted at the beginning of the file or a LICENSE file present in the directory subtree declares a separate license. Within the "x-pack" folder, source code in a given file is licensed under the Elastic License, unless otherwise noted at the beginning of the file or a LICENSE file present in the directory subtree declares a separate license.

The build produces two sets of binaries - one set that falls under the Elastic License and another set that falls under Apache License Version 2.0. The binaries that contain `-oss` in the artifact name are licensed under the Apache License Version 2.0.
Logstash
Copyright 2012-2018 Elasticsearch

This product includes software developed by The Apache Software Foundation (http://www.apache.org/).

==========
Third party libraries bundled by the Logstash project:

Library: addressable v2.3.8
Url: https://github.com/sporkmonger/addressable
License: Apache-2.0

Copyright Â© Bob Aman

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: asm v6.0
Url: http://asm.objectweb.org
License: BSD-3-Clause

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
 notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
 notice, this list of conditions and the following disclaimer in the
 documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its

contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: atomic v1.1.99
Url: http://github.com/ruby-concurrency/atomic
License: Apache-2.0

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Library: avl_tree v1.2.1
Url: https://github.com/nahi/avl_tree
License: BSD-2-Clause-FreeBSD

Copyright (c) 2012, Hiroshi Nakamura <nahi@ruby-lang.org>
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: awesome_print v1.8.0
Url: https://github.com/awesome-print/awesome_print
License: MIT

Copyright (c) 2010-2013 Michael Dvorkin
http://www.dvorkin.net
%w(mike dvorkin.net) * "@" || "twitter.com/mid"

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: aws-sdk-core v2.11.37
Url: http://github.com/aws/aws-sdk-ruby
License: Apache-2.0

copyright 2013. amazon web services, inc. all rights reserved.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: aws-sdk-resources v2.11.37
Url: http://github.com/aws/aws-sdk-ruby
License: Apache-2.0

copyright 2013. amazon web services, inc. all rights reserved.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object

form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: aws-sdk-v1 v1.67.0
Url: http://aws.amazon.com/sdkforruby
License: Apache-2.0

Copyright 2011-2015 Amazon.com, Inc. or its affiliates. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License"). You
may not use this file except in compliance with the License. A copy of
the License is located at

http://aws.amazon.com/apache2.0/

or in the "license" file accompanying this file. This file is
distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF
ANY KIND, either express or implied. See the License for the specific
language governing permissions and limitations under the License.

Library: aws-sdk v2.11.37
Url: http://github.com/aws/aws-sdk-ruby
License: Apache-2.0

copyright 2013. amazon web services, inc. all rights reserved.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: aws-sigv4 v1.0.2
Url: http://github.com/aws/aws-sdk-ruby
License: Apache-2.0

copyright 2013. amazon web services, inc. all rights reserved.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: backports v3.11.3
Url: http://github.com/marcandre/backports
License: MIT

Copyright (c) 2009 Marc-Andre Lafortune

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: bindata v2.4.3
Url: http://github.com/dmendel/bindata
License: BSD-2-Clause

Copyright (C) 2007-2012 Dion Mendel. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Library: bouncycastle v
Url: https://www.bouncycastle.org
License: MIT

Copyright (c) 2000 - 2018 The Legion of the Bouncy Castle Inc. (https://www.bouncycastle.org)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION

WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: buftok v0.2.0
Url: https://github.com/sferik/buftok
License: MIT OR Ruby|BSD-2-Clause

Copyright (c) 2006-2013 Tony Arcieri, Martin Emde, Erik Michaels-Ober. Distributed under the [Ruby license][license]. [license]: http://www.ruby-lang.org/en/LICENSE.txt

Ruby is copyrighted free software by Yukihiro Matsumoto <matz@netlab.jp>.
You can redistribute it and/or modify it under either the terms of the
2-clause BSDL (see the file BSDL), or the conditions below:

1. You may make and give away verbatim copies of the source form of the
   software without restriction, provided that you duplicate all of the
   original copyright notices and associated disclaimers.

2. You may modify your copy of the software in any way, provided that
   you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise
      make them Freely Available, such as by posting said
   modifications to Usenet or an equivalent medium, or by allowing
   the author to include your modifications in the software.

   b) use the modified software only within your corporation or
      organization.

   c) give non-standard binaries non-standard names, with
      instructions on where to get the original software distribution.

   d) make other distribution arrangements with the author.

3. You may distribute the software in object code or binary form,
   provided that you do at least ONE of the following:

   a) distribute the binaries and library files of the software,
   together with instructions (in the manual page or equivalent)
   on where to get the original distribution.

   b) accompany the distribution with the machine-readable source of
   the software.

   c) give non-standard binaries non-standard names, with
      instructions on where to get the original software distribution.

   d) make other distribution arrangements with the author.

4. You may modify and include the part of the software into any other software (possibly commercial). But some files in the distribution are not written by the author, so that they are not under these terms.

   For the list of those files and their copying conditions, see the file LEGAL.

5. The scripts and library files supplied as input to or produced as output from the software do not automatically fall under the copyright of the software, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this software.

6. THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Library: builder v3.2.3
Url: http://onestepback.org
License: MIT

Copyright (c) 2003-2012 Jim Weirich (jim.weirich@gmail.com)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: bundler v1.9.10
Url: https://bundler.io/

License: MIT

Portions copyright (c) 2010 Andre Arko
Portions copyright (c) 2009 Engine Yard

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: bytelist v
Url: http://github.com/jruby/bytelist
License: EPL-1.0

JRuby is Copyright (c) 2007-2013 The JRuby project, and is released
under a tri EPL/GPL/LGPL license. You can use it, redistribute it
and/or modify it under the terms of the:

Eclipse Public License version 1.0
GNU General Public License version 2
GNU Lesser General Public License version 2.1

The complete text of the Eclipse Public License is as follows:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the

Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity

(including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The complete text of the Common Public License is as follows:

Common Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON

PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to

sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance

claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or

hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The complete text of the GNU General Public License v2 is as follows:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively

when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you

received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or

otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software

Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

The complete text of the GNU Lesser General Public License 2.1 is as follows:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot

effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file

that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system,

rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed

through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME

THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your

school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

The following licenses cover code other than JRuby which is included with JRuby.

Licenses listed below include:

* GNU General Public License version 3
* Apache 2.0 License
* BSD License
* Apache Software License Version 1.1
* MIT License

The complete text of the GNU General Public License version 3 is as follows:

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to
share and change all versions of a program--to make sure it remains free
software for all its users. We, the Free Software Foundation, use the
GNU General Public License for most of our software; it applies also to
any other work released this way by its authors. You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
them if you wish), that you receive source code or can get it if you

want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do

not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is

released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this

conveying of source, or (2) access to copy the
Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must

suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you

add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly

provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever

licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner

consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.

The complete text of the Apache 2.0 License is as follows:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The complete text of the BSD license can be is as follows:

Copyright (c) The Regents of the University of California.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The complete text of the Apache Software License Version 1.1 is as follows:

```
/*
 * ====================================================================
 *                The Apache Software License, Version 1.1
 * ====================================================================
 *
 *    Copyright (C) 2000-2002 The Apache Software Foundation. All
 *    rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following
 * conditions are met:
 *
 * 1. Redistributions of  source code must  retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. The end-user documentation included with the redistribution, if
 *    any, must include  the following  acknowledgment:  "This product
 *    includes  software developed  by the  Apache Software Foundation
 *    (http://www.apache.org/)." Alternately, this  acknowledgment may
 *    appear in the software itself, if and wherever such third-party
 *    acknowledgments normally appear.
 *
```

```
* 4. The names "Ant" and  "Apache Software Foundation" must not be
*    used to endorse  or promote  products derived  from this software
*    without prior written permission. For written permission, please
*    contact apache@apache.org.
*
* 5. Products derived from this software may not  be called "Apache",
*    nor may "Apache" appear  in their name,  without prior written
*    permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS  FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION
* OR ITS CONTRIBUTORS BE LIABLE FOR  ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL,  EXEMPLARY, OR CONSEQUENTIAL  DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR  PROFITS; OR BUSINESS  INTERRUPTION)  HOWEVER CAUSED AND ON
* ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
* TORT (INCLUDING  NEGLIGENCE OR  OTHERWISE) ARISING IN ANY WAY OUT
* OF THE  USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
* OF SUCH DAMAGE.
*
* This software  consists of voluntary contributions made by many
* individuals on behalf of the  Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*
*/
```

The complete text of the MIT license is as follows:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the âSoftwareâ), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED âAS ISâ, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,

WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The complete text of the Eclipse Public License v1.0 is as follows:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENTâS ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributorâs behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform,

distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipientâs responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by

that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributorâs responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial

Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipientâs patent(s), then such Recipientâs rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipientâs rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipientâs rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipientâs obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in

order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: cabin v0.9.0
Url: https://github.com/jordansissel/ruby-cabin
License: Apache-2.0

Copyright 2011 Jordan Sissel

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: childprocess v0.9.0
Url: http://github.com/enkessler/childprocess
License: MIT

Copyright (c) 2010-2015 Jari Bakken

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: chronic_duration v0.10.6
Url: https://github.com/hpoydar/chronic_duration
License: MIT

Copyright (c) Henry Poydar

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Library: clamp v0.6.5
Url: http://github.com/mdub/clamp
License: MIT

Copyright (c) 2010 Mike Williams <mdub@dogbiscuit.org>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: coderay v1.1.2
Url: http://coderay.rubychan.de
License: MIT

Copyright (C) 2005-2012 Kornelius Kalnbach <murphy@rubychan.de> (@murphy_karasu)

http://coderay.rubychan.de/

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: jackson-annotations v2.9.5
Url: https://github.com/FasterXML/jackson-annotations
License: Apache-2.0

This copy of Jackson JSON processor annotations is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jackson-core v2.9.5
Url: https://github.com/FasterXML/jackson-core
License: Apache-2.0

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

Licensing

Jackson core and extension components may licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (http://fasterxml.com).

Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jackson-databind v2.9.5
Url: https://github.com/FasterXML/jackson-databind
License: Apache-2.0

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.
It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (http://fasterxml.com).

Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

This copy of Jackson JSON processor databind module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jackson-dataformat-cbor v2.9.5
Url: https://github.com/FasterXML/jackson-dataformats-binary
License: Apache-2.0

This copy of Jackson JSON processor databind module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jackson-module-afterburner v2.9.5
Url: https://github.com/FasterXML/jackson-modules-base

License: Apache-2.0

This copy of Jackson JSON processor databind module is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

http://www.apache.org/licenses/LICENSE-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jnr-unixsocket v
Url: "https://github.com/jnr/jnr-unixsocket"
License: Apache-2.0

source: https://github.com/jnr/jnr-unixsocket/search?q=copyright

Files in this project are licensed under the Apache-2.0 License, and are variably copyright as follows:

Copyright (C) 2009 Wayne Meissner
Copyright (C) 2014 Greg Vanore
Copyright (C) 2016 Fritz Elfert
Copyright (C) 2016 Marcus Linke

source: https://github.com/jnr/jnr-unixsocket/blob/jnr-unixsocket-0.18/LICENSE

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and

limitations under the License.

Library: jnr-x86asm v
Url: "https://github.com/jnr/jnr-x86asm"
License: MIT

source: https://github.com/jnr/jnr-x86asm/blob/1.0.2/LICENSE

Copyright (C) 2010 Wayne Meissner
Copyright (c) 2008-2009, Petr Kobalicek <kobalicek.petr@gmail.com>

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Library: google-java-format v1.1
Url: https://github.com/google/google-java-format
License: Apache-2.0

Copyright 2015 Google Inc.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and

limitations under the License.

The following NCSA license applies only to google-java-format-diff.py.

==========
LLVM Release License
==========
University of Illinois/NCSA
Open Source License

Copyright (c) 2007-2015 University of Illinois at Urbana-Champaign.
All rights reserved.

Developed by:

LLVM Team

University of Illinois at Urbana-Champaign

http://llvm.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal with the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution.

* Neither the names of the LLVM Team, University of Illinois at Urbana-Champaign, nor the names of its contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE

SOFTWARE.

==========
The LLVM software contains code written by third parties. Such software will
have its own individual LICENSE.TXT file in the directory in which it appears.
This file will describe the copyrights, license, and restrictions which apply
to that code.

The disclaimer of warranty in the University of Illinois Open Source License
applies to all code in the LLVM Distribution, and nothing in any of the
other licenses gives permission to use the names of the LLVM Team or the
University of Illinois to endorse or promote products derived from this
Software.

The following pieces of software have additional or alternate copyrights,
licenses, and/or restrictions:

Program Directory
------- ---------
<none yet>

Library: guava v19.0
Url: https://github.com/google/guava
License: Apache-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: commons-codec v1.10.0
Url: http://commons.apache.org/proper/commons-codec/
License: Apache-2.0

Apache Commons Codec
Copyright 2002-2017 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from http://aspell.net/test/orig/batch0.tab.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

==========

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at http://stevemorse.org/phoneticinfo.htm
with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: commons-logging v1.2.0
Url: http://commons.apache.org/proper/commons-logging/
License: Apache-2.0

Apache Commons Logging
Copyright 2003-2016 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but
excluding communication that is conspicuously marked or otherwise
designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: concurrent-ruby v1.0.5

Url: http://www.concurrent-ruby.com
License: MIT

Copyright (c) Jerry D'Antonio -- released under the MIT license.

http://www.opensource.org/licenses/mit-license.php

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: controls.js v
Url: https://github.com/controlsjs/controls.js
License: GPL-3.0-only

404: Not Found

Library: crack v0.4.3
Url: http://github.com/jnunemaker/crack
License: MIT

Copyright (c) 2009 John Nunemaker

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be

included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: diff-lcs v1.3
Url: https://github.com/halostatue/diff-lcs
License: MIT

Copyright 2004â2013 Austin Ziegler.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: domain_name v0.5.20180417
Url: https://github.com/knu/ruby-domain_name
License: BSD-2-Clause

Copyright (c) 2011-2017 Akinori MUSHA

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
1. Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

* lib/domain_name/punycode.rb

This file is derived from the implementation of punycode available at
here:

https://www.verisign.com/en_US/channel-resources/domain-registry-products/idn-sdks/index.xhtml

Copyright (C) 2000-2002 Verisign Inc., All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following
conditions are met:

1) Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2) Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the
distribution.

3) Neither the name of the VeriSign Inc. nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING,

BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software is licensed under the BSD open source license. For more information visit www.opensource.org.

Authors:
John Colosi (VeriSign)
Srikanth Veeramachaneni (VeriSign)
Nagesh Chigurupati (Verisign)
Praveen Srinivasan(Verisign)

* lib/domain_name/etld_data.rb

This file is generated from the Public Suffix List (https://publicsuffix.org/), which is licensed under MPL 2.0:

https://mozilla.org/MPL/2.0/

Library: dotenv v2.4.0
Url: https://github.com/bkeepers/dotenv
License: MIT

Copyright (c) 2012 Brandon Keepers

MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION

OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: edn v1.1.1
Url: https://github.com/relevance/edn-ruby
License: MIT

Copyright (c) 2012 Relevance Inc & Clinton N. Dreisbach

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: elasticsearch-api v5.0.4
Url: https://github.com/elastic/elasticsearch-ruby
License: Apache-2.0

source: https://github.com/elastic/elasticsearch-ruby/blob/v5.0.4/elasticsearch-transport/LICENSE.txt

Copyright (c) 2013 Elasticsearch

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: elasticsearch-transport v5.0.4
Url: https://github.com/elastic/elasticsearch-ruby
License: Apache-2.0

source: https://github.com/elastic/elasticsearch-ruby/blob/v5.0.4/elasticsearch/LICENSE.txt

Copyright (c) 2013 Elasticsearch

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: elasticsearch v5.0.4
Url: https://github.com/elastic/elasticsearch-ruby
License: Apache-2.0

source: https://github.com/elastic/elasticsearch-ruby/blob/v5.0.4/elasticsearch-api/LICENSE.txt

Copyright 2013 Elasticsearch

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: equalizer v0.0.10

Url: https://github.com/dkubb/equalizer
License: MIT

source: https://github.com/dkubb/equalizer/blob/v0.0.10/LICENSE

Copyright (c) 2009-2013 Dan Kubb
Copyright (c) 2012 Markus Schirp (packaging)

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: faraday v0.9.2
Url: https://github.com/lostisland/faraday
License: MIT

source: https://github.com/lostisland/faraday/blob/v0.9.2/LICENSE.md

Copyright (c) 2009-2015 Rick Olson, Zack Hobson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN

AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: ffi v1.9.23
Url: https://github.com/ffi/ffi
License: BSD-3-CLAUSE

source: https://github.com/ffi/ffi/blob/1.9.23/LICENSE

Copyright (c) 2008-2016, Ruby FFI project contributors
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of the Ruby FFI project nor the
names of its contributors may be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL <COPYRIGHT HOLDER> BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: filesize v0.0.4
Url: https://github.com/dominikh
License: MIT

https://github.com/dominikh/filesize/blob/v0.0.4/LICENSE

Copyright (c) 2009 Dominik Honnef

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: filewatch v0.9.0
Url: https://github.com/jordansissel/ruby-filewatch
License: MIT

https://github.com/jordansissel/ruby-filewatch/blob/master/LICENSE

License is MIT (very permissive).

Library: ftw v0.0.48
Url: https://github.com/jordansissel/ruby-ftw
License: Apache-2.0

source: https://github.com/jordansissel/ruby-ftw/blob/master/ftw.gemspec#L19

Apache License (2.0)

Library: gelf v3.0.0
Url: https://github.com/graylog-labs/gelf-rb
License: MIT

source: https://github.com/graylog-labs/gelf-rb/blob/v3.0.0/LICENSE

Copyright (c) 2010-2016 Lennart Koopmann, Alexey Palazhchenko

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: gelfd v0.2.0
Url: https://github.com/lusis/gelfd
License: Apache-2.0

source: https://github.com/lusis/gelfd/blob/master/LICENSE

Copyright 2011-2013 John E. Vincent and contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: gems v0.8.3
Url: https://github.com/rubygems/gems
License: MIT

source: https://github.com/rubygems/gems/blob/v0.8.3/LICENSE.md

Copyright (c) 2011-2013 Erik Michaels-Ober

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: gmetric v0.1.3
Url: https://github.com/igrigorik/gmetric
License: MIT

source: https://github.com/igrigorik/gmetric/blob/master/README.md

The MIT License, Copyright (c) 2009 Ilya Grigorik

Library: gradle-license-report v0.7.1
Url: https://github.com/jk1/Gradle-License-Report
License: Apache-2.0

source: https://github.com/jk1/Gradle-License-Report/blob/master/LICENSE

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all
other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "{}"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: gserver v0.0.1
Url: https://github.com/ruby/gserver/blob
License: Ruby

Source: https://github.com/ruby/gserver/blob/v0.0.1/LICENSE.txt

Ruby is copyrighted free software by Yukihiro Matsumoto <matz@netlab.jp>.
You can redistribute it and/or modify it under either the terms of the
2-clause BSDL (see the file BSDL), or the conditions below:

1. You may make and give away verbatim copies of the source form of the

software without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may modify your copy of the software in any way, provided that you do at least ONE of the following:

a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or by allowing the author to include your modifications in the software.

b) use the modified software only within your corporation or organization.

c) give non-standard binaries non-standard names, with instructions on where to get the original software distribution.

d) make other distribution arrangements with the author.

3. You may distribute the software in object code or binary form, provided that you do at least ONE of the following:

a) distribute the binaries and library files of the software, together with instructions (in the manual page or equivalent) on where to get the original distribution.

b) accompany the distribution with the machine-readable source of the software.

c) give non-standard binaries non-standard names, with instructions on where to get the original software distribution.

d) make other distribution arrangements with the author.

4. You may modify and include the part of the software into any other software (possibly commercial). But some files in the distribution are not written by the author, so that they are not under these terms.

For the list of those files and their copying conditions, see the file LEGAL.

5. The scripts and library files supplied as input to or produced as output from the software do not automatically fall under the copyright of the software, but belong to whomever generated them, and may be sold commercially, and may be aggregated with this software.

6. THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR

IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Library: hitimes v1.2.6
Url: https://github.com/copiousfreetime/hitimes
License: ISC

source: https://github.com/copiousfreetime/hitimes/blob/v1.2.6/LICENSE

ISC LICENSE - http://opensource.org/licenses/isc-license.txt

Copyright (c) 2008-2015 Jeremy Hinegardner

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Library: http-cookie v1.0.3
Url: https://github.com/sparklemotion/http-cookie
License: MIT

source: https://github.com/sparklemotion/http-cookie/blob/v1.0.3/LICENSE.txt

Copyright (c) 2013 Akinori MUSHA
Copyright (c) 2011-2012 Akinori MUSHA, Eric Hodel
Copyright (c) 2006-2011 Aaron Patterson, Mike Dalessio

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: http-form_data v1.0.1
Url: https://github.com/httprb/form_data
License: MIT

source: https://github.com/httprb/form_data/blob/v1.0.1/LICENSE.txt

Copyright (c) 2015 Aleksey V Zapparov

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: http v0.9.9
Url: https://github.com/httprb/http
License: MIT

source: https://github.com/httprb/http/blob/v0.9.9/LICENSE.txt

Copyright (c) 2011-2015 Tony Arcieri, Erik Michaels-Ober, Alexey V. Zapparov, Zachary Anker

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: http-parser.rb v0.6.0
Url: https://github.com/tmm1/http_parser.rb
License: MIT

source: https://github.com/tmm1/http_parser.rb/blob/v0.6.0/LICENSE-MIT

Copyright 2009,2010 Marc-AndrÃ© Cournoyer <macournoyer@gmail.com>
Copyright 2010,2011 Aman Gupta <aman@tmm1.net>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING

FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: i18n v0.6.9
Url: https://github.com/svenfuchs/i18n
License: MIT

source: https://github.com/svenfuchs/i18n/blob/v0.6.9/MIT-LICENSE

Copyright (c) 2008 The Ruby I18n team

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: insist v1.0.0
Url: https://github.com/jordansissel/ruby-insist
License: Apache-2.0

source: https://github.com/jordansissel/ruby-insist/blob/master/LICENSE

Copyright 2012-2013 Jordan Sissel and contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: invokebinder v1.7
Url: https://github.com/headius/invokebinder
License: Apache-2.0

source: https://github.com/headius/invokebinder/search?q=copyright

Copyright 2012-2014 headius.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jar-dependencies v0.3.12
Url: https://github.com/mkristian/jar-dependencies
License: MIT

source: https://github.com/mkristian/jar-dependencies/blob/0.3.12/MIT-LICENSE

Copyright (c) 2014 Christian Meier

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: jay-yydebug v
Url: https://github.com/jruby/jay-yydebug
License: EPL-2.0|GPL-2.0|LGPL-2.1

source: https://github.com/jruby/jruby/blob/master/COPYING

JRuby is Copyright (c) 2007-2018 The JRuby project, and is released
under a tri EPL/GPL/LGPL license. You can use it, redistribute it
and/or modify it under the terms of the:

Eclipse Public License version 2.0
OR
GNU General Public License version 2
OR
GNU Lesser General Public License version 2.1

bytelist (http://github.com/jruby/bytelist),
jnr-posix (https://github.com/jnr/jnr-posix),
jruby-openssl (https://github.com/jruby/jruby-openssl),
jruby-readline (https://github.com/jruby/jruby-readline),
psych (https://github.com/ruby/psych),
yydebug (https://github.com/jruby/jay-yydebug/)
are released under the same copyright/license.

Library: jcodings v1.0.18
Url: https://github.com/jruby/jcodings
License: MIT

source: https://github.com/jruby/jcodings/blob/master/LICENSE.txt

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: jdbc-derby v10.12.1.1
Url: https://github.com/jruby/activerecord-jdbc-adapter
License: Apache-2.0

source: https://github.com/jruby/activerecord-jdbc-adapter/blob/master/jdbc-derby/LICENSE.txt

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jffi v1.2.16
Url: https://github.com/jnr/jffi
License: Apache-2.0|LGPL-3.0

source: https://github.com/jnr/jffi/blob/master/LICENSE

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Alternatively, you can redistribute it and/or modify it under
the terms of the GNU Lesser General Public License as published by

the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This code is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License
version 3 for more details.

You should have received a copy of the GNU Lesser General Public License
version 3 along with this work. If not, see <http://www.gnu.org/licenses/>.

Library: jitescript v0.4.1
Url: https://github.com/qmx/jitescript
License: Apache-2.0

source: https://github.com/qmx/jitescript/blob/master/LICENSE

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Library: jline2 v2.1.1
Url: https://github.com/jline/jline2
License: BSD-2-Clause

source: https://github.com/jline/jline2/blob/master/LICENSE.txt

Copyright (c) 2002-2016, the original author or authors.
All rights reserved.

http://www.opensource.org/licenses/bsd-license.php

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following
conditions are met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with
the distribution.

Neither the name of JLine nor the names of its contributors
may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED
AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING
IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.

Library: jls-grok v0.11.4

Url: https://github.com/jordansissel/ruby-grok
License: Apache-2.0

source: https://github.com/jordansissel/ruby-grok/blob/master/LICENSE

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

(a) You must give any other recipients of the Work or
Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices
stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works
that You distribute, all copyright, patent, trademark, and
attribution notices from the Source form of the Work,
excluding those notices that do not pertain to any part of
the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its
distribution, then any Derivative Works that You distribute must
include a readable copy of the attribution notices contained
within such NOTICE file, excluding those notices that do not
pertain to any part of the Derivative Works, in at least one
of the following places: within a NOTICE text file distributed
as part of the Derivative Works; within the Source form or
documentation, if provided along with the Derivative Works; or,
within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jls-lumberjack v0.0.26
Url: https://github.com/elastic/ruby-lumberjack
License: Apache-2.0

source: https://github.com/elastic/ruby-lumberjack/blob/master/LICENSE

Copyright 2012â2015 Jordan Sissel and contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jmespath v1.4.0
Url: https://github.com/jmespath/jmespath.rb
License: Apache-2.0

source: https://github.com/jmespath/jmespath.rb/blob/v1.4.0/LICENSE.txt

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Library: jnr-constants v0.9.9
Url: https://github.com/jnr/jnr-constants
License: Apache-2.0|LGPL-3.0

source: https://github.com/jnr/jnr-constants/blob/jnr-constants-0.9.9/LICENSE

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Alternatively, you can redistribute it and/or modify it under

the terms of the GNU Lesser General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This code is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License
version 3 for more details.

You should have received a copy of the GNU Lesser General Public License
version 3 along with this work. If not, see <http://www.gnu.org/licenses/>.

Library: jnr-enxio v0.16
Url: https://github.com/jnr/jnr-enxio
License: Apache-2.0|LGPL-3.0

source: https://github.com/jnr/jnr-enxio/blob/jnr-enxio-0.16/LICENSE

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jnr-ffi v2.1.7
Url: https://github.com/jnr/jnr-ffi
License: Apache-2.0|LGPL-3.0

source: https://github.com/jnr/jnr-ffi/blob/master/LICENSE

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and

limitations under the License.

Alternatively, you can redistribute it and/or modify it under
the terms of the GNU Lesser General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This code is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License
version 3 for more details.

You should have received a copy of the GNU Lesser General Public License
version 3 along with this work. If not, see <http://www.gnu.org/licenses/>.

Library: jnr-netdb v1.1.6
Url: https://github.com/jnr/jnr-netdb
License: Apache-2.0

source: https://github.com/jnr/jnr-netdb/blob/master/LICENSE

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: jnr-posix v3.0.41
Url: https://github.com/jnr/jnr-posix
License: EPL-2.0|GPL-2.0|LGPL-2.1dependency

source: https://github.com/jnr/jnr-posix/blob/master/LICENSE.txt

jnr-posix is released under a tri EPL/GPL/LGPL license. You can use it,
redistribute it and/or modify it under the terms of the:

Eclipse Public License version 1.0
GNU General Public License version 2
GNU Lesser General Public License version 2.1

The complete text of the Eclipse Public License is as follows:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;
where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide,

royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all

warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial

Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and

the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The complete text of the Common Public License is as follows:

Common Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties

or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a)

promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the

minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The complete text of the GNU General Public License v2 is as follows:

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we

want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the

Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions

of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

The complete text of the GNU Lesser General Public License 2.1 is as follows:

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of

free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for

writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you

distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a

work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that

uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

The following licenses cover code other than JRuby which is included with JRuby.

Licenses listed below include:

* GNU General Public License version 3
* Apache 2.0 License
* BSD License
* Apache Software License Version 1.1
* MIT License

The complete text of the GNU General Public License version 3 is as follows:

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for
software and other kinds of works.

The licenses for most software and other practical works are designed
to take away your freedom to share and change the works. By contrast,
the GNU General Public License is intended to guarantee your freedom to

share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to

avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a

covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status

of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by

this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the

additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a

party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not

excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY

APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<http://www.gnu.org/philosophy/why-not-lgpl.html>.

The complete text of the Apache 2.0 License is as follows:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or
Object form, made available under the License, as indicated by a
copyright notice that is included in or attached to the work
(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object
form, that is based on (or derived from) the Work and for which the
editorial revisions, annotations, elaborations, or other modifications
represent, as a whole, an original work of authorship. For the purposes
of this License, Derivative Works shall not include works that remain
separable from, or merely link (or bind by name) to the interfaces of,
the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including
the original version of the Work and any modifications or additions
to that Work or Derivative Works thereof, that is intentionally
submitted to Licensor for inclusion in the Work by the copyright owner
or by an individual or Legal Entity authorized to submit on behalf of
the copyright owner. For the purposes of this definition, "submitted"
means any form of electronic, verbal, or written communication sent
to the Licensor or its representatives, including but not limited to
communication on electronic mailing lists, source code control systems,
and issue tracking systems that are managed by, or on behalf of, the
Licensor for the purpose of discussing and improving the Work, but

excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its

distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,

unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The complete text of the BSD license can be is as follows:

Copyright (c) The Regents of the University of California.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. Neither the name of the University nor the names of its contributors
   may be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

The complete text of the Apache Software License Version 1.1 is as follows:

```
/*
 * ============================================================
 *              The Apache Software License, Version 1.1
 * ============================================================
 *
 *   Copyright (C) 2000-2002 The Apache Software Foundation. All
 *   rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following
 * conditions are met:
 *
 * 1. Redistributions of  source code must  retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in
```

```
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. The end-user documentation included with the redistribution, if
*    any, must include  the following  acknowledgment:  "This product
*    includes  software developed  by the  Apache Software Foundation
*    (http://www.apache.org/)." Alternately, this  acknowledgment may
*    appear in the software itself, if and wherever such third-party
*    acknowledgments normally appear.
*
* 4. The names "Ant" and  "Apache Software Foundation" must not be
*    used to endorse  or promote  products derived  from this software
*    without prior written permission. For written permission, please
*    contact apache@apache.org.
*
* 5. Products derived from this software may not  be called "Apache",
*    nor may "Apache" appear  in their name,  without prior written
*    permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS  FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION
* OR ITS CONTRIBUTORS BE LIABLE FOR  ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL,  EXEMPLARY, OR CONSEQUENTIAL  DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR  PROFITS; OR BUSINESS  INTERRUPTION)  HOWEVER CAUSED AND ON
* ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
* TORT (INCLUDING  NEGLIGENCE OR  OTHERWISE) ARISING IN ANY WAY OUT
* OF THE  USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY
* OF SUCH DAMAGE.
*
* This software  consists of voluntary contributions made by many
* individuals on behalf of the  Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
*
*/
```

The complete text of the MIT license is as follows:

```
Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the âSoftwareâ), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
```

conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED âAS ISâ, WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

The complete text of the Eclipse Public License v1.0 is as follows:

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENTâS ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributorâs behalf. Contributions do not include additions to
the Program which: (i) are separate modules of software distributed in
conjunction with the Program under their own license agreement, and (ii) are not
derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipientâs responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and
b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in

any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributorâs responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipientâs patent(s), then such Recipientâs rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipientâs rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipientâs rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipientâs obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: joda-time v
Url: "http://www.joda.org/joda-time/"
License: Apache-2.0

source: https://github.com/JodaOrg/joda-time/blob/v2.9.9/NOTICE.txt

This product includes software developed by
Joda.org (http://www.joda.org/).

Library: joni v
Url: "https://github.com/jruby/joni/"
License: MIT

source: https://github.com/jruby/joni/blob/joni-2.1.16/LICENSE

MIT License

Copyright (c) 2017 JRuby Team

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: jrjackson v0.4.6
Url: ""
License: Apache-2.0

https://github.com/guyboertje/jrjackson/blob/v0.4.6/README.md

LICENSE applicable to this library:

Apache License 2.0 see http://www.apache.org/licenses/LICENSE-2.0

Library: jruby-openssl v0.9.21
Url: "https://github.com/jruby/jruby-openssl/"
License: EPL-1.0

source: https://github.com/jruby/jruby-openssl/blob/v0.9.21/LICENSE.txt

JRuby-OpenSSL is distributed under the same license as JRuby a tri EPL/GPL/LGPL
license. You can use it, redistribute it and/or modify it under the terms of the:

Eclipse Public License version 1.0
GNU General Public License version 2.0
GNU Lesser General Public License version 2.1

The contents of this file are subject to the Common Public License Version 1.0
(the "License"); you may not use this file except in compliance with the License.
You may obtain a copy of the License at http://www.eclipse.org/legal/cpl-v10.html

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR APARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2007-2009 Ola Bini <ola.bini@gmail.com>
Copyright (C) 2009-2017 The JRuby Team

Alternatively, the contents of this file may be used under the terms of
either of the GNU General Public License Version 2 or later (the "GPL"),
or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"),
in which case the provisions of the GPL or the LGPL are applicable instead
of those above. If you wish to allow use of your version of this file only
under the terms of either the GPL or the LGPL, and not to allow others to
use your version of this file under the terms of the EPL, indicate your
decision by deleting the provisions above and replace them with the notice
and other provisions required by the GPL or the LGPL. If you do not delete
the provisions above, a recipient may use your version of this file under
the terms of any one of the EPL, the GPL or the LGPL.

JRuby-OpenSSL includes software by The Legion of the Bouncy Castle Inc.
Please, visit (http://bouncycastle.org/license.html) for licensing details.

Library: jruby-readline v
Url: "https://github.com/jruby/jruby-readline"
License: EPL-1.0

source: https://github.com/jruby/jruby-readline/blob/1.2.2/License.txt

JRuby-OpenSSL is distributed under the same license as JRuby (http://www.jruby.org/).

Version: EPL 1.0/GPL 2.0/LGPL 2.1

The contents of this file are subject to the Common Public
License Version 1.0 (the "License"); you may not use this file
except in compliance with the License. You may obtain a copy of
the License at http://www.eclipse.org/legal/cpl-v10.html

Software distributed under the License is distributed on an "AS
IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or
implied. See the License for the specific language governing
rights and limitations under the License.

Copyright (C) 2007 Ola Bini <ola.bini@gmail.com>

Alternatively, the contents of this file may be used under the terms of
either of the GNU General Public License Version 2 or later (the "GPL"),
or the GNU Lesser General Public License Version 2.1 or later (the "LGPL"),
in which case the provisions of the GPL or the LGPL are applicable instead
of those above. If you wish to allow use of your version of this file only
under the terms of either the GPL or the LGPL, and not to allow others to
use your version of this file under the terms of the EPL, indicate your
decision by deleting the provisions above and replace them with the notice
and other provisions required by the GPL or the LGPL. If you do not delete
the provisions above, a recipient may use your version of this file under
the terms of any one of the EPL, the GPL or the LGPL.

JRuby-OpenSSL includes software by the Legion of the Bouncy Castle
(http://bouncycastle.org/license.html).

Library: jruby-stdin-channel v0.2.0
Url: "https://github.com/colinsurprenant/jruby-stdin-channel"
License: Apache-2.0

source: https://github.com/colinsurprenant/jruby-stdin-channel/blob/v0.2.0/LICENSE.md

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: json-generator v
Url: "https://github.com/tmattia/json-generator/"
License: MIT

source: https://github.com/tmattia/json-generator/blob/v0.1.0/LICENSE.txt

Copyright (c) 2013 ThoughtWorks Brasil & Abril Midia

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: json v1.8.6
Url: z
License: Ruby

source: https://github.com/flori/json/blob/v1.8.6/README.md#license

Ruby License, see https://www.ruby-lang.org/en/about/license.txt.

Library: jzlib v
Url: "http://www.jcraft.com/jzlib/"
License: BSD

source: https://github.com/ymnk/jzlib/blob/1.1.3/LICENSE.txt#L5-L29

Copyright (c) 2000-2011 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright

notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

3. The names of the authors may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: lru_redux v1.1.0
Url: "https://github.com/SamSaffron/lru_redux/"
License: MIT

source: https://github.com/SamSaffron/lru_redux/blob/v1.1.0/LICENSE.txt

Copyright (c) 2013 Sam Saffron

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: mail v2.6.6
Url: "https://github.com/mikel/mail/"
License: MIT

source: https://github.com/mikel/mail/blob/2.6.6/MIT-LICENSE

Copyright (c) 2009-2017 Mikel Lindsaar

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: manticore v0.6.1
Url: "https://github.com/cheald/manticore/"
License: MIT

source: https://github.com/cheald/manticore/blob/v0.6.1/LICENSE.txt

Copyright (c) 2015 Chris Heald

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: march_hare v3.1.1
Url: "https://github.com/ruby-amqp/march_hare/"
License: MIT

source: https://github.com/ruby-amqp/march_hare/blob/v3.1.1/LICENSE

Copyright (c) 2011-2013 Theo Hultberg
Copyright (c) 2013-2017 Michael S. Klishin and contributors

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: memoizable v0.4.2
Url: "https://github.com/dkubb/memoizable/"
License: MIT

source: https://github.com/dkubb/memoizable/blob/v0.4.2/LICENSE.md

Copyright (c) 2013 Dan Kubb, Erik Michaels-Ober

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including

without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: method_source v0.8.2
Url: "https://github.com/banister/method_source/"
License: MIT

source: https://github.com/banister/method_source/blob/v0.9.0/README.markdown#license

Copyright (c) 2011 John Mair (banisterfiend)

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: metriks v0.9.9.8
Url: "https://github.com/eric/metriks/"
License: MIT

source: https://github.com/eric/metriks/blob/v0.9.9.8/LICENSE

Copyright (c) 2012 Eric Lindvall

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: mime-types v2.6.2
Url:
License: MIT

source: https://github.com/mime-types/ruby-mime-types/blob/v2.6.2/Licence.rdoc

* Copyright 2003â2015 Austin Ziegler.
* Adapted from MIME::Types (Perl) by Mark Overmeer.

=== MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of
this software and associated documentation files (the "Software"), to deal in
the Software without restriction, including without limitation the rights to
use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies
of the Software, and to permit persons to whom the Software is furnished to do
so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: minitar v0.6.1
Url:
License: RUBY|BSD-2-Clause

source: https://github.com/halostatue/minitar/blob/v0.6.1/Licence.md

minitar is free software that may be redistributed and/or modified under the
terms of Rubyâs licence or the Simplified BSD licence.

* Copyright 2004â2017 Austin Ziegler.
* Portions copyright 2004 Mauricio Julio FernÃ¡ndez Pradier.

Library: msgpack v1.2.4
Url: "https://github.com/msgpack/msgpack-ruby"
License: Apache-2.0

source: https://github.com/msgpack/msgpack-ruby/blob/v1.2.4/ext/msgpack/

Copyright (C) 2008-2013 Sadayuki Furuhashi

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: multi_json v1.13.1
Url: "https://github.com/intridea/multi_json"
License: MIT

source: https://github.com/intridea/multi_json/blob/v1.13.1/LICENSE.md

Copyright (c) 2010-2013 Michael Bleigh, Josh Kalderimis, Erik Michaels-Ober, Pavel Pravosud

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: multipart-post v2.0.0
Url: "https://github.com/nicksieger/multipart-post"
License: MIT

source: https://github.com/nicksieger/multipart-post/blob/v2.0.0/README.md#license

Copyright (c) 2007-2013 Nick Sieger <nick@nicksieger.com>

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: murmurhash3 v0.1.6
Url: "https://github.com/funny-falcon/"
License: MIT

source: https://github.com/funny-falcon/murmurhash3-ruby/blob/v0.1.5/LICENSE

Copyright (c) 2012 Sokolov Yura 'funny-falcon'

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: mustache v0.99.8
Url: "https://github.com/mustache/mustache"
License: MIT

source: https://github.com/mustache/mustache/blob/v0.99.8/LICENSE

Copyright (c) 2009 Chris Wanstrath

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: nailgun v
Url: "http://martiansoftware.com/nailgun"
License: Apache-2.0

source: https://github.com/facebook/nailgun/blob/nailgun-all-0.9.3/LICENSE.txt

Nailgun Copyright Â© 2004-2012, Martian Software, Inc.

Apache License

Library: naught v1.1.0
Url: "https://github.com/avdi/naught/"
License: MIT

source: https://github.com/avdi/naught/blob/v1.1.0/LICENSE.txt

Copyright (c) 2013 Avdi Grimm

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION

WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: nokogiri v1.8.2
Url: "http://nokogiri.org/"
License: MIT

source: https://github.com/sparklemotion/nokogiri/blob/v1.8.2/LICENSE.md

Copyright (c) 2008 - 2017:

* [Aaron Patterson](http://tenderlovemaking.com)
* [Mike Dalessio](http://mike.daless.io)
* [Charles Nutter](http://blog.headius.com)
* [Sergio Arbeo](http://www.serabe.com)
* [Patrick Mahoney](http://polycrystal.org)
* [Yoko Harada](http://yokolet.blogspot.com)
* [Akinori MUSHA](https://akinori.org)
* [John Shahid](https://github.com/jvshahid)

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: numerizer v0.1.1
Url: "https://github.com/jduff/numerizer/"
License: MIT

FAKER

Library: options v

Url: https://github.com/headius/options
License: Apache-2.0

source: https://github.com/headius/options/blob/options-1.4/src/main/java/com/headius/options/Option.java#L1-L13

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: log4j-1.2-api v2.6.2
Url: "https://logging.apache.org/log4j"
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-api v2.6.2
Url: "https://logging.apache.org/log4j"
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-api v2.8.2
Url: https://logging.apache.org/log4j/2.x/index.html
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-api v2.9.1
Url: https://logging.apache.org/log4j/2.x/index.html
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-core v2.8.2
Url: https://logging.apache.org/log4j/2.x/index.html
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-core v2.9.1
Url: https://logging.apache.org/log4j/2.x/index.html
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java

Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-slf4j-impl v2.8.2
Url: https://logging.apache.org/log4j/2.x/index.html
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: log4j-slf4j-impl v2.9.1
Url: https://logging.apache.org/log4j/2.x/index.html
License: Apache-2.0

Apache Log4j
Copyright 1999-2017 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Dumbster SMTP test server
Copyright 2004 Jason Paul Kitchen

TypeUtil.java
Copyright 2002-2012 Ramnivas Laddad, Juergen Hoeller, Chris Beams

picocli (http://picocli.info)
Copyright 2017 Remko Popma

Library: commons-compiler v3.0.8
Url: https://github.com/janino-compiler/janino
License: BSD-3-Clause # Question: tweaked license on bullet point 3

Janino - An embedded Java[TM] compiler

Copyright (c) 2001-2016, Arno Unkrig
Copyright (c) 2015-2016 TIBCO Software Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above
   copyright notice, this list of conditions and the following
   disclaimer in the documentation and/or other materials
   provided with the distribution.
3. Neither the name of JANINO nor the names of its contributors
   may be used to endorse or promote products derived from this
   software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER
IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: janino v3.0.8

Url: https://github.com/janino-compiler/janino
License: BSD-3-Clause # Question: tweaked license on bullet point 3

Janino - An embedded Java[TM] compiler

Copyright (c) 2001-2016, Arno Unkrig
Copyright (c) 2015-2016 TIBCO Software Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above
   copyright notice, this list of conditions and the following
   disclaimer in the documentation and/or other materials
   provided with the distribution.
3. Neither the name of JANINO nor the names of its contributors
   may be used to endorse or promote products derived from this
   software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER
IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR
OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN
IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: org.eclipse.core.commands v3.6.0
Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a)

promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.core.contenttype v3.4.100
Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special,

incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is

solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.core.expressions v3.4.300
Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide,

royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows

subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.core.filesystem v1.3.100
Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the

right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.core.jobs v3.5.100
Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale

of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any

other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and

unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.core.resources v3.7.100

Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such

addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.core.runtime v3.7.0
Url: http://www.eclipse.org
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided

that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or

offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable

separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: org.eclipse.equinox.app v1.3.100
Url: http://www.eclipse.org/platform
License: EPL-1.0

Copyright (c) 2012 Eclipse Foundation.
All rights reserved. This program and the accompanying materials
are made available under the terms of the Eclipse Distribution License v1.0
which accompanies this distribution, and is available at
http://www.eclipse.org/org/documents/edl-v10.php

Library: org.eclipse.equinox.common v3.6.0
Url: http://www.eclipse.org/platform
License: EPL-1.0

Copyright (c) 2012 Eclipse Foundation.
All rights reserved. This program and the accompanying materials
are made available under the terms of the Eclipse Distribution License v1.0
which accompanies this distribution, and is available at
http://www.eclipse.org/org/documents/edl-v10.php

Library: org.eclipse.equinox.preferences v3.4.1
Url: http://www.eclipse.org/platform
License: EPL-1.0

Copyright (c) 2012 Eclipse Foundation.
All rights reserved. This program and the accompanying materials
are made available under the terms of the Eclipse Distribution License v1.0
which accompanies this distribution, and is available at
http://www.eclipse.org/org/documents/edl-v10.php

Library: org.eclipse.equinox.registry v3.5.101
Url: http://www.eclipse.org/platform

License: EPL-1.0

Copyright (c) 2012 Eclipse Foundation.
All rights reserved. This program and the accompanying materials
are made available under the terms of the Eclipse Distribution License v1.0
which accompanies this distribution, and is available at
http://www.eclipse.org/org/documents/edl-v10.php

Library: org.eclipse.jdt.core v3.10.0
Url: http://www.eclipse.org/jdt
License: EPL-1.0

Copyright (c) 2012, 2016 Eclipse Foundation and others.
All rights reserved. This program and the accompanying materials
are made available under the terms of the Eclipse Distribution License v1.0
which accompanies this distribution, and is available at
http://www.eclipse.org/org/documents/edl-v10.php

Library: org.eclipse.osgi v3.7.1
Url: http://www.eclipse.org/jdt
License: EPL-1.0

Copyright (c) 2012, 2017 Eclipse Foundation.
All rights reserved. This program and the accompanying materials
are made available under the terms of the Eclipse Distribution License v1.0
which accompanies this distribution, and is available at
http://www.eclipse.org/org/documents/edl-v10.php

Library: org.eclipse.text v3.5.101
Url: http://www.eclipse.org/jdt
License: EPL-1.0

Copyright Â© Eclipse Foundation, Inc. All Rights Reserved.

Eclipse Public License - v 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That

Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement

Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

Library: jruby-complete v9.1.13.0
Url: http://jruby.org/
License: EPL-2.0

JRuby is Copyright (c) 2007-2018 The JRuby project

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition

of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in

writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the

minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A
is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to
look for such a notice.

You may add additional accurate notices of copyright ownership.

Library: slf4j-api v1.7.25
Url: http://www.slf4j.org/
License: MIT

Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: paquet v0.2.1
Url: https://github.com/elastic/logstash
License: Apache-2.0

Copyright (c) 2012â2016 Elasticsearch <http://www.elastic.co>

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: pleaserun v0.0.30
Url: https://github.com/jordansissel/pleaserun
License: Apache-2.0

Copyright 2014 Jordan Sissel contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: polyglot v0.3.5
Url: http://github.com/cjheath/polyglot
License: MIT

Copyright (c) 2007 Clifford Heath

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: poseidon v0.0.5
Url: https://github.com/bpot/poseidon
License: MIT

Copyright (c) 2013 Bob Potter

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: prototype.js v
Url: http://prototypejs.org
License: MIT TODO: Another jruby dependency that is not shipped with it

TODO

Library: pry v0.10.4
Url: http://pryrepl.org
License: MIT

Copyright (c) 2016 John Mair (banisterfiend)

Permission is hereby granted, free of charge, to any person obtaining

a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: psych v2.2.4
Url: https://github.com/ruby/psych
License: MIT

Copyright 2009 Aaron Patterson, et al.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: public_suffix v1.4.6
Url: https://simonecarletti.com/code/publicsuffix-ruby
License: MIT

Copyright (c) 2009-2018 Simone Carletti <weppos@weppos.net>

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: puma v2.16.0
Url: http://puma.io
License: BSD-3-Clause

Some code copyright (c) 2005, Zed Shaw
Copyright (c) 2011, Evan Phoenix
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
 list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice
 this list of conditions and the following disclaimer in the documentation
 and/or other materials provided with the distribution.
* Neither the name of the Evan Phoenix nor the names of its contributors
 may be used to endorse or promote products derived from this software
 without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER

CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Library: racc v
Url: https://github.com/tenderlove/racc
License: TODO: https://github.com/tenderlove/racc/blob/master/COPYING vs https://github.com/jruby/jruby/blob/master/COPYING#L89

TODO

Racc is distributed under the terms of the GNU Lesser General Public License version 2. Note that you do NOT need to follow LGPL for your own parser (racc outputs). You can distribute those files under any licenses you want.Racc is

---

Library: rack-protection v1.5.5
Url: http://github.com/rkh/rack-protection
License: MIT

The MIT License (MIT)

Copyright (c) 2011-2017 Konstantin Haase
Copyright (c) 2015-2017 Zachary Scott

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rack v1.6.6
Url: http://rack.github.io/
License: MIT

The MIT License (MIT)

Copyright (C) 2007-2018 Christian Neukirchen <http://chneukirchen.org/infopage.html>

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER
IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rake v10.4.2
Url: https://github.com/ruby/rake
License: MIT

Copyright (c) Jim Weirich

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE

LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: redis v3.3.5
Url: https://github.com/redis/redis-rb
License: MIT

Copyright (c) 2009 Ezra Zygmuntowicz

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rspec-core v3.7.1
Url: https://github.com/rspec/rspec-core
License: MIT

The MIT License (MIT)
==========

* Copyright Â© 2012 Chad Humphries, David Chelimsky, Myron Marston
* Copyright Â© 2009 Chad Humphries, David Chelimsky
* Copyright Â© 2006 David Chelimsky, The RSpec Development Team
* Copyright Â© 2005 Steven Baker

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish,

distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rspec-expectations v3.7.0
Url: https://github.com/rspec/rspec-expectations
License: MIT

The MIT License (MIT)
==========

* Copyright Â© 2012 David Chelimsky, Myron Marston
* Copyright Â© 2006 David Chelimsky, The RSpec Development Team
* Copyright Â© 2005 Steven Baker

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rspec-mocks v3.7.0

Url: https://github.com/rspec/rspec-mocks
License: MIT

The MIT License (MIT)
==========

* Copyright Â© 2012 David Chelimsky, Myron Marston
* Copyright Â© 2006 David Chelimsky, The RSpec Development Team
* Copyright Â© 2005 Steven Baker

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rspec-sequencing v0.1.0
Url: https://github.com/elastic/rspec-sequencing
License: Apache-2.0

Copyright (c) 2016 Elasticsearch <http://www.elastic.co>

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: rspec-support v3.7.1
Url: https://github.com/rspec/rspec-support

License: MIT

* Copyright Â© 2013 David Chelimsky, Myron Marston, Jon Rowe, Sam Phippen, Xavier Shay, Bradley Schaefer

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rspec-wait v0.0.9
Url: https://github.com/laserlemon/rspec-wait
License: MIT

Copyright (c) 2014 Steve Richert

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION

OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rspec v3.7.0
Url: https://github.com/rspec/rspec
License: MIT

Copyright Â© 2009 Chad Humphries, David Chelimsky
Copyright Â© 2006 David Chelimsky, The RSpec Development Team
Copyright Â© 2005 Steven Baker

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the âSoftwareâ), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED âAS ISâ, WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Library: ruby-progressbar v1.8.3
Url: https://github.com/jfelchner/ruby-progressbar
License: MIT

Copyright (c) 2010-2016 The Kompanee, Ltd

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in

all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: rubyzip v1.2.1
Url: https://github.com/rubyzip/rubyzip
License: BSD-2-Clause-FreeBSD

Authors:

* Alexander Simonov (alex at simonov.me)
* Alan Harper (alan at aussiegeek.net)
* Thomas Sondergaard (thomas at sondergaard.cc)
* Technorama Ltd. (oss-ruby-zip at technorama.net)
* extra-field support contributed by Tatsuki Sugiura (sugi at nemui.org)

Rubyzip is distributed under the same license as ruby. See http://www.ruby-lang.org/en/LICENSE.txt:

Ruby is copyrighted free software by Yukihiro Matsumoto <matz@netlab.jp>.
You can redistribute it and/or modify it under either the terms of the
2-clause BSDL (see the file BSDL), or the conditions below:

1. You may make and give away verbatim copies of the source form of the software without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.

2. You may modify your copy of the software in any way, provided that you do at least ONE of the following:

   a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or by allowing the author to include your modifications in the software.

   b) use the modified software only within your corporation or organization.

   c) give non-standard binaries non-standard names, with instructions on where to get the original software distribution.

   d) make other distribution arrangements with the author.

3. You may distribute the software in object code or binary form,
provided that you do at least ONE of the following:

a) distribute the binaries and library files of the software,
together with instructions (in the manual page or equivalent)
on where to get the original distribution.

b) accompany the distribution with the machine-readable source of
the software.

c) give non-standard binaries non-standard names, with
instructions on where to get the original software distribution.

d) make other distribution arrangements with the author.

4. You may modify and include the part of the software into any other
software (possibly commercial). But some files in the distribution
are not written by the author, so that they are not under these terms.

For the list of those files and their copying conditions, see the
file LEGAL.

5. The scripts and library files supplied as input to or produced as
output from the software do not automatically fall under the
copyright of the software, but belong to whomever generated them,
and may be sold commercially, and may be aggregated with this
software.

6. THIS SOFTWARE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE.

Library: rumbster v1.1.0
Url: https://github.com/aesterline/rumbster
License: Apache-2.0

Copyright (c) 2011 Adam Esterline. See LICENSE.txt for further details.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: safe_yaml v1.0.4
Url: https://github.com/dtao/safe_yaml
License: MIT

Copyright (c) 2013 Dan Tao

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: sequel v5.7.1
Url: https://github.com/jeremyevans/sequel
License: MIT

Copyright (c) 2007-2008 Sharon Rosner
Copyright (c) 2008-2018 Jeremy Evans

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal in the Software without restriction, including without limitation the
rights to use, copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: simple_oauth v0.3.1
Url: https://github.com/laserlemon/simple_oauth
License: MIT

Copyright (c) 2010-2013 Steve Richert, Erik Michaels-Ober

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: sinatra v1.4.8
Url: https://github.com/sinatra/sinatra
License: MIT

Copyright (c) 2007, 2008, 2009 Blake Mizerany
Copyright (c) 2010, 2011, 2012, 2013, 2014, 2015, 2016 Konstantin Haase

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,

copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES
OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT
HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR
OTHER DEALINGS IN THE SOFTWARE.

Library: slop v3.6.0
Url: https://github.com/leejarvis/slop
License: MIT

Copyright (c) 2012 Lee Jarvis

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: snakeyaml v1.18
Url: https://bitbucket.org/asomov/snakeyaml/
License: Apache-2.0

Copyright (c) 2008, http://www.snakeyaml.org

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: snappy-jars v1.1.0.1.2
Url: https://github.com/doxavore/snappy-jars
License: Apache-2.0

Copyright Doug Mayer (https://github.com/doxavore)

README.md: "Per the real implementation, Apache License v2.0."

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: snappy v0.0.12
Url: https://github.com/miyucy/snappy
License: MIT

Copyright (c) 2011-2013 miyucy

MIT License

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,

distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: snmp v1.2.0
Url: https://github.com/hallidave/ruby-snmp
License: MIT

Copyright (c) 2004-2014 David R. Halliday

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
THE SOFTWARE.

Library: spoon v0.0.6
Url: https://github.com/headius/spoon
License: Apache-2.0

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking

systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum

to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Library: stud v0.0.23

Url: https://github.com/jordansissel/ruby-stud
License: Apache-2.0

Copyright 2012-2013 Jordan Sissel and contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: thread_safe v0.3.6
Url: https://github.com/ruby-concurrency/thread_safe
License: Apache-2.0

Apache License

Version 2.0, January 2004

http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as
defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that
is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that
control, are controlled by, or are under common control with that entity. For the purposes
of this definition, "control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or otherwise, or (ii) ownership
of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of
such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by
this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall

terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for

determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Library: tilt v2.0.8
Url: https://github.com/rtomayko/tilt
License: MIT

Copyright (c) 2010-2016 Ryan Tomayko <http://tomayko.com/about>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: timecop v0.9.1
Url: https://github.com/travisjeffery/timecop
License: MIT

(The MIT License)

Copyright (c) 2012 â Travis Jeffery, John Trupiano

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
'Software'), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: treetop v1.4.15
Url: https://github.com/cjheath/treetop
License: MIT

Copyright (c) 2007 Nathan Sobo.

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: twitter v5.15.0
Url: https://github.com/sferik/twitter
License: MIT

Copyright (c) 2006-2015 Erik Michaels-Ober, John Nunemaker, Wynn Netherland, Steve Richert, Steve Agalloco

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: tzinfo-data v1.2018.4
Url: https://github.com/tzinfo/tzinfo-data
License: MIT

Copyright (c) 2005-2018 Philip Ross

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: tzinfo v1.2.5
Url: https://github.com/tzinfo/tzinfo
License: MIT

Copyright (c) 2005-2018 Philip Ross

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: unf v0.1.4
Url: https://github.com/knu/ruby-unf
License: BSD-2-Clause

Copyright (c) 2011, 2012 Akinori MUSHA

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the

documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Library: unsafe-fences v1.0
Url: https://github.com/headius/unsafe-fences
License: Apache-2.0

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: webhdfs v0.8.0
Url: https://github.com/kzk/webhdfs
License: Apache-2.0

Copyright (C) 2012 Fluentd Project

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Library: webmock v1.21.0
Url: https://github.com/bblimke/webmock
License: MIT

Copyright (c) 2009-2010 Bartosz Blimke

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Library: webrick v1.3.1
Url: https://github.com/ruby/webrick
License: BSD-2-Clause

Copyright (C) 1993-2013 Yukihiro Matsumoto. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

Library: xml-simple v1.1.5
Url: https://github.com/maik/xml-simple
License: BSD-2-Clause

Readme.md:

XmlSimple is Copyright (c) 2003-2017 Maik Schmidt. It is free software, and may
be redistributed under the terms specified in the COPYING file of the current
Ruby distribution.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
jruby,9.1.14.0,https://jruby.org/,EPL-1.0|GPL-2.0|LGPL-2.1
bytelist,,http://github.com/jruby/bytelist,EPL-1.0|GPL-2.0|LGPL-2.1
jnr-posix,,https://github.com/jnr/jnr-posix,EPL-1.0|GPL-2.0|LGPL-2.1
jruby-openssl,,https://github.com/jruby/jruby-openssl,EPL-1.0|GPL-2.0|LGPL-2.1
jruby-readline,,https://github.com/jruby/jruby-readline,EPL-1.0|GPL-2.0|LGPL-2.1
psych,,https://github.com/ruby/psych),MIT
jay-yydebug,,http://svn.codehaus.org/jruby/trunk/jay/yydebug,EPL-1.0|GPL-2.0|LGPL-2.0
control.js,,,MIT
dragdrop.js,,,MIT
effect.js,,,MIT
prototype.js,,,MIT
asm,,http://asm.objectweb.org,BSD
jline2,,https://github.com/jline/jline2,BSD

jzlib,,https://www.jcraft.com/jzlib/,BSD-3-Clause
rake,,https://github.com/ruby/rake,MIT
jcodings,,http://github.com/jruby/jcodings,MIT
joni,,http://github.com/jruby/joni,MIT
bouncycastle,,http://www.bouncycastle.org,MIT
jnr-x86asm,,https://github.com/jnr/jnr-x86asm,MIT
invokebinder,,https://github.com/headius/invokebinder,Apache-2.0
jffi,,https://github.com/jnr/jffi,Apache-2.0
jitescript,,https://github.com/qmx/jitescript,Apache-2.0
jnr-constants,,http://github.com/jnr/jnr-constants,Apache-2.0
jnr-enxio,,https://github.com/jnr/jnr-enxio,Apache-2.0
jnr-ffi,,https://github.com/jnr/jnr-jffi,Apache-2.0
jnr-netdb,,http://github.com/jnr/jnr-netdb,Apache-2.0
jnr-unixsocket,,https://github.com/jnr/jnr-unixsocket,Apache-2.0
joda-time,,http://joda-time.sourceforge.net,Apache-2.0
maven,,http://maven.apache.org/,Apache-2.0
nailgun,,http://martiansoftware.com/nailgun,Apache-2.0
options,,https://github.com/headius/options,Apache-2.0
snakeyaml,,https://github.com/asomov/snakeyaml,Apache-2.0
unsafe-fences,,https://github.com/headius/unsafe-fences,Apache-2.0
racc,,https://github.com/tenderlove/racc,Ruby
json-generator,,https://github.com/flori/json,Ruby
json-parser,,https://github.com/flori/json,Ruby

1.19 mongodb-org-shell 4.0.5

1.19.1 Available under license :

Mozilla Public License Version 2.0
==========

1. Definitions

1.1. "Contributor"
means each individual or legal entity that creates, contributes to
the creation of, or owns Covered Software.

1.2. "Contributor Version"
means the combination of the Contributions of others (if any) used
by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"
means Covered Software of a particular Contributor.

1.4. "Covered Software"
means Source Code Form to which the initial Contributor has attached
the notice in Exhibit A, the Executable Form of such Source Code

Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"
means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"
means any form of the work other than Source Code Form.

1.7. "Larger Work"
means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"
means this document.

1.9. "Licensable"
means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"
means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor
means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"
means either the GNU General Public License, Version 2.0, the GNU

Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"
means the form of the work preferred for making modifications.

1.14. "You" (or "Your")
means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source

Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by

You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
************************************************************************
*                                                                      *
*  6. Disclaimer of Warranty                                           *
*  -------------------------                                           *
*                                                                      *
*  Covered Software is provided under this License on an "as is"       *
*  basis, without warranty of any kind, either expressed, implied, or  *
*  statutory, including, without limitation, warranties that the       *
*  Covered Software is free of defects, merchantable, fit for a        *
*  particular purpose or non-infringing. The entire risk as to the     *
*  quality and performance of the Covered Software is with You.        *
*  Should any Covered Software prove defective in any respect, You     *
*  (not any Contributor) assume the cost of any necessary servicing,   *
*  repair, or correction. This disclaimer of warranty constitutes an   *
*  essential part of this License. No use of any Covered Software is   *
*  authorized under this License except under this disclaimer.         *
*                                                                      *
************************************************************************

************************************************************************
*                                                                      *
*  7. Limitation of Liability                                          *
*  --------------------------                                          *
*                                                                      *
*  Under no circumstances and under no legal theory, whether tort      *
*  (including negligence), contract, or otherwise, shall any           *
*  Contributor, or anyone who distributes Covered Software as          *
*  permitted above, be liable to You for any direct, indirect,         *
*  special, incidental, or consequential damages of any character      *
*  including, without limitation, damages for lost profits, loss of    *
*  goodwill, work stoppage, computer failure or malfunction, or any    *
*  and all other commercial damages or losses, even if such party      *
*  shall have been informed of the possibility of such damages. This   *
*  limitation of liability shall not apply to liability for death or   *
*  personal injury resulting from such party's negligence to the       *
*  extent applicable law prohibits such limitation. Some               *
*  jurisdictions do not allow the exclusion or limitation of           *
*  incidental or consequential damages, so this exclusion and          *
*  limitation may not apply to You.                                    *
*                                                                      *
************************************************************************
```

8. Litigation

Any litigation relating to this License may be brought only in the
courts of a jurisdiction where the defendant maintains its principal
place of business and such litigation shall be governed by laws of that

jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.

If it is not possible or desirable to put the notice in a particular
file, then You may include the notice in a location (such as a LICENSE
file in a relevant directory) where a recipient would be likely to look
for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as
defined by the Mozilla Public License, v. 2.0.
MongoDB uses third-party libraries or other resources that may
be distributed under licenses different than the MongoDB software.

In the event that we accidentally failed to list a required notice,
please bring it to our attention through any of the ways detailed here :

mongodb-dev@googlegroups.com

The attached notices are provided for information only.

For any licenses that require disclosure of source, sources are available at
https://github.com/mongodb/mongo.

1) License Notice for Boost
---

http://www.boost.org/LICENSE_1_0.txt

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization
obtaining a copy of the software and accompanying documentation covered by
this license (the "Software") to use, reproduce, display, distribute,
execute, and transmit the Software, and to prepare derivative works of the
Software, and to permit third-parties to whom the Software is furnished to
do so, all subject to the following:

The copyright notices in the Software and this entire statement, including
the above license grant, this restriction and the following disclaimer,
must be included in all copies of the Software, in whole or in part, and

all derivative works of the Software, unless such copies or derivative
works are solely in the form of machine-executable object code generated by
a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

3) License Notice for PCRE
---

http://www.pcre.org/licence.txt

PCRE LICENCE

PCRE is a library of functions to support regular expressions whose syntax
and semantics are as close as possible to those of the Perl 5 language.

Release 7 of PCRE is distributed under the terms of the "BSD" licence, as
specified below. The documentation for PCRE, supplied in the "doc"
directory, is distributed under the same terms as the software itself.

The basic library functions are written in C and are freestanding. Also
included in the distribution is a set of C++ wrapper functions.

THE BASIC LIBRARY FUNCTIONS

Written by: Philip Hazel
Email local part: ph10
Email domain: cam.ac.uk

University of Cambridge Computing Service,
Cambridge, England.

Copyright (c) 1997-2008 University of Cambridge
All rights reserved.

THE C++ WRAPPER FUNCTIONS

Contributed by: Google Inc.

Copyright (c) 2007-2008, Google Inc.
All rights reserved.

THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google Inc. nor the names of their contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

4) License notice for Aladdin MD5

Copyright (C) 1999, 2002 Aladdin Enterprises. All rights reserved.

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,

including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
  claim that you wrote the original software. If you use this software
  in a product, an acknowledgment in the product documentation would be
  appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
  misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

L. Peter Deutsch
ghost@aladdin.com

5) License notice for Snappy - http://code.google.com/p/snappy/

Copyright 2005 and onwards Google Inc.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

A light-weight compression algorithm. It is designed for speed of
compression and decompression, rather than for the utmost in space
savings.

For getting better compression ratios when you are compressing data with long repeated sequences or compressing data that is similar to other data, while still compressing fast, you might look at first using BMDiff and then compressing the output of BMDiff with Snappy.

6) License notice for Google Perftools (TCMalloc utility)

New BSD License

Copyright (c) 1998-2006, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7) License notice for Linenoise

Copyright (c) 2010, Salvatore Sanfilippo <antirez at gmail dot com>
Copyright (c) 2010, Pieter Noordhuis <pcnoordhuis at gmail dot com>

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of Redis nor the names of its contributors may be used
to endorse or promote products derived from this software without
specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

8) License notice for S2 Geometry Library

Copyright 2005 Google Inc. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

9) License notice for MurmurHash

Copyright (c) 2010-2012 Austin Appleby

Permission is hereby granted, free of charge, to any person obtaining a copy

of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

10) License notice for Snowball
Copyright (c) 2001, Dr Martin Porter
All rights reserved.

THE "BSD" LICENCE

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

* Neither the name of the University of Cambridge nor the name of Google
Inc. nor the names of their contributors may be used to endorse or
promote products derived from this software without specific prior
written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

11) License notice for yaml-cpp

Copyright (c) 2008 Jesse Beder.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

12) License notice for zlib
---

http://www.zlib.net/zlib_license.html

zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.8, April 28th, 2013

Copyright (C) 1995-2013 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be
  misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

13) License notice for 3rd party software included in the WiredTiger library
---

http://source.wiredtiger.com/license.html

WiredTiger Distribution Files	Copyright Holder	License
src/include/bitstring.i	University of California, Berkeley	BSD-3-Clause License
src/include/queue.h	University of California, Berkeley	BSD-3-Clause License
src/os_posix/os_getopt.c	University of California, Berkeley	BSD-3-Clause License
src/support/hash_city.c	Google, Inc.	The MIT License
src/support/hash_fnv.c	Authors	Public Domain

Other optional 3rd party software included in the WiredTiger distribution is removed by MongoDB.

BSD-3-CLAUSE LICENSE
---

http://www.opensource.org/licenses/BSD-3-Clause

Copyright (c) 1987, 1989, 1991, 1993, 1994
  The Regents of the University of California. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.
4. Neither the name of the University nor the names of its contributors
  may be used to endorse or promote products derived from this software
  without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE MIT LICENSE
---

http://www.opensource.org/licenses/MIT

Copyright (c) 2011 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

14) License Notice for SpiderMonkey

SpiderMonkey Distribution Files	Copyright Holder	License
js/src/jit/shared/AssemblerBuffer-x86-shared.h	Apple, Inc	BSD-2-Clause
js/src/jit/shared/BaseAssembler-x86-shared.h		
js/src/builtin/	Google, Inc	BSD-3-Clause
js/src/irregexp/		
js/src/jit/arm/		

js/src/jit/mips/		
mfbt/double-conversion/		
intl/icu/source/common/unicode/	IBM, Inc	ICU
js/src/asmjs/	Mozilla, Inc	Apache2
js/public/	Mozilla, Inc	MPL2
js/src/		
mfbt		
js/src/vm/Unicode.cpp	None	Public Domain
mfbt/lz4.c	Yann Collet	BSD-2-Clause
mfbt/lz4.h		

Other optional 3rd party software included in the SpiderMonkey distribution is removed by MongoDB.

Apple, Inc: BSD-2-Clause

Copyright (C) 2008 Apple Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
1. Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
  notice, this list of conditions and the following disclaimer in the
  documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY APPLE INC. ``AS IS'' AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL APPLE INC. OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY
OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Google, Inc: BSD-3-Clause

Copyright 2012 the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.
* Neither the name of Google Inc. nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2012 International Business Machines Corporation and
others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the "Software"),
to deal in the Software without restriction, including without limitation
the rights to use, copy, modify, merge, publish, distribute, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, provided that the above copyright notice(s) and this
permission notice appear in all copies of the Software and that both the
above copyright notice(s) and this permission notice appear in supporting

documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall
not be used in advertising or otherwise to promote the sale, use or other
dealings in this Software without prior written authorization of the
copyright holder.

All trademarks and registered trademarks mentioned herein are the property
of their respective owners.

Mozilla, Inc: Apache 2

Copyright 2014 Mozilla Foundation

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Mozilla, Inc: MPL 2

Copyright 2014 Mozilla Foundation

This Source Code Form is subject to the terms of the Mozilla Public
License, v. 2.0. If a copy of the MPL was not distributed with this
file, You can obtain one at http://mozilla.org/MPL/2.0/.

Public Domain

Any copyright is dedicated to the Public Domain.
http://creativecommons.org/licenses/publicdomain/

LZ4: BSD-2-Clause

Copyright (C) 2011-2014, Yann Collet.
BSD 2-Clause License (http://www.opensource.org/licenses/bsd-license.php)

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

You can contact the author at :
- LZ4 source repository : http://code.google.com/p/lz4/
- LZ4 public forum : https://groups.google.com/forum/#!forum/lz4c

15) License Notice for Intel DFP Math Library

Copyright (c) 2011, Intel Corp.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, his list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

16) License Notice for Unicode Data

Copyright 1991-2015 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in
http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that
(a) this copyright and permission notice appear with all copies
of the Data Files or Software,
(b) this copyright and permission notice appear in associated
documentation, and
(c) there is clear notice in each modified Data File or in the Software
as well as in the documentation associated with the Data File(s) or
Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

17) License Notice for Valgrind.h

Notice that the following BSD-style license applies to this one file (valgrind.h) only. The rest of Valgrind is licensed under the terms of the GNU General Public License, version 2, unless otherwise indicated. See the COPYING file in the source distribution for details.

This file is part of Valgrind, a dynamic binary instrumentation framework.

Copyright (C) 2000-2015 Julian Seward. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.

3. Altered source versions must be plainly marked as such, and must

not be misrepresented as being the original software.

4. The name of the author may not be used to endorse or promote
products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS
OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Notice that the above BSD-style license applies to this one file
(valgrind.h) only. The entire rest of Valgrind is licensed under
the terms of the GNU General Public License, version 2. See the
COPYING file in the source distribution for details.

---

18) License notice for ICU4C

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
"Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, and/or sell copies of the Software, and to permit persons
to whom the Software is furnished to do so, provided that the above
copyright notice(s) and this permission notice appear in all copies of
the Software and that both the above copyright notice(s) and this
permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

Third-Party Software Licenses

This section contains third-party software notices and/or additional terms for licensed third-party software components included within ICU libraries.

1. Unicode Data Files and Software

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2016 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in
http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that
(a) this copyright and permission notice appear with all copies of the Data Files or Software,
(b) this copyright and permission notice appear in associated documentation, and
(c) there is clear notice in each modified Data File or in the Software

as well as in the documentation associated with the Data File(s) or
Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

2. Chinese/Japanese Word Break Dictionary Data (cjdict.txt)

```
#     The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
#  The BSD License
#  http://opensource.org/licenses/bsd-license.php
#  Copyright (C) 2006-2008, Google Inc.
#
#  All rights reserved.
#
#  Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
#
#  Redistributions of source code must retain the above copyright notice,
# this list of conditions and the following disclaimer.
#  Redistributions in binary form must reproduce the above
# copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided with
# the distribution.
#  Neither the name of  Google Inc. nor the names of its
# contributors may be used to endorse or promote products derived from
# this software without specific prior written permission.
#
#
#  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
```

MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#
#
The word list in cjdict.txt are generated by combining three word lists
listed below with further processing for compound word breaking. The
frequency is generated with an iterative training against Google web
corpora.
#
* Libtabe (Chinese)
- https://sourceforge.net/project/?group_id=1519
- Its license terms and conditions are shown below.
#
* IPADIC (Japanese)
- http://chasen.aist-nara.ac.jp/chasen/distribution.html
- Its license terms and conditions are shown below.
#
---------COPYING.libtabe ---- BEGIN--------------------
#
/*
* Copyrighy (c) 1999 TaBE Project.
* Copyright (c) 1999 Pai-Hsiang Hsiao.
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* . Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* . Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in
*   the documentation and/or other materials provided with the
*   distribution.
* . Neither the name of the TaBE Project nor the names of its
*   contributors may be used to endorse or promote products derived
*   from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS

```
#   * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
#   * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
#   * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
#   * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
#   * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
#   * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
#   * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
#   * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
#   * OF THE POSSIBILITY OF SUCH DAMAGE.
#   */
#
#  /*
#   * Copyright (c) 1999 Computer Systems and Communication Lab,
#   *                  Institute of Information Science, Academia
#      *                  Sinica. All rights reserved.
#   *
#   * Redistribution and use in source and binary forms, with or without
#   * modification, are permitted provided that the following conditions
#   * are met:
#   *
#   * . Redistributions of source code must retain the above copyright
#   *   notice, this list of conditions and the following disclaimer.
#   * . Redistributions in binary form must reproduce the above copyright
#   *   notice, this list of conditions and the following disclaimer in
#   *   the documentation and/or other materials provided with the
#   *   distribution.
#   * . Neither the name of the Computer Systems and Communication Lab
#   *   nor the names of its contributors may be used to endorse or
#   *   promote products derived from this software without specific
#   *   prior written permission.
#   *
#   * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
#   * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
#   * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
#   * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
#   * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
#   * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
#   * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
#   * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
#   * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
#   * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
#   * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
#   * OF THE POSSIBILITY OF SUCH DAMAGE.
#   */
#
#  Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
#     University of Illinois
#  c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4
```

```
#
# ---------------COPYING.libtabe-----END-------------------------------
#
#
# ---------------COPYING.ipadic-----BEGIN------------------------------
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology.  All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software.  The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
#
# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
# to users as so produced on an experimental basis.  Accordingly, the
# program is provided without any warranty whatsoever, whether express,
# implied, statutory or otherwise.  The term "warranty" used herein
# includes, but is not limited to, any warranty of the quality,
# performance, merchantability and fitness for a particular purpose of
# the program and the nonexistence of any infringement or violation of
# any right of any third party.
#
```

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.
#
Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
"use" as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.
#
In the case where the program, whether in its original form or
modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.
#
----------COPYING.ipadic-----END----------

3. Lao Word Break Dictionary Data (laodict.txt)

Copyright (c) 2013 International Business Machines Corporation
and others. All Rights Reserved.
#
Project: http://code.google.com/p/lao-dictionary/
Dictionary: http://lao-dictionary.googlecode.com/git/Lao-Dictionary.txt
License: http://lao-dictionary.googlecode.com/git/Lao-Dictionary-LICENSE.txt
(copied below)
#
This file is derived from the above dictionary, with slight
modifications.

Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
All rights reserved.
#
Redistribution and use in source and binary forms, with or without

modification,
are permitted provided that the following conditions are met:
#
#
Redistributions of source code must retain the above copyright notice, this
list of conditions and the following disclaimer. Redistributions in
binary form must reproduce the above copyright notice, this list of
conditions and the following disclaimer in the documentation and/or
other materials provided with the distribution.
#
#
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
OF THE POSSIBILITY OF SUCH DAMAGE.
----

4. Burmese Word Break Dictionary Data (burmesedict.txt)

Copyright (c) 2014 International Business Machines Corporation
and others. All Rights Reserved.
#
This list is part of a project hosted at:
github.com/kanyawtech/myanmar-karen-word-lists
#

Copyright (c) 2013, LeRoy Benjamin Sharon
All rights reserved.
#
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met: Redistributions of source code must retain the above
copyright notice, this list of conditions and the following
disclaimer. Redistributions in binary form must reproduce the
above copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.
#
Neither the name Myanmar Karen Word Lists, nor the names of its
contributors may be used to endorse or promote products derived

from this software without specific prior written permission.
#
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.
---

5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone
Database for its time zone support. The ownership of the TZ database
is explained in BCP 175: Procedure for Maintaining the Time Zone
Database section 7.

7. Database Ownership
#
The TZ database itself is not an IETF Contribution or an IETF
document. Rather it is a pre-existing and regularly updated work
that is in the public domain, and is intended to remain in the
public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do
not apply to the TZ Database or contributions that individuals make
to it. Should any claims be made and substantiated against the TZ
Database, the organization that is providing the IANA
Considerations defined in this RFC, under the memorandum of
understanding with the IETF, currently ICANN, may act in accordance
with all competent court orders. No ownership claims will be made
by ICANN or the IETF Trust on the database or the code. Any person
making a contribution to the database or code waives all rights to
future claims in that contribution or in the TZ Database.

End

19) License notice for timelib

The MIT License (MIT)

Copyright (c) 2015-2017 Derick Rethans

Copyright (c) 2017 MongoDB, Inc

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

20) License notice for windows dirent implementation
---

* Dirent interface for Microsoft Visual Studio
* Version 1.21
*
* Copyright (C) 2006-2012 Toni Ronkko
* This file is part of dirent. Dirent may be freely distributed
* under the MIT license. For all details and documentation, see
* https://github.com/tronkko/dirent

Server Side Public License
VERSION 1, OCTOBER 16, 2018

Copyright 2018 MongoDB, Inc.

Everyone is permitted to copy and distribute verbatim copies of this
license document, but changing it is not allowed.

TERMS AND CONDITIONS

0. Definitions.

This License refers to Server Side Public License.

Copyright also means copyright-like laws that apply to other kinds of
works, such as semiconductor masks.

The Program refers to any copyrightable work licensed under this

License. Each licensee is addressed as you. Licensees and recipients may be individuals or organizations.

To modify a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a modified version of the earlier work or a work based on the earlier work.

A covered work means either the unmodified Program or a work based on the Program.

To propagate a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To convey a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays Appropriate Legal Notices to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The source code for a work means the preferred form of the work for making modifications to it. Object code means any non-source form of a work.

A Standard Interface means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language. The System Libraries of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A Major Component, in this context, means a major essential component (kernel, window system, and so on) of the

specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The Corresponding Source for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program, subject to section 13. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law. Subject to section 13, you may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program. You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to keep intact all notices.

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an aggregate if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on

a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A User Product is either (1) a consumer product, which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, normally used refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

Installation Information for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

Additional permissions are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions. When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered further restrictions within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms. Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An entity transaction is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A contributor is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's contributor version.

A contributor's essential patent claims are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, control includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a patent license is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To grant such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. Knowingly relying means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is discriminatory if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of

distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot use, propagate or convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not use, propagate or convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Offering the Program as a Service.

If you make the functionality of the Program or a modified version available to third parties as a service, you must make the Service Source Code available via network download to everyone at no charge, under the terms of this License. Making the functionality of the Program or modified version available to third parties as a service includes, without limitation, enabling third parties to interact with the functionality of the Program or modified version remotely through a computer network, offering a service the value of which entirely or primarily derives from the value of the Program or modified version, or offering a service that accomplishes for users the primary purpose of the Program or modified version.

Service Source Code means the Corresponding Source for the Program or the modified version, and the Corresponding Source for all programs that you use to make the Program or modified version available as a service, including, without limitation, management software, user interfaces, application program interfaces, automation software, monitoring software, backup software, storage software and hosting software, all such that a user could run an instance of the service using the Service Source Code

you make available.

14. Revised Versions of this License.

MongoDB, Inc. may publish revised and/or new versions of the Server Side Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the Server Side Public License or any later version applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by MongoDB, Inc. If the Program does not specify a version number of the Server Side Public License, you may choose any version ever published by MongoDB, Inc.

If the Program specifies that a proxy can decide which future versions of the Server Side Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM AS IS WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above
cannot be given local legal effect according to their terms, reviewing
courts shall apply local law that most closely approximates an absolute
waiver of all civil liability in connection with the Program, unless a
warranty or assumption of liability accompanies a copy of the Program in
return for a fee.

END OF TERMS AND CONDITIONS

1.20 MySQL Community Server 5.7 5.7 :21

1.20.1 Available under license :

Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: @DEB_COPYRIGHT_UPSTREAMNAME@
Upstream-Contact: MySQL Release Engineering <mysql-build@oss.oracle.com>
Source: http://dev.mysql.com/

Copyright: 2015, 2016, Oracle and/or its affiliates. All rights reserved.
License:
This is a release of MySQL, a dual-license SQL database server.
For the avoidance of doubt, this particular copy of the software
is released under the version 2 of the GNU General Public License.
MySQL is brought to you by Oracle.
.
MySQL FOSS License Exception
We want free and open source software applications under certain
licenses to be able to use specified GPL-licensed MySQL client
libraries despite the fact that not all such FOSS licenses are
compatible with version 2 of the GNU General Public License.
Therefore there are special exceptions to the terms and conditions
of the GPLv2 as applied to these client libraries, which are
identified and described in more detail in the FOSS License
Exception at
<http://www.mysql.com/about/legal/licensing/foss-exception.html>.
.
This distribution may include materials developed by third
parties. For license and attribution notices for these
materials, please refer to the documentation that accompanies
this distribution (see the "Licenses for Third-Party Components"
appendix) or view the online documentation at
<http://dev.mysql.com/doc/>.
.
GPLv2 Disclaimer
For the avoidance of doubt, except that if any license choice

other than GPL or LGPL is available it will apply instead,
Oracle elects to use only the General Public License version 2
(GPLv2) at this time for any software where a choice of GPL
license versions is made available with the language indicating
that GPLv2 or any later version may be used, or where a choice
.
The full text of the GNU General Public License version 2 can
be found in the file
`/usr/share/mysql/doc/@DEB_INSTALL_LICENSEFILE@'.
Format: http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/
Upstream-Name: @DEB_COPYRIGHT_UPSTREAMNAME@
Upstream-Contact: MySQL Release Engineering <mysql-build@oss.oracle.com>
Source: http://dev.mysql.com/

Files: *
Copyright: 2000, 2016, Oracle and/or its affiliates. All rights reserved.
License: Commercial
This is a release of MySQL, a dual-license SQL database server.
For the avoidance of doubt, this particular copy of the software
is released under a commercial license and the GNU General Public
License does not apply. MySQL is brought to you by Oracle.
.
This distribution may include materials developed by third
parties. For license and attribution notices for these
materials, please refer to the documentation that accompanies
this distribution (see the "Licenses for Third-Party Components"
appendix) or view the online documentation at
<http://dev.mysql.com/doc/>.
.
This software and related documentation are provided under a license
agreement containing restrictions on use and disclosure and are protected by
intellectual property laws. Except as expressly permitted in your license
agreement or allowed by law, you may not use, copy, reproduce, translate,
broadcast, modify, license, transmit, distribute, exhibit, perform, publish,
or display any part, in any form, or by any means. Reverse engineering,
disassembly, or decompilation of this software, unless required by law for
interoperability, is prohibited.
.
If this software or related documentation is delivered to the U.S.
Government or anyone licensing it on behalf of the U.S. Government, the
following notice is applicable:
.
U.S. GOVERNMENT RIGHTS Programs, software, databases, and related
documentation and technical data delivered to U.S. Government customers are
"commercial computer software" or "commercial technical data" pursuant to
the applicable Federal Acquisition Regulation and agency-specific
supplemental regulations. As such, the use, duplication, disclosure,
modification, and adaptation shall be subject to the restrictions and

license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for

this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based

on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent

access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any

patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED

OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than `show w' and `show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

```
################## mysql-test\t\license_basic.test ############################
#                                                          #
# Variable Name: license                                   #
# Scope: Global                                            #
# Access Type: Static                                      #
# Data Type: string                                        #
#                                                          #
#                                                          #
# Creation Date: 2008-02-07                                #
# Author : Sharique Abdullah                               #
#                                                          #
#                                                          #
# Description:Test Cases of Dynamic System Variable license                #
#           that checks the behavior of this variable in the following ways #
#            * Value Check                                 #
#            * Scope Check                                 #
#                                                          #
# Reference: http://dev.mysql.com/doc/refman/5.1/en/                      #
```

```
#  server-system-variables.html                                    #
#                                                                  #
##############################################################################

--echo '#---------------------BS_STVARS_040_01----------------------#'
#####################################################################
#   Displaying default value                                        #
#####################################################################
SELECT COUNT(@@GLOBAL.license);
--echo 1 Expected


--echo '#---------------------BS_STVARS_040_02----------------------#'
#####################################################################
#   Check if Value can set                                          #
#####################################################################

--error ER_INCORRECT_GLOBAL_LOCAL_VAR
SET @@GLOBAL.license=1;
--echo Expected error 'Read only variable'

SELECT COUNT(@@GLOBAL.license);
--echo 1 Expected




--echo '#---------------------BS_STVARS_040_03----------------------#'
#################################################################
# Check if the value in GLOBAL Table matches value in variable  #
#################################################################

--disable_warnings
SELECT @@GLOBAL.license = VARIABLE_VALUE
FROM INFORMATION_SCHEMA.GLOBAL_VARIABLES
WHERE VARIABLE_NAME='license';
--enable_warnings
--echo 1 Expected

SELECT COUNT(@@GLOBAL.license);
--echo 1 Expected

--disable_warnings
SELECT COUNT(VARIABLE_VALUE)
FROM INFORMATION_SCHEMA.GLOBAL_VARIABLES
WHERE VARIABLE_NAME='license';
--enable_warnings
--echo 1 Expected
```

```
--echo '#---------------------BS_STVARS_040_04----------------------#'
################################################################################
#  Check if accessing variable with and without GLOBAL point to same variable  #
################################################################################
SELECT @@license = @@GLOBAL.license;
--echo 1 Expected



--echo '#---------------------BS_STVARS_040_05----------------------#'
################################################################################
#   Check if license can be accessed with and without @@ sign                 #
################################################################################

SELECT COUNT(@@license);
--echo 1 Expected

--Error ER_INCORRECT_GLOBAL_LOCAL_VAR
SELECT COUNT(@@local.license);
--echo Expected error 'Variable is a GLOBAL variable'

--Error ER_INCORRECT_GLOBAL_LOCAL_VAR
SELECT COUNT(@@SESSION.license);
--echo Expected error 'Variable is a GLOBAL variable'

SELECT COUNT(@@GLOBAL.license);
--echo 1 Expected

--Error ER_BAD_FIELD_ERROR
SELECT license = @@SESSION.license;
--echo Expected error 'Readonly variable'
'#---------------------BS_STVARS_040_01----------------------#'
SELECT COUNT(@@GLOBAL.license);
COUNT(@@GLOBAL.license)
1
1 Expected
'#---------------------BS_STVARS_040_02----------------------#'
SET @@GLOBAL.license=1;
ERROR HY000: Variable 'license' is a read only variable
Expected error 'Read only variable'
SELECT COUNT(@@GLOBAL.license);
COUNT(@@GLOBAL.license)
1
1 Expected
'#---------------------BS_STVARS_040_03----------------------#'
```

```
SELECT @@GLOBAL.license = VARIABLE_VALUE
FROM INFORMATION_SCHEMA.GLOBAL_VARIABLES
WHERE VARIABLE_NAME='license';
@@GLOBAL.license = VARIABLE_VALUE
1
1 Expected
SELECT COUNT(@@GLOBAL.license);
COUNT(@@GLOBAL.license)
1
1 Expected
SELECT COUNT(VARIABLE_VALUE)
FROM INFORMATION_SCHEMA.GLOBAL_VARIABLES
WHERE VARIABLE_NAME='license';
COUNT(VARIABLE_VALUE)
1
1 Expected
'#---------------------BS_STVARS_040_04----------------------#'
SELECT @@license = @@GLOBAL.license;
@@license = @@GLOBAL.license
1
1 Expected
'#---------------------BS_STVARS_040_05----------------------#'
SELECT COUNT(@@license);
COUNT(@@license)
1
1 Expected
SELECT COUNT(@@local.license);
ERROR HY000: Variable 'license' is a GLOBAL variable
Expected error 'Variable is a GLOBAL variable'
SELECT COUNT(@@SESSION.license);
ERROR HY000: Variable 'license' is a GLOBAL variable
Expected error 'Variable is a GLOBAL variable'
SELECT COUNT(@@GLOBAL.license);
COUNT(@@GLOBAL.license)
1
1 Expected
SELECT license = @@SESSION.license;
ERROR 42S22: Unknown column 'license' in 'field list'
Expected error 'Readonly variable'
```

Copyright 1992, 1993, 1994 Henry Spencer. All rights reserved.
This software is not subject to any license of the American Telephone
and Telegraph Company or of the Regents of the University of California.

Permission is granted to anyone to use this software for any purpose on
any computer system, and to alter it and redistribute it, subject
to the following restrictions:

1. The author is not responsible for the consequences of use of this

software, no matter how awful, even if they arise from flaws in it.

2. The origin of this software must not be misrepresented, either by
explicit claim or by omission. Since few users ever read sources,
credits must appear in the documentation.

3. Altered versions must be plainly marked as such, and must not be
misrepresented as being the original software. Since few users
ever read sources, credits must appear in the documentation.

4. This notice may not be removed or altered.

MEMCACHED CONTRIBUTORS

This file contains a list of people who have contributed code and
effort to the memcached project. If you don't see your name mentioned
send email to the memcached mailing list so you can be immortalized.

Also see the ChangeLog for even more people who have helped over the
years by submitting fixes, patches and reporting bugs.

Major authors:

Brad Fitzpatrick <brad@danga.com> -- maintainer, original implementations

Anatoly Vorobey <mellon@pobox.com> -- lots of the modern server code

Steven Grimm <sgrimm@facebook.com> -- iov writing (less CPU), UDP mode,
non-2.0 slab mantissas, multithread, ...

Other Contributors

Evan Martin <evan@danga.com>
Nathan Neulinger <nneul@umr.edu>
Eric Hodel <drbrain@segment7.net>
Michael Johnson <ahze@ahze.net>
Paul Querna <chip@corelands.com>
Jamie McCarthy <jamie@mccarthy.vg>
Philip Neustrom <philipn@gmail.com>
Andrew O'Brien <andrewo@oriel.com.au>
Josh Rotenberg <joshrotenberg@gmail.com>
Robin H. Johnson <robbat2@gentoo.org>
Tim Yardley <liquid@haveheart.com>
Paolo Borelli <paolo.borelli@gmail.com>
Eli Bingham <eli@pandora.com>
Jean-Francois Bustarret <jfbustarret@wat.tv>

```
Paul G <paul-lists@perforge.com>
Paul Lindner <lindner@inuus.com>
Dormando <dormando@rydia.net>
Dustin Sallings <dustin@spy.net>
Chris Goffinet <goffinet@yahoo-inc.com>
Tomash Brechko <tomash.brechko@gmail.com>
Brian Aker <brian@tangent.org>
Trond Norbye <trond.norbye@sun.com>
call mtr.add_suppression("Message of size .* received, exceeding the limit of");

--echo Setup
--source ../include/xplugin_preamble.inc

--echo Test
--let $_server_license= `SELECT IF(@@license = 'Commercial', 'PROPRIETARY', @@license);`
--let $_plugin_license= `select plugin_license from information_schema.plugins where plugin_name='mysqlx'`

if ($_server_license != $_plugin_license)
{
--echo Licenses doesn't match
--echo Servers license: $_server_license
--echo Plugins license: $_plugin_license
}

if ($_server_license == $_plugin_license)
{
--echo License ok
}

--echo Cleanup
UNINSTALL PLUGIN `mysqlx`;
call mtr.add_suppression("Message of size .* received, exceeding the limit of");
Setup
install plugin mysqlx soname "mysqlx.so";
call mtr.add_suppression("Plugin mysqlx reported: .Failed at SSL configuration: .SSL context is not usable without
certificate and private key..");
call mtr.add_suppression("Plugin mysqlx reported: .SSL_CTX_load_verify_locations failed.");
Test
License ok
Cleanup
UNINSTALL PLUGIN `mysqlx`;

 LICENSE ISSUES
 ==============

 The OpenSSL toolkit stays under a dual license, i.e. both the conditions of
 the OpenSSL License and the original SSLeay license apply to the toolkit.
 See below for the actual license texts. Actually both licenses are BSD-style
```

Open Source licenses. In case of any license issues related to OpenSSL
please contact openssl-core@openssl.org.

OpenSSL License

```
/* ====================================================================
* Copyright (c) 1998-1999 The OpenSSL Project.  All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in
*    the documentation and/or other materials provided with the
*    distribution.
*
* 3. All advertising materials mentioning features or use of this
*    software must display the following acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
*
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
*    endorse or promote products derived from this software without
*    prior written permission. For written permission, please contact
*    openssl-core@openssl.org.
*
* 5. Products derived from this software may not be called "OpenSSL"
*    nor may "OpenSSL" appear in their names without prior written
*    permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
*    acknowledgment:
*    "This product includes software developed by the OpenSSL Project
*    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
```

```
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* ====================================================================
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com).  This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/
```

Original SSLeay License

```
/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to.  The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code.  The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
*    must display the following acknowledgement:
*    "This product includes cryptographic software written by
*     Eric Young (eay@cryptsoft.com)"
```

```
*    The word 'cryptographic' can be left out if the rouines from the library
*    being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
*    the apps directory (application code) you must include an acknowledgement:
*    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed.  i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/
```

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it

in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of

running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the

entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not

compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the

integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS

TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this

when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year  name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.
This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:
Sanjay Ghemawat <sanjay@google.com>
Jeff Dean <jeff@google.com>
Daniel Dulitz <daniel@google.com>
Craig Silverstein
Paul Haahr <haahr@google.com>
Corey Anderson <corin@google.com>
(and many others)

Proto2 C++ and Java primary author:
Kenton Varda <kenton@google.com>

Proto2 Python primary authors:
Will Robinson <robinson@google.com>
Petar Petrov <petar@google.com>

Large code contributions:
Jason Hsueh <jasonh@google.com>

Joseph Schorr <jschorr@google.com>
Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:
Scott Bruce <sbruce@google.com>
Frank Yellin
Neal Norwitz <nnorwitz@google.com>
Jeffrey Yasskin <jyasskin@google.com>
Ambrose Feinstein <ambrose@google.com>

Documentation:
Lisa Carey <lcarey@google.com>

Maven packaging:
Gregory Kick <gak@google.com>

Patch contributors:
Kevin Ko <kevin.s.ko@gmail.com>
  * Small patch to handle trailing slashes in --proto_path flag.
Johan Euphrosine <proppy@aminche.com>
  * Small patch to fix Python CallMethod().
Ulrich Kunitz <kune@deine-taler.de>
  * Small optimizations to Python serialization.
Leandro Lucarella <llucax@gmail.com>
  * VI syntax highlighting tweaks.
  * Fix compiler to not make output executable.
Dilip Joseph <dilip.antony.joseph@gmail.com>
  * Heuristic detection of sub-messages when printing unknown fields in
    text format.
Brian Atkinson <nairb774@gmail.com>
  * Added @Override annotation to generated Java code where appropriate.
Vincent Choinire <Choiniere.Vincent@hydro.qc.ca>
  * Tru64 support.
Monty Taylor <monty.taylor@gmail.com>
  * Solaris 10 + Sun Studio fixes.
Alek Storm <alek.storm@gmail.com>
  * Slicing support for repeated scalar fields for the Python API.
Oleg Smolsky <oleg.smolsky@gmail.com>
  * MS Visual Studio error format option.
  * Detect unordered_map in stl_hash.m4.
Brian Olson <brianolson@google.com>
  * gzip/zlib I/O support.
Michael Poole <mdpoole@troilus.org>
  * Fixed warnings about generated constructors not explicitly initializing
    all fields (only present with certain compiler settings).
  * Added generation of field number constants.
Wink Saville <wink@google.com>
  * Fixed initialization ordering problem in logging code.

Will Pierce <willp@nuclei.com>
  * Small patch improving performance of in Python serialization.
Alexandre Vassalotti <alexandre@peadrop.com>
  * Emacs mode for Protocol Buffers (editors/protobuf-mode.el).
Scott Stafford <scott.stafford@gmail.com>
  * Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.
Alexander Melnikov <alm@sibmail.ru>
  * HPUX support.
Oliver Jowett <oliver.jowett@gmail.com>
  * Detect whether zlib is new enough in configure script.
  * Fixes for Solaris 10 32/64-bit confusion.
Evan Jones <evanj@mit.edu>
  * Optimize Java serialization code when writing a small message to a stream.
  * Optimize Java serialization of strings so that UTF-8 encoding happens only
    once per string per serialization call.
  * Clean up some Java warnings.
  * Fix bug with permanent callbacks that delete themselves when run.
Michael Kucharski <m.kucharski@gmail.com>
  * Added CodedInputStream.getTotalBytesRead().
Kacper Kowalik <xarthisius.kk@gmail.com>
  * Fixed m4/acx_pthread.m4 problem for some Linux distributions.
William Orr <will@worrbase.com>
  * Fixed detection of sched_yield on Solaris.
  * Added atomicops for Solaris

Copyright 2008, Google Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

  * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
  * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.
  * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner
of the input file used when generating it. This code is not
standalone and requires a support library to be linked with it. This
support library is itself covered by the above license.
Copyright (C) 2011 Milo Yip

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
LZ4 Library
Copyright (c) 2011-2014, Yann Collet
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this
 list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this
 list of conditions and the following disclaimer in the documentation and/or
 other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
License Disclaimer:

All contents of this directory are Copyright (c) the Dojo Foundation, with the
following exceptions:
---

dojo.css:
 * parts Copyright (c) 2007, Yahoo! Inc. All rights reserved.
   Distributed under the terms of the BSD License

The Program includes all or portions of the following software which was obtained under the terms and conditions of the BSD License.

http://developer.yahoo.com/yui/license.html

Copyright (c) 2007, Yahoo! Inc.
All rights reserved.
Redistribution and use of this software in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
* Neither the name of Yahoo! Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without
specific prior written permission of Yahoo! Inc.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Dojo is available under *either* the terms of the modified BSD license *or* the
Academic Free License version 2.1. As a recipient of Dojo, you may choose which

license to receive this code under (except as noted in per-module LICENSE files). Some modules may not be the copyright of the Dojo Foundation. These modules contain explicit declarations of copyright in both the LICENSE files in the directories in which they reside and in the code itself. No external contributions are allowed under licenses which are fundamentally incompatible with the AFL or BSD licenses that Dojo is distributed under.

The text of the AFL and BSD licenses is reproduced below.

The "New" BSD License:
****

Copyright (c) 2005-2011, The Dojo Foundation
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Dojo Foundation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Academic Free License, v. 2.1:
****

This Academic Free License (the "License") applies to any original work of authorship (the "Original Work") whose owner (the "Licensor") has placed the following notice immediately following the copyright notice for the Original Work:

Licensed under the Academic Free License version 2.1

1) Grant of Copyright License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license to do the following:

a) to reproduce the Original Work in copies;

b) to prepare derivative works ("Derivative Works") based upon the Original Work;

c) to distribute copies of the Original Work and Derivative Works to the public;

d) to perform the Original Work publicly; and

e) to display the Original Work publicly.

2) Grant of Patent License. Licensor hereby grants You a world-wide, royalty-free, non-exclusive, perpetual, sublicenseable license, under patent claims owned or controlled by the Licensor that are embodied in the Original Work as furnished by the Licensor, to make, use, sell and offer for sale the Original Work and Derivative Works.

3) Grant of Source Code License. The term "Source Code" means the preferred form of the Original Work for making modifications to it and all available documentation describing how to modify the Original Work. Licensor hereby agrees to provide a machine-readable copy of the Source Code of the Original Work along with each copy of the Original Work that Licensor distributes. Licensor reserves the right to satisfy this obligation by placing a machine-readable copy of the Source Code in an information repository reasonably calculated to permit inexpensive and convenient access by You for as long as Licensor continues to distribute the Original Work, and by publishing the address of that information repository in a notice immediately following the copyright notice that applies to the Original Work.

4) Exclusions From License Grant. Neither the names of Licensor, nor the names of any contributors to the Original Work, nor any of their trademarks or service marks, may be used to endorse or promote products derived from this Original Work without express prior written permission of the Licensor. Nothing in this License shall be deemed to grant any rights to trademarks, copyrights, patents, trade secrets or any other intellectual property of Licensor except as expressly stated herein. No patent license is granted to make, use, sell or offer to sell embodiments of any patent claims other than the licensed claims defined in Section 2. No right is granted to the trademarks of Licensor even if such marks are included in the Original Work. Nothing in this License shall be interpreted to prohibit Licensor from licensing under different terms from this License any Original Work that Licensor otherwise would have a right to

license.

5) This section intentionally omitted.

6) Attribution Rights. You must retain, in the Source Code of any Derivative Works that You create, all copyright, patent or trademark notices from the Source Code of the Original Work, as well as any notices of licensing and any descriptive text identified therein as an "Attribution Notice." You must cause the Source Code for any Derivative Works that You create to carry a prominent Attribution Notice reasonably calculated to inform recipients that You have modified the Original Work.

7) Warranty of Provenance and Disclaimer of Warranty. Licensor warrants that the copyright in and to the Original Work and the patent rights granted herein by Licensor are owned by the Licensor or are sublicensed to You under the terms of this License with the permission of the contributor(s) of those copyrights and patent rights. Except as expressly stated in the immediately proceeding sentence, the Original Work is provided under this License on an "AS IS" BASIS and WITHOUT WARRANTY, either express or implied, including, without limitation, the warranties of NON-INFRINGEMENT, MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY OF THE ORIGINAL WORK IS WITH YOU. This DISCLAIMER OF WARRANTY constitutes an essential part of this License. No license to Original Work is granted hereunder except under this disclaimer.

8) Limitation of Liability. Under no circumstances and under no legal theory, whether in tort (including negligence), contract, or otherwise, shall the Licensor be liable to any person for any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or the use of the Original Work including, without limitation, damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses. This limitation of liability shall not apply to liability for death or personal injury resulting from Licensor's negligence to the extent applicable law prohibits such limitation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so this exclusion and limitation may not apply to You.

9) Acceptance and Termination. If You distribute copies of the Original Work or a Derivative Work, You must make a reasonable effort under the circumstances to obtain the express assent of recipients to the terms of this License. Nothing else but this License (or another written agreement between Licensor and You) grants You permission to create Derivative Works based upon the Original Work or to exercise any of the rights granted in Section 1 herein, and any attempt to do so except under the terms of this License (or another written agreement between Licensor and You) is expressly prohibited by U.S. copyright law, the equivalent laws of other countries, and by international treaty. Therefore, by exercising any of the rights granted to You in Section 1 herein, You indicate Your acceptance of this License and all of its terms and conditions.

10) Termination for Patent Action. This License shall terminate automatically and You may no longer exercise any of the rights granted to You by this License as of the date You commence an action, including a cross-claim or counterclaim, against Licensor or any licensee alleging that the Original Work infringes a patent. This termination provision shall not apply for an action alleging patent infringement by combinations of the Original Work with other software or hardware.

11) Jurisdiction, Venue and Governing Law. Any action or suit relating to this License may be brought only in the courts of a jurisdiction wherein the Licensor resides or in which Licensor conducts its primary business, and under the laws of that jurisdiction excluding its conflict-of-law provisions. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any use of the Original Work outside the scope of this License or after its termination shall be subject to the requirements and penalties of the U.S. Copyright Act, 17 U.S.C. 101 et seq., the equivalent laws of other countries, and international treaty. This section shall survive the termination of this License.

12) Attorneys Fees. In any action to enforce the terms of this License or seeking damages relating thereto, the prevailing party shall be entitled to recover its costs and expenses, including, without limitation, reasonable attorneys' fees and costs incurred in connection with such action, including any appeal of such action. This section shall survive the termination of this License.

13) Miscellaneous. This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable.

14) Definition of "You" in This License. "You" throughout this License, whether in upper or lower case, means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with you. For purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

15) Right to Use. You may use the Original Work in all ways not otherwise restricted or conditioned by this License or by law, and Licensor promises not to interfere with or be responsible for such uses by You.

This license is Copyright (C) 2003-2004 Lawrence E. Rosen. All rights reserved. Permission is hereby granted to copy and distribute this license without modification. This license may not be modified without the express written

permission of its copyright owner.
UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/,
and http://www.unicode.org/cldr/data/ . Unicode Software includes any source code published in the Unicode Standard or under
the directories http://www.unicode.org/Public/, http://www.unicode.org/reports/, and http://www.unicode.org/cldr/data/.
NOTICE TO USER: Carefully read the following legal agreement. BY DOWNLOADING, INSTALLING, COPYING OR
OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU
UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS
AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE
Copyright 1991-2007 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in http://www.unicode.org/copyright.html.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated
documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data
Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell
copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided
that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the
above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File
or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.
THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,
INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS
INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES,
OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF
CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR

PERFORMANCE OF THE DATA FILES OR SOFTWARE.
Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other
dealings in these Data Files or Software without prior written authorization of the copyright holder.
License Disclaimer:

All contents of this directory are Copyright (c) the Dojo Foundation, with the
following exceptions:

_crypto.js - internally uses AES algorithm
 * AES algorithm copyright Chris Veness (CLA signed and permission given to use code under BSD license)
    Taken from http://www.movable-type.co.uk/scripts/aes.html
Copyright (c) 2003, Danga Interactive, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

  * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

  * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

  * Neither the name of the Danga Interactive nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Portions of this software contain modifications contributed by Percona, Inc.
These contributions are used with the following license:

Copyright (c) 2008, 2009, Percona Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Percona Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Portions of this software contain modifications contributed by Google, Inc.
These contributions are used with the following license:

Copyright (c) 2008, Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
* Neither the name of the Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT

OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.21 mysql-connector-java 5.1.45

1.21.1 Available under license :

The GNU General Public License, Version 2
http://www.gnu.org/licenses/old-licenses/gpl-2.0.html
MySQL Connector/J contains exceptions to GPL requirements when linking with other components that are licensed under OSI-approved open source licenses, see EXCEPTIONS-CONNECTOR-J in this distribution for more details.

```
/*
 Copyright (c) 2002, 2014, Oracle and/or its affiliates. All rights reserved.

 The MySQL Connector/J is licensed under the terms of the GPLv2
 <http://www.gnu.org/licenses/old-licenses/gpl-2.0.html>, like most MySQL Connectors.
 There are special exceptions to the terms and conditions of the GPLv2 as it is applied to
 this software, see the FOSS License Exception
 <http://www.mysql.com/about/legal/licensing/foss-exception.html>.

 This program is free software; you can redistribute it and/or modify it under the terms
 of the GNU General Public License as published by the Free Software Foundation; version 2
 of the License.

 This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY;
 without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
 See the GNU General Public License for more details.

 You should have received a copy of the GNU General Public License along with this
 program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth
 Floor, Boston, MA 02110-1301  USA
```

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a

notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this

License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free

programs whose distribution conditions are different, write to the author
to ask for permission. For software which is copyrighted by the Free
Software Foundation, write to the Free Software Foundation; we sometimes
make exceptions for this. Our decision will be guided by the two goals
of preserving the free status of all derivatives of our free software and
of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY
FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN
OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES
PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED
OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS
TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE
PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING,
REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING
WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR
REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES,
INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING
OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED
TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY
YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER
PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE
POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest
possible use to the public, the best way to achieve this is to make it
free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest
to attach them to the start of each source file to most effectively
convey the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License along
with this program; if not, write to the Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type 'show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate
parts of the General Public License. Of course, the commands you use may
be called something other than 'show w' and 'show c'; they could even be
mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

1.22 nginx 1.14.2

1.22.1 Available under license :

```
/*
* Copyright (C) 2002-2018 Igor Sysoev
* Copyright (C) 2011-2018 Nginx, Inc.
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*    notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*    notice, this list of conditions and the following disclaimer in the
*    documentation and/or other materials provided with the distribution.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR AND CONTRIBUTORS ``AS IS'' AND
```

* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*/

1.23 pygraphviz 1.5

1.23.1 Available under license :

Copyright (C) 2004-2018 by
Aric Hagberg <hagberg@lanl.gov>
Dan Schult <dschult@colgate.edu>
Manos Renieris, http://www.cs.brown.edu/~er/
Distributed with BSD license.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
  notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
  copyright notice, this list of conditions and the following
  disclaimer in the documentation and/or other materials provided
  with the distribution.

* Neither the name of the <ORGANIZATION> nor the names of its
  contributors may be used to endorse or promote products derived
  from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Legal
=====

PyGraphviz License

Copyright (C) 2004-2010 by
Aric Hagberg <hagberg@lanl.gov>
Dan Schult <dschult@colgate.edu>
Manos Renieris, http://www.cs.brown.edu/~er/
Distributed with BSD license.
All rights reserved, see LICENSE for details.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of the <ORGANIZATION> nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Notice

This software and ancillary information (herein called SOFTWARE)
called pygraphviz is made available under the terms described
here. The SOFTWARE has been approved for release with associated LA-CC
number 04-073.

Unless otherwise indicated, this SOFTWARE has been authored by an
employee or employees of the University of California, operator of the
Los Alamos National Laboratory under Contract No. W-7405-ENG-36 with
the U.S. Department of Energy. The U.S. Government has rights to use,
reproduce, and distribute this SOFTWARE. The public may copy,
distribute, prepare derivative works and publicly display this
SOFTWARE without charge, provided that this Notice and any statement
of authorship are reproduced on all copies. Neither the Government nor
the University makes any warranty, express or implied, or assumes any
liability or responsibility for the use of this SOFTWARE.

If SOFTWARE is modified to produce derivative works, such modified
SOFTWARE should be clearly marked, so as not to confuse it with the
version available from Los Alamos National Laboratory.

1.24 pymongo 3.7.2

1.24.1 Available under license :

PyMongo uses third-party libraries or other resources that may
be distributed under licenses different than the PyMongo software.

In the event that we accidentally failed to list a required notice,
please bring it to our attention through any of the ways detailed here:

mongodb-dev@googlegroups.com

The attached notices are provided for information only.

For any licenses that require disclosure of source, sources are available at
https://github.com/mongodb/mongo-python-driver.

1) License Notice for time64.c

Copyright (c) 2007-2010 Michael G Schwern

This software originally derived from Paul Sheer's pivotal_gmtime_r.c.

The MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy

of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in
all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

2) License Notice for bson-stdint-win32.h

ISO C9x compliant stdint.h for Microsoft Visual Studio
Based on ISO/IEC 9899:TC2 Committee draft (May 6, 2005) WG14/N1124

Copyright (c) 2006-2013 Alexander Chemeris

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice,
   this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.

3. Neither the name of the product nor the names of its contributors may
   be used to endorse or promote products derived from this software
   without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF

ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

3) License Notice for encoding_helpers.c

Portions Copyright 2001 Unicode, Inc.

Disclaimer

This source code is provided as is by Unicode, Inc. No claims are made as to fitness for any particular purpose. No warranties of any kind are expressed or implied. The recipient agrees to determine applicability of information provided. If this file has been purchased on magnetic or optical media from Unicode, Inc., the sole remedy for any claim will be exchange of defective media within 90 days of receipt.

Limitations on Rights to Redistribute This Code

Unicode, Inc. hereby grants the right to freely use the information supplied in this file in the creation of products supporting the Unicode Standard, and to make copies of this file in any form for internal or external distribution as long as this notice remains attached.

4) License Notice for ssl_match_hostname.py

Python License (Python-2.0)

Python License, Version 2 (Python-2.0)

PYTHON SOFTWARE FOUNDATION LICENSE VERSION 2
---

1. This LICENSE AGREEMENT is between the Python Software Foundation ("PSF"), and the Individual or Organization ("Licensee") accessing and otherwise using this software ("Python") in source or binary form and its associated documentation.

2. Subject to the terms and conditions of this License Agreement, PSF hereby grants Licensee a nonexclusive, royalty-free, world-wide license to reproduce, analyze, test, perform and/or display publicly, prepare derivative works, distribute, and otherwise use Python alone or in any derivative version, provided, however, that PSF's License Agreement and PSF's notice of copyright, i.e., "Copyright (c) 2001-2013 Python Software Foundation; All Rights Reserved" are retained in Python alone or in any derivative version prepared by Licensee.

3. In the event Licensee prepares a derivative work that is based on
or incorporates Python or any part thereof, and wants to make
the derivative work available to others as provided herein, then
Licensee hereby agrees to include in any such work a brief summary of
the changes made to Python.

4. PSF is making Python available to Licensee on an "AS IS"
basis. PSF MAKES NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR
IMPLIED. BY WAY OF EXAMPLE, BUT NOT LIMITATION, PSF MAKES NO AND
DISCLAIMS ANY REPRESENTATION OR WARRANTY OF MERCHANTABILITY OR FITNESS
FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF PYTHON WILL NOT
INFRINGE ANY THIRD PARTY RIGHTS.

5. PSF SHALL NOT BE LIABLE TO LICENSEE OR ANY OTHER USERS OF PYTHON
FOR ANY INCIDENTAL, SPECIAL, OR CONSEQUENTIAL DAMAGES OR LOSS AS
A RESULT OF MODIFYING, DISTRIBUTING, OR OTHERWISE USING PYTHON,
OR ANY DERIVATIVE THEREOF, EVEN IF ADVISED OF THE POSSIBILITY THEREOF.

6. This License Agreement will automatically terminate upon a material
breach of its terms and conditions.

7. Nothing in this License Agreement shall be deemed to create any
relationship of agency, partnership, or joint venture between PSF and
Licensee. This License Agreement does not grant permission to use PSF
trademarks or trade name in a trademark sense to endorse or promote
products or services of Licensee, or any third party.

8. By copying, installing or otherwise using Python, Licensee
agrees to be bound by the terms and conditions of this License
Agreement.

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and
wherever such third-party notices normally appear. The contents
of the NOTICE file are for informational purposes only and
do not modify the License. You may add Your own attribution
notices within Derivative Works that You distribute, alongside
or as an addendum to the NOTICE text from the Work, provided
that such additional attribution notices cannot be construed
as modifying the License.

You may add Your own copyright statement to Your modifications and
may provide additional or different license terms and conditions
for use, reproduction, or distribution of Your modifications, or
for any such Derivative Works as a whole, provided Your use,
reproduction, and distribution of the Work otherwise complies with
the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,
any Contribution intentionally submitted for inclusion in the Work
by You to the Licensor shall be under the terms and conditions of
this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify
the terms of any separate license agreement you may have executed
with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade
names, trademarks, service marks, or product names of the Licensor,
except as required for reasonable and customary use in describing the
origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or
agreed to in writing, Licensor provides the Work (and each
Contributor provides its Contributions) on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
implied, including, without limitation, any warranties or conditions
of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
PARTICULAR PURPOSE. You are solely responsible for determining the
appropriateness of using or redistributing the Work and assume any
risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,
whether in tort (including negligence), contract, or otherwise,
unless required by applicable law (such as deliberate and grossly
negligent acts) or agreed to in writing, shall any Contributor be
liable to You for damages, including any direct, indirect, special,
incidental, or consequential damages of any character arising as a
result of this License or out of the use or inability to use the
Work (including but not limited to damages for loss of goodwill,
work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor
has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Contributors
==========
The following is a list of people who have contributed to
PyMongo. If you belong here and are missing please let us know
(or send a pull request after adding yourself to the list):

- Mike Dirolf (mdirolf)
- Jeff Jenkins (jeffjenkins)

- Jim Jones
- Eliot Horowitz (erh)
- Michael Stephens (mikejs)
- Joakim Sernbrant (serbaut)
- Alexander Artemenko (svetlyak40wt)
- Mathias Stearn (RedBeard0531)
- Fajran Iman Rusadi (fajran)
- Brad Clements (bkc)
- Andrey Fedorov (andreyf)
- Joshua Roesslein (joshthecoder)
- Gregg Lind (gregglind)
- Michael Schurter (schmichael)
- Daniel Lundin
- Michael Richardson (mtrichardson)
- Dan McKinley (mcfunley)
- David Wolever (wolever)
- Carlos Valiente (carletes)
- Jehiah Czebotar (jehiah)
- Drew Perttula (drewp)
- Carl Baatz (c-w-b)
- Johan Bergstrom (jbergstroem)
- Jonas Haag (jonashaag)
- Kristina Chodorow (kchodorow)
- Andrew Sibley (sibsibsib)
- Flavio Percoco Premoli (FlaPer87)
- Ken Kurzweil (kurzweil)
- Christian Wyglendowski (dowski)
- James Murty (jmurty)
- Brendan W. McAdams (bwmcadams)
- Bernie Hackett (behackett)
- Reed O'Brien (reedobrien)
- Francisco Souza (fsouza)
- Alexey I. Froloff (raorn)
- Steve Lacy (slacy)
- Richard Shea (shearic)
- Vladimir Sidorenko (gearheart)
- Aaron Westendorf (awestendorf)
- Dan Crosta (dcrosta)
- Ryan Smith-Roberts (rmsr)
- David Pisoni (gefilte)
- Abhay Vardhan (abhayv)
- Alexey Borzenkov (snaury)
- Kostya Rybnikov (k-bx)
- A Jesse Jiryu Davis (ajdavis)
- Samuel Clay (samuelclay)
- Ross Lawley (rozza)
- Wouter Bolsterlee (wbolster)
- Alex Grnholm (agronholm)

- Christoph Simon (kalanzun)
- Chris Tompkinson (tompko)
- Mike O'Brien (mpobrien)
- T Dampier (dampier)
- Michael Henson (hensom)
- Craig Hobbs (craigahobbs)
- Emily Stolfo (estolfo)
- Sam Helman (shelman)
- Justin Patrin (reversefold)
- Xiuming Chen (cxmcc)
- Tyler Jones (thomascirca)
- Amalia Hawkins (hawka)
- Yuchen Ying (yegle)
- Kyle Erf (3rf)
- Luke Lovett (lovett89)
- Jaroslav Semank (girogiro)
- Don Mitchell (dmitchell)
- Ximing (armnotstrong)
- Can Zhang (cannium)
- Sergey Azovskov (last-g)
- Heewa Barfchin (heewa)
- Anna Herlihy (aherlihy)
- Len Buckens (buckensl)
- ultrabug
- Shane Harvey (ShaneHarvey)
- Cao Siyang (caosiyang)
- Zhecong Kwok (gzcf)
- TaoBeier(tao12345666333)
- Jagrut Trivedi(Jagrut)

1.25 rabbitmq-server 3.7.9

1.25.1 Available under license :

This package, the RabbitMQ server is licensed under the MPL 1.1. For the MPL 1.1, please see LICENSE-MPL-RabbitMQ.

If you have any questions regarding licensing, please contact us at info@rabbitmq.com.

MOZILLA PUBLIC LICENSE
Version 1.1

---

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or

hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.
The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused

by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.
Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.
The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this

License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.
Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.
You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.
If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.
If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by

the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.
You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.
Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.
Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.
If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your

license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declatory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under

Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995),

all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.mozilla.org/MPL/

Software distributed under the License is distributed on an "AS IS"
basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the
License for the specific language governing rights and limitations
under the License.

The Original Code is RabbitMQ.

The Initial Developer of the Original Code is Pivotal Software, Inc.
Copyright (c) 2007-2016 Pivotal Software, Inc. All rights reserved."

[NOTE: The text of this Exhibit A may differ slightly from the text of
the notices in the Source Code files of the Original Code. You should
use the text of this Exhibit A rather than the text found in the
Original Code Source Code for Your Modifications.]

1.26 redhat-lsb-core 4.1 :27.el7

1.26.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change free
software--to make sure the software is free for all its users. This
General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit to
using it. (Some other Free Software Foundation software is covered by
the GNU Library General Public License instead.) You can apply it to
your programs, too.

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that you
have the freedom to distribute copies of free software (and charge for
this service if you wish), that you receive source code or can get it
if you want it, that you can change the software or use pieces of it
in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the rights.

These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to

exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt

otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that

system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.

This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

1.27 slowhttptest 1.7

1.27.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing
the Work or Derivative Works thereof, You may choose to offer,
and charge a fee for, acceptance of support, warranty, indemnity,
or other liability obligations and/or rights consistent with this
License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright 2012 Sergey Shekyan Victor Agababov

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.28 tetpyclient 1.0.7

1.28.1 Available under license :

CISCO API LICENSE

These terms govern your use of this Cisco Application Programming Interface and
associated documentation (Cisco API). By accessing, downloading, copying,
distributing, calling, or otherwise making any use of the Cisco API, you

accept and agree to be bound by the following terms and conditions (the License). If you are accepting the License on behalf of an entity, you represent that you have the authority to do so (either you or the entity, you). All use of the Cisco API must be in accordance with this License and all rights not expressly granted herein are reserved.

1. LICENSE GRANT: Subject to the terms and conditions of this License, Cisco hereby grants to you a perpetual, worldwide, non-exclusive, non-transferable, non-sublicensable, royalty-free license to use and make calls to the Cisco API for the sole purpose of developing, implementing, and distributing software applications that work, communicate, or interact with Cisco products or services.

2. CONDITIONS: Your use of the Cisco API is subject to full compliance with Ciscos Cloud Services Acceptable Use Policy (http://www.cisco.com/c/en/us/about/legal/end-user-license-and-cloud-terms/cloud-services-acceptable-use-policy.html).
You shall not use the Cisco API to substantially replicate and compete with the underlying Cisco product or service. You will not collect, store, or transmit any confidential, sensitive, or protected data or information that you receive as a result of your use of the Cisco API. The underlying Cisco product or service is licensed under its own separate terms and you shall not use the Cisco API in any way that violates or is inconsistent with those terms (for more information, please visit: www.cisco.com/go/terms). Failure to comply with any part of this section constitutes a material breach of this License and may result in further repercussions and actions against you, at Ciscos sole discretion.

3. OWNERSHIP: The Cisco API, in whole or in part, in all forms, is the sole and exclusive property of Cisco. This License does not grant you permission to use any trade names, trademarks, service marks, or product names of Cisco. If you provide any feedback to Cisco regarding the Cisco API, you agree that Cisco, its suppliers, and its customers shall be free to use and incorporate such feedback into the Cisco API and Cisco products and services without restriction, payment, or additional consideration of any kind. If you initiate or participate in any litigation against Cisco, its partners, or its customers (including cross-claims and counter-claims) alleging that the Cisco API or its use infringe any patent, copyright, or other intellectual property right, then all rights granted to you under this License shall terminate immediately.

4. LIMITATION OF LIABILITY: CISCO SHALL HAVE NO LIABILITY IN CONNECTION WITH OR RELATING TO THIS LICENSE OR USE OF THE CISCO API, FOR DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO DIRECT, INCIDENTAL, AND CONSEQUENTIAL DAMAGES, OR FOR ANY LOSS OF USE, DATA, INFORMATION, PROFITS, BUSINESS, OR GOODWILL, HOWEVER CAUSED, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

5. DISCLAIMER OF WARRANTY: CISCO PROVIDES THE CISCO API AS IS WITH ALL FAULTS, WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED BY LAW, ALL

EXPRESS AND IMPLIED CONDITIONS, REPRESENTATIONS, AND WARRANTIES INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTY OR CONDITION OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, SATISFACTORY QUALITY, NON-INTERFERENCE, AND ACCURACY, ARE HEREBY EXCLUDED AND EXPRESSLY DISCLAIMED BY
CISCO. CISCO DOES NOT WARRANT THAT THE CISCO API IS SUITABLE FOR YOUR USE, WILL OPERATE PROPERLY WITH YOUR APPLICATION, IS ACCURATE OR COMPLETE, OR IS WITHOUT ERROR OR DEFECT.

6. GENERAL: This License shall be governed by and interpreted in accordance with the laws of the State of California, excluding its conflict of laws provisions. You agree to comply with all applicable United States export laws, rules, and regulations. If any provision of this License is judged illegal, invalid, or otherwise unenforceable, that provision shall be severed and the rest of the License shall remain in full force and effect. No failure by Cisco to enforce any of its rights related to the Cisco API or to a breach of this License in a particular situation will act as a waiver of such rights. In the event of any inconsistencies with any other terms that may apply to the Cisco API, this License shall take precedence.

1.29 tomcat-native 1.1.34 :1.el7

1.29.1 Available under license :

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only
on Your own behalf and on Your sole responsibility, not on behalf
of any other Contributor, and only if You agree to indemnify,
defend, and hold each Contributor harmless for any liability
incurred by, or claims asserted against, such Contributor by reason
of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Tomcat Native Library
Copyright 2002-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (http://www.apache.org/).

1.30 wget 1.19.5

1.30.1 Available under license :

GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <https://fsf.org/>
Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic

pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you

receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded

from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or

b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you

received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or

run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement,

or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you

to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest

to attach them to the start of each source file to most effectively
state the exclusion of warranty; and each file should have at least
the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <https://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<https://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<https://www.gnu.org/philosophy/why-not-lgpl.html>.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries.To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a

partnership relationship between Cisco and any other company. (1110R)

©2019 Cisco Systems, Inc. All rights reserved.