

Open Source Licenses

Chris Hinds-Tennie, iNOC inc.

Terje Krogdahl, BaneTele AS, and

Kari Asheim, BaneTele AS

Abstract

Although Open Source software is designed to be very free in terms of its use and distribution, it is still common to have a license governing the software. The term 'license' is commonly used to describe something one receives from a software vendor which gives the right to use a particular product. Like other licenses Open Source licenses aim to protect the rights of authors but differ in the fact that they are also aimed at protecting the rights of the users of the software by explicitly granting privileges regarding modification, re-distribution, and access to source-code.

Use, modification, distribution and sale

Software licenses are typically used to protect the authors of the software and restrict exactly what an end-user is and is not allowed to do. The clauses meant to protect the author, often stating that the product comes without a warranty and other such disclaimers, are common in both 'closed-source' as well as Open Source software. However, these licenses tend to differ regarding acceptable use: a typical 'closed-source' software license may explicitly outline how the software may or may not be used such as forbidding reverse engineering or dictating under what circumstances you may make backup copies of the software. In contrast Open Source software licenses are focused on granting end-users rights and typically only put restrictions in place where it is believed they are necessary to ensure greater freedom for the user community as a whole.

Another key aspect of Open Source packages is the right to modify them. Open Source licenses explicitly grant this right and may regulate the terms that allow modification; for instance, by possibly requiring that the modified source code be published.

Redistribution is another defining component of Open Source software. The right to redistribute the software with or without modification is something allowed in the licensing of all Open Source packages, typically through explicitly granting that right in the license.

Commercial distribution and sale of Open Source software can be a tricky area. Open Source licenses normally address this issue directly; however, there can be considerable variation in how this is handled between different Open Source licenses. If someone publishes a piece of code, the author may wish to ensure that others do not take unfair commercial advantage of his or her efforts by making sure that the source code is also made available to the end users. As Open Source software is becoming popular in embedded devices like DSL modems, routers, multimedia devices and home appliances this area is becoming increasingly important.

An often misunderstood area regarding licensing is retention of copyright and patents. Contributing or creating Open Source software does not remove or transfer those rights as part of the license itself but it does grant end-users the right to use, modify, and redistribute the software under the terms of the chosen license. The authors still retains full ownership of their copyrights and patents and may use them as they desire.

Copyleft

Copyleft is a concept advanced by the Free Software Foundation, one of the key organizations in the Open Source movement. Copyleft subscribes to the mindset that copyright law can be subverted to ensure that the freedoms of end-users remain in place instead of being used, as it frequently is, as a means to restrict the rights of end-users. All Open Source licenses, as noted above, follow this to some extent; however, the ability to integrate with or redistribute Open Source software as 'closed-source' software is a point of contention. In the copyleft ideal every Open Source package will remain a completely Open Source package explicitly maintaining the same freedoms even after modification. Licenses that enforce this generational passing of total freedoms are sometimes referred to as 'strong-copyleft' licenses. Some licenses explicitly allow linking to works that are 'closed-source' to form a single package with the modified code continuing to be made Open Source, this is often referred to as 'weak-copyleft' licenses. Finally, licenses that allow Open Source software to be made 'closed-source' or take away freedoms of others in another fashion are considered 'non-copyleft' licenses.

Popular licenses

There are at least 50 well-known licenses out there, and this paper can only touch on a few of the more popular ones. See the references section for pointers to some more comprehensive lists of licenses. To get a better grasp on the differences between each license each has been broken down into families of either strong, weak, or non-copyleft licenses.

Strong Copyleft

GPL

Perhaps the most famous Open Source license is the GNU General Public License, or GPL, published by the Free Software Foundation (FSF). It has been around since 1989 and is used for many popular Open Source software products, the Linux kernel being a very well known example. The license is rooted in FSF founder Richard Stallman's wish to provide freedom to the programming community. The 'free' in free software is meant to stress the free in freedom, not necessarily free in terms of price. The license states that you are free to copy, redistribute and modify the program as long as you maintain that the rights of the recipients are the same as your own so any recipient may copy, redistribute or modify the program. This makes the GPL a strong-copyleft license. Maintaining this same level of rights means making the source code available is a requirement of the license. The GPL does, however, explicitly state that you may charge a fee for distributing the software or for offering a warranty. If you make any changes the license says you have to explicitly note the changes in order to protect the original author. The second part of the license states that the program is provided 'as is', and someone who provides a program under GPL is not liable for any loss etc. related to the program.

Weak Copyleft

LGPL

The LGPL or GNU Lesser Public License is also published by the Free Software Foundation. The LGPL is designed to allow Open Source software to be linked to 'closed-source' software in situations where this is necessary typically, but not restricted to, linking to 'closed-source' libraries. Additional language within the license describes exactly what situations linking with these other incompatibly licensed code is permitted. The fact that LGPL licensed software can be linked to other 'closed-source' or otherwise incompatibly licensed software to create

a complete package is what leads to the LGPL being considered a weak-copyleft license. Outside of this condition, the LGPL is very similar to the GPL in its terms allowing the use, modification, redistribution, and licensing of the software.

MPL

The Mozilla Public License from the Mozilla Project is a weak-copyleft license similar to the LGPL but with several additional conditions. The license specifically details steps regarding patent infringement concerns, notably if patent infringement claims are filed all patent rights provided to the party pursuing the litigation are forfeited after 60 days. This section regarding patent right revocation is similar to those outlined in the Apache license. The license also outlines the jurisdiction for litigation within the United State, this is common of many custom Open Source licenses from software vendors. Interestingly the MPL both allows a person to provide warranty or indemnification within the license as long as he or she clearly denote that he are she is assuming that responsibility. Finally, the MPL also allows for dual licensing of the software or sections of the software such as for instance, concurrently offering the MPL and GPL as license options.

Non-Copyleft

BSD

The BSD license, along with the Free Software Foundation's GPL, is one of the most influential and widely used Open Source licenses. The BSD license is quite different from the GPL in that there are no requirements to share the source code. In fact, it allows any use and redistribution of source code or binaries, original or modified, and does not require that modifications be shared with the public. This makes the BSD license a non-copyleft license. The only requirements are that the copyright and license terms accompany the product and that the University of California Berkley or other contributors are not used to promote software derived from this product without permission. A warranty disclaimer is also included, stating that the software is provided 'as-is' and that contributors are not liable for any consequences of using the software. The original version of the BSD license used to contain an additional clause that required crediting the University of California Berkley in any advertising or use of the software, this clause was removed in 1999. The relative simplicity of the license and the ability to use this software in 'closed-source' commercial offerings has made BSD licensed software popular for embedding in commercial products, an example is the use of a modified FreeBSD kernel in the JUNOS software governing the operation of many Juniper Networks routers.

MIT / Expat

The MIT / Expat license is for all intents and purposes the same as the modern BSD license. The only notable difference is that the MIT / Expat license lacks the clause prohibiting promotion or endorsement by the developers of the software that the BSD license contains. As with the BSD license the MIT / Expat license is a non-copyleft license.

Apache

From the Apache Software Foundation comes the Apache license. The current version of the Apache license, version 2.0, is an expansion on the concepts of the BSD license while still overall maintaining its non-copyleft aspects, such as allowing closed-source changes. First it clarifies the terms of all submitted contributions to the software, explicitly stating that they fall under the same terms as the original Apache license unless they are noted

otherwise. The license also explicitly discusses patent concerns, noting that contributors are explicitly granting rights by their submission. It goes on to note that those who pursue patent litigation in regard to the Apache licensed product have any of the fore-mentioned patent rights revoked. Finally, the Apache license also restricts the use of trademark and service marks owned by the Apache Software Foundation. Many of these expanded items are similar to conditions expressed in the Mozilla Public License.

Conclusion

The number of licenses is a good indication that there are numerous different opinions on exactly what conditions an Open Source license should have. The sheer quantity of these licenses can, at times, make picking and differentiating between them difficult. Specifically, there will be ongoing discussions of which requirements should be fulfilled in order for a modification to be published under a different license, and when it must be included in the original license. The finer details regarding patent rights will also be an issue for years to come. In conclusion, Open Source licenses will continue to mature in the face of greater public awareness and we will likely see greater numbers of licenses explore different options before any possible consolidation takes place.

References

Licenses:

GPL: <http://www.gnu.org/copyleft/gpl.html>
LGPL: <http://www.gnu.org/copyleft/lgpl.html>
MPL: <http://www.mozilla.org/MPL/MPL-1.1.html>
BSD: <http://opensource.org/licenses/bsd-license.php>
MIT / Expat: <http://opensource.org/licenses/mit-license.php>
Apache: <http://www.apache.org/licenses/LICENSE-2.0>

More comprehensive lists may be found at:

Open Source Initiative <http://opensource.org/licenses/>
GNU Project <http://www.gnu.org/licenses/license-list.html>