



qLegal at the Legal Advice Centre
lac@qmul.ac.uk
qlegal.qmul.ac.uk

Disclaimer: This toolkit describes the law in general terms. It is not intended to provide legal advice on specific situations and should not be relied upon as a source of legal advice.

Date produced: 20/01/2015

qLegal Toolkits

OPEN SOURCE AND LICENSING

Introduction to Open Source

Protection of computer programmes, including the so-called Open Source Software (OSS), is largely regulated by copyright law. Copyright protection of computer programs is regulated in the UK under the Copyright, Designs and Patents Act 1988 and European Union Software Directive 2009/24/EC. Under CDPA 1988 computer programs are included into the copyright category of literary works.

Copyright begins automatically with the creation of a work and without the need for compliance with any formal procedures if the work is original, i.e., according to European Union Software Directive, it means it's "author's own intellectual creation".

The first owner of copyright in a work is usually the author of the work. However, there are exceptions to this common rule. The most important exception is that, subject to a contrary agreement, the first owner of copyright in a work created by an employee during the course of his or her employment is the employer, not the employee. The length of the term of protection of computer programs is 70 years from the end of the year in which the author dies.

The owner of copyright has an exclusive right to control the copying of the computer program, the adaptation of it, the issuing of copies to the public (distribution) and its communication to the public (including making available). The violation of those rights is a copyright infringement.

At the same time lawful users of the program have a right to observe, study and test the software; make back-up copies of it; decompile a program in order to achieve interoperability; copy and adapt programs if it is necessary for lawful use and is not prohibited by contract.

Copyright can be given away, bought and sold like a personal or movable property. Copyright owners can make money from their work by assignment of copyright (handing

over ownership of the work) or by licensing it (permitting the licensee to take certain actions with software that would otherwise be prohibited by law).

Licenses other than exclusive licenses can be made informally without evidence in writing. Software licenses are usually recorded in a written statement of terms without signing agreement or contract. Assignments of copyright or future copyright will only be effective if made in writing and signed by or on behalf of the assignor.

The fundamental difference between the proprietary licensing model and open source licensing lies in the way the copyright owner grants their permission to others to use the software. Where in the proprietary licensing model such permission is usually granted in consideration of a payment, either a one-off payment or periodic payments, and the licence also typically contains strict conditions on the use of the software and its further development by the users; an open source licence does not commonly require any payments from users. Instead, it obliges users to keep the software and derivative works as open source and not to foreclose further development of that software by other users through proprietary licensing (see below).

Open Source Initiative (<http://opensource.org/>), a non-profit corporation formed in 1998, provides the Open Source Definition that sets ten criteria in order for licence to qualify as open source. (See below)

The desired effect of the conditions is to promote a wide distribution of software, and to encourage people to contribute to software functionality by modifying the source code. According to the OSI Mission Statement, open source is a development method for software that harnesses the power of distributed peer review and transparency of process. The advantages of open source are better quality, higher reliability, more flexibility, lower cost and no vendor lock-in.

Essentially, such software needs to be freely available to the public without the possibility of any patent (where applicable) or other types of restricted protection on the software further development that may cause users to be at risk of infringing proprietary rights of the author.

Open Source Definition

According to the Open Source Initiative, there are ten criteria which must be satisfied in order for any software in question to be considered “open source”.

Free redistribution

The license given cannot be restrictive in terms of the methods of distribution. It also cannot request a fee or royalties to be given for the use, sale, or distribution of the software.

Source Code

The source code of the program must be included, or there must be a publically available means of obtaining the source code, for a relatively low reproduction cost.

Derived works

The software must allow under its licence, modifications and derivations of the original software. The distribution of any derived or modified software must also comply with the same terms.

Integrity of the author's source code

If the license permits the distribution of "patch files" along with the source code, then the license may restrict the source-code from being distributed in an altered state. However, it must be expressed that the permit allows for the distribution of the software built from a modified source code. It may be required that any altered works be communicated through the use of a different name or version number.

No Discrimination Against Persons or Groups

No Discrimination Against Fields of Endeavour

It must be permissible to allow for the use of the program in any specific field of endeavour.

Distribution of License

The rights accompanying the software cannot be changed or revoked due to redistribution. Nor should a user need to obtain a supplementary license.

License Must Not Be Specific to a Product

The software's afforded rights must not be contingent on the type of distribution it partakes in.

License Must Not Restrict Other Software

There must not be any restrictions on the type of software that can be distributed alongside the open source software.

License Must Be Technology-Neutral

The license must not require that its use be only available to certain technologies, platforms, or interfaces.

Introduction to Licenses

There are 2 main types of OSS licences: a Viral Licence and Non-Viral Licence.

1. Viral Licence/Copyleft Licence

A Viral Licence is the most common type of licence used regarding Open Source Software. Usually called a "copyleft licence," as "copyleft" is supposed to be the very opposite of "copyright" because the right attached to copying is left, abandoned.

A copyleft licence obliges all derivative works extracted from the primary software to be licensed in compliance with the terms of that first copyleft licence granted. In other words each derivative is bound by the same conditions in the first licence.

The author of a work protected by a copyleft license doesn't want to allow any future restriction regarding rights of copying, modifying or distributing his primary work all along its existence. Thus the software remains free without consideration of improvements or modifications it suffered.

But the reverse of this type of licence is that it never can be joined, associated or become a proprietary licence if the terms of the primary licence forbid it.

Putting your software in the public domain does not lead to a greater protection. Software evolving in the public domain is certainly accessible and free for everyone but thus it allows also anyone to make important changes and then distribute the derivative work with a copyright licence.

Which are the copyleft licences available?

A lot of copyleft licences are available. The most common is the GNU GPL, also said General Public Licence.

It's important to note that not all copyleft licences can comply with each other.

For instance, if the primary software was under a GPLv.3 licence a derivative work can be licensed under an Apache Licence v.2. However, it would not be able to be licensed under an Apache Licence v1.1.

Therefore it's mandatory to check on the GNU website or other specialised website the consistency of each licence.

2. Non-Viral Licence / Permissive Licence

In addition to a viral licence, a permissive licence doesn't spread automatically in all derivative works. Thus a work licensed under this kind of protection can, in the course of different modifications, either remain free and accessible to everyone or become a proprietary work whose distribution or use is charged.

Whereas copyleft licence cannot give birth to or be associated with any proprietary licence, permissive licences comply with any kind of licence. The aim of this licence is best viewed bearing in mind the open source philosophy as collaborative work, and unique creator philosophy as individual work.

Therefore an author can use a work protected under a permissive licence, then modify it greatly, and protect the new version under a copyright.

A wide range of permissive licences is available. The most used are the Licence BSD and licence MIT.

Examples of Licenses

The following are examples of licenses. They set out standard templates of terms and conditions that authors can use to share their rights in original works.

1. Creative Commons

The Creative Commons is a non-profit organisation that aims to facilitate the sharing of creativity and knowledge through free legal tools. They provide standardised legal codes for the sharing of licensed work with the public. The licenses provided by the Creative Commons work alongside copyrights to shift the traditional 'all rights reserved' approach to a newer, more open 'some rights reserved' approach to permissions granted by right holders. There are currently 6 available license types covering commercial and non-commercial use.

<http://creativecommons.org/licenses>

2. Open Data Commons (ODC)

Similar to Creative Commons mentioned above. It is run by the non-profit Open Knowledge, which aims to unlock data to allow greater access to information and the ability to use it effectively; without legal, technical or social restrictions. ODC currently provide 3 types of licenses that protect 'databases', as opposed to 'content' that most other licenses protect.

<http://opendatacommons.org/licenses/>

3. Open Government

The Open Government License (OGL) facilitates the re-use of public sector information. Governmental departments and bodies making their information public under the license must comply with the OGL terms, without the need to 'apply' for the license. Users are authorised under the terms of the OGL to use the information anywhere in the world, for both commercial and non-commercial purposes. The aim of the OGL is to encourage the use of information under the license freely and flexibly, with minimal restrictions.

<http://www.nationalarchives.gov.uk/information-management/re-using-public-sector-information/re-use-and-licensing/what-ogl-covers/>

4. Homebrew

Homebrew is a form of open source package management software for the Mac operating systems. It helps users install software packages by downloading the software's source codes and any dependencies required into a common location for easy management.

Homebrew employs a BSD license to allow contributors to expand the support available to the various software packages. The BSD license was first used for Berkeley Source Distribution (BSD) UNIX, an update of the open source UNIX operating system created in the 1960s. It is relatively simple and has far less restrictions than other classes of licenses.

<http://opensource.org/licenses/BSD-3-Clause>

Multiple Licensing

OSS can be distributed under two or more different software licenses or sets of licenses. Prefixes may be used to indicate the number of licenses used (e.g. dual-licensed for software licensed under two different licenses).

Multiple Licensing As A Business Model

The copyright owners of open source software may license that same software under a non-open source license. This means that its software is made available simultaneously both under an open source licence and under a different licensing scheme that may incur a license fee.

The most common reason for choosing a non-free license is to avoid the restrictions that an open source license can impose. This can occur, for example, when the open source code will need to be re-used within a proprietary software product. Some open source licenses do not allow this; instead the owner of the copyright allows you only to use their open source software if you are using it in and/or for a product that will be open source also.

A popular example is MySQL data base, released under the GNU General Public License (GPL) and also under a separate commercial license for any developers that do not want to be obliged to release their software under the GPL license.

Multiple Licensing For License Compatibility

Another application of the multi-licensing model is the case of a developer desiring to fuse elements from two or more sources into a new software product. With multiple licensing it is possible to ensure compatibility among open source licenses, allowing code from differently licensed free software projects to be combined, or to provide users the preference to pick a license.

A notable example is the Mozilla Foundation's decision to implement a tri-licensing model to license certain software under the Mozilla Public License (MPL), the General Public License (GPL) and the Lesser General Public License (LGPL) in an effort to avoid incompatibility with the GPL, before the GPL-compatible MPL 2.0 made the tri-licensing unnecessary.

This toolkit was drafted by students from the Centre for Commercial Law Studies, Queen Mary University of London: James Conroy, Mirabella Foo, Marie-Capucine Nicolas, Maria Alessandra Russo, Bahdan Sarazhynski and Khareem Sawwaf ©