Free and open-source software

Jeszenszky, Péter
University of Debrecen, Faculty of Informatics
jeszenszky.peter@inf.unideb.hu

Kocsis, Gergely
University of Debrecen, Faculty of Informatics
kocsis.gergely@inf.unideb.hu

Basic definitions

- *Proprietary software*
- *Free software*
- *Open-source software*
- FOSS – *free and open-source software*
- *Commercial software*
- Software license
- EULA – *end-user license agreement*
- *Copyleft*, *copyleft* license
- License compatibility
- *Public domain*
- *Freeware*, *shareware*
- *Dual licensing, Multi-licensing*

Proprietary software

- The use, distribution and modification is restricted or enabled only with permission
  - *The term* ***closed source software*** is also used for these
  - They form the complementary set of free and open source software

Free software

A program is free software if the program's users have the four essential freedoms:

0) The freedom to run the program as you wish, for any purpose (freedom 0).

1) The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.

2) The freedom to redistribute copies so you can help your neighbor (freedom 2).

3) The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.

- See: *The Free Software Definition*
  http://www.gnu.org/philosophy/free-sw.html

Open-source software (1)

- A different definition of free software
  - definition: *The Open Source Definition* http://www.opensource.org/docs/osd
- „Open source” sounds better from the point of marketing than „free”
  - The word „free” can be misunderstood because one meaning of it is that you „do not have to pay”

Open-source software (2)

- Followers of the free software definition emphasize the ethic aspects of free software while those who prefer the definition open source emphasize the technological advantages
- Beside all the disagreements between the two aspects the two definitions practically mean the same.

Open-source software (3)

- *Open Source Initiative* (OSI)
  http://www.opensource.org/
  - A non-profit initiative founded in 1998 for the support of open source software
  - Till 2005 the leader of it was Eric S. Raymond, a founder of the open source movement

Open-source software (4)

- See more:
  - Richard M. Stallman. *Why “Free Software” is better than “Open Source”.* http://www.gnu.org/philosophy/free-software-for-freedom.html
  - Eric Steven Raymond. *The Cathedral and the Bazaar.* February 18, 2010. http://www.catb.org/esr/writings/cathedral-bazaar/

FOSS – free and open-source software

- A category that includes free and open source software
  - Also used: free/libre/open source software (FLOSS)
- See: http://freeopensourcesoftware.org/

Commercial software

- Software developed to be sold on the market for financial benefits
- Most commercial software are not free however there are several exceptions

Software license

- A legal formula that regulates the ways of using and distributing software
  - In case of non-free software we use the term

    EULA – end-user license agreement

Free and open-source licenses (1)

- Several free and open-source licenses exist
  - See on the page of Open Source Initiative e.g:
  - Open Source Licenses http://www.opensource.org/licenses
  - See more: *Various Licenses and Comments about Them* http://www.gnu.org/licenses/license-list.html

Free and open-source licenses (2)

- On the lists of FSF (Free Software Foundation) and OSI (Open Source Initiative) all the frequently used licenses are available
- Most of them are on both lists but there are exceptions like: *NASA Open Source Agreement v1.3 (accepted by OSI but not accepted by FSF)* https://opensource.org/licenses/NASA-1.3

Permissive free and open-source licenses (1)

- A subset of free and open-source licenses
- Their characteristic attribute is that they do not restrict too much the use and distribution of software
- The term ***non-copyleft*** is also used for them

Permissive free and open-source licenses (2)

- Example:
  - *Apache License 2.0* https://www.apache.org/licenses/LICENSE-2.0
  - BSD licenses:
    - *3-Clause BSD License*/*New BSD License*/*Modified BSD License* https://opensource.org/licenses/BSD-3-Clause
    - *Simplified (2-Clause) BSD License*/*FreeBSD License*) https://www.freebsd.org/copyright/freebsd-license.html
  - *X11 License* (*MIT License*) https://opensource.org/licenses/MIT

Permissive free and open-source licenses (3)

- All these licenses above allow the use of their subject software for commercial software
  - Also including the change of the source code and the building of the modified code into the commercial software

Permissive free and open-source licenses (4)

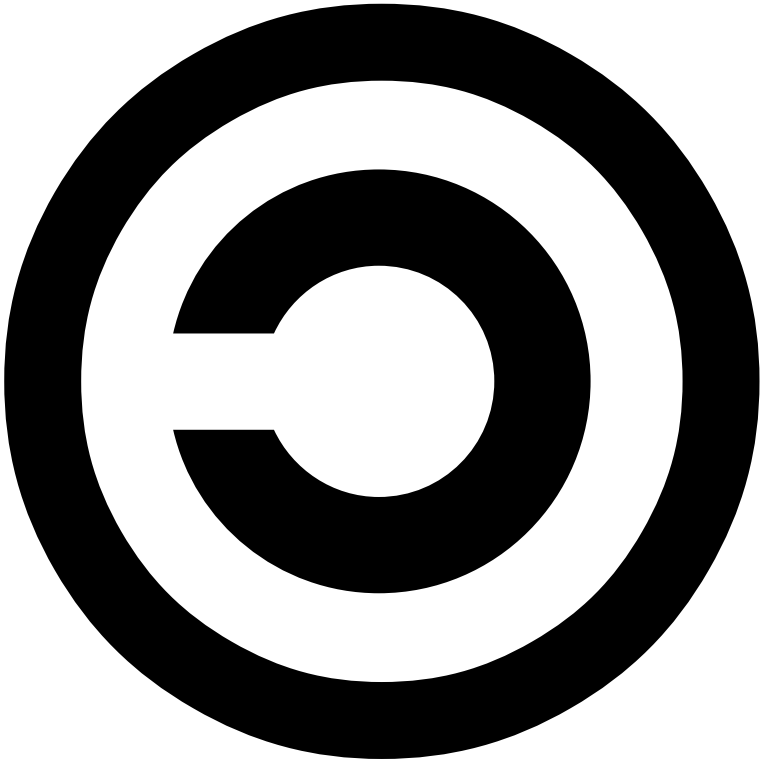
- Example:
  - Apache Maven (license: Apache License 2.0) https://www.apache.org/licenses/
  - FreeBSD (license: FreeBSD License) https://www.freebsd.org/copyright/freebsd-license.html
  - Gradle (license: Apache License 2.0) http://gradle.org/license/
  - HSQLDB (license: 3 clause BSD license) http://hsqldb.org/web/hsqlLicense.html
  - MINIX 3 (license: MINIX 3 License) http://www.minix3.org/doc/faq.html#legal
  - PostgreSQL (license: PostgreSQL License) http://www.postgresql.org/about/licence/

Copyleft (1)

- The word is a result of a wordplay with *copy**right***
- It means a legal solution for free and open-source software for an opposite aim as what copyright has
- By the use of it a software can be made an „inalienable public property”
- The idea is from Richard M. Stallman (1985)
- Definition: *What is Copyleft?* http://www.gnu.org/copyleft/

Copyleft (2)

- Symbol:
  - Source: *Wikimedia Commons* / Public domain https://commons.wikimedia.org/wiki/File:Copyleft.svg



Copyleft (3)

- Guarantees that all the products derived from the copylefted product will stay free
  - This means that the derived software is to be distributed with the same parameters as the one that is the subject of the license
- First implementation: Emacs General Public License
  - See: http://www.free-soft.org/gpl_history/emacs_gpl.html
- This solution is not used in all free software licenses

Copyleft license (1)

- Software licenses using the copyleft legal formula
- They form a disjunctive set from permissive licenses but still a subset of free and open source licenses
- There are two different types of them: strong and weak copyleft licenses

Copyleft license (2)

- **Strong *copyleft* license**: *copyleft* is used without restrictions so these software can not be used as parts of commercial software.
  - Example:
    - GNU General Public License (GNU GPL) v2.0 http://www.gnu.org/licenses/old-licenses/gpl-2.0.html
    - GNU General Public License (GNU GPL) v3.0 http://www.gnu.org/licenses/old-licenses/gpl-3.0.html
- **Weak *copyleft* license**: provides a possibility to use the software to create products that are not subjects of copyleft

Weak copyleft license (1)

- Typically used e.g. for program libraries
  - Make it possible to link the library to the software but the copyleft does not hold for the product
    - Typically dynamic linking is allowed without restrictions

Weak copyleft license (2)

- Example:
  - *GNU Lesser General Public License* (GNU LGPL) http://www.gnu.org/copyleft/lesser.html
    - Usage:
      - glibc https://www.gnu.org/software/libc/
      - GTK+ http://www.gtk.org/
      - Hibernate http://hibernate.org/community/license/
      - Qt https://www.qt.io/licensing/
  - *Mozilla Public License* http://www.mozilla.org/MPL/2.0/
    - Usage:
      - SpiderMonkey https://developer.mozilla.org/en-US/docs/Mozilla/Projects/SpiderMonkey

Use of software with GNU GPL license (1)

- *Copyleft* licenses let software using the subject of the license to be distributed only with special conditions that cannot be accepted for commercial software
- The use of the software however is not restricted
  - Example: Blender https://www.blender.org/
    - A software with GNU GPL v3.0 license for 3D products created with it copyleft is not held any more

Use of software with GNU GPL license (2)

- Generally the GNU GPL license hold for the outputs of the software subject if they contain parts of it.
  - For example while a graphical software can be used without restrictions the program Autoconf will not be such if there would not be an exception for it.
  - See: *Frequently Asked Questions about the GNU Licenses – In what cases is the output of a GPL program covered by the GPL too?* http://www.gnu.org/licenses/gpl-faq.html#WhatCaseIsOutputGPL

License compatibility

- Software licenses are compatible if it is allowed to combine the subject software (or parts) of them

GNU GPL compatibility (1)

- GNU GPL compatible free software license:
  - For distribution the GNU GPL license is to be used for the combination as well
  - Examples are::
    - Apache License v2.0 (just GNU GPL v3 compatible)
    - 3 clause BSD license, 2 clause BSD license
    - *Mozilla Public License v2.0*
    - *X11 License*
  - See more: http://www.gnu.org/licenses/gpl-faq.html

GNU GPL compatibility (2)

- Not GNU GPL compatible free software licenses are e.g.:
  - *BitTorrent Open Source License* http://www.bittorrent.com/license
  - *Eclipse Public License v1.0* https://www.eclipse.org/legal/epl-v10.html
  - *Mozilla Public License v1.1* https://www.mozilla.org/MPL/1.1/
  - *PHP License v3.01* http://php.net/license/3_01.txt

Weakening GNU GPL with exceptions (1)

- GNU GPL v3.0 enables the use of exceptions that make the license more permissive for one or more conditions
  - Examples:
    - *GCC Runtime Library Exception*
      http://www.gnu.org/licenses/gcc-exception-3.1.html
    - *GNU Classpath Exception*
      http://www.gnu.org/software/classpath/license.html
    - *GPL Font Exception*
      http://www.gnu.org/licenses/gpl-faq.html#FontException

Weakening GNU GPL with exceptions (2)

- *GCC Runtime Library Exception*:
  - An extension of GNU GPL v3.0 that is used for some program libraries distributed together with GCC Class path
  - (e.g: libgcc and libstdc++-v3)
  - When these libraries are linked to a program the product will not be a subject of GNU GPL
  - (for `glibc` library is GNU LGPL v2.1 is used for distribution)

Weakening GNU GPL with exceptions (3)

- *GNU Classpath Exception*:
  - An extension of GNU GPL that is used e.g. inOpenJDK
    - It enables Java software built on it to be used free
- *GPL Font Exception*:
  - An extension of GNU GPL for the distribution of fonts
  - It lets fonts to be embedded to products without the use of GNU GPL

GNU LGPL problems (1)

- GNU LGPL makes it really hard (or even impossible) to develop non-free software using elements that are subjects of the license
- GNU LGPL generally lets only linking of program libraries for other software
  - This has to be done in restricted way however!
    - E.g. only dynamic linking is enabled

GNU LGPL problems (2)

- In the followings under library we mean a library that is a subject of GNU LGPL license
- Information is for GNU LGPL 2.1
  - There is a newer version (3.0) however the version 2.1 I still used widely
  - There is no major difference between the two version from the aspect of the following information

GNU LGPL problems (3)

- ***Work that uses the library***: all source code that is developed to work linked together with the library
  - These works are not subjects of the license
  - If the work is linked statically to the library the product is a subject of the license

GNU LGPL problems (4)

- For distributing the product that is a result of linking the library to the work the followings have to be possible:
  - The user must be able to modify the library (for a newer version for example) and restructure the program
  - For this reverse engineering of the code has to be made possible
- For these the object code files have to be provided (however the source is not needed)

GNU LGPL problems (5)

- An exception from the above is distribution of the work in object file form.
  - This exception is not held if the object file uses longer than 10 lines macros or inline functions from the library
    - Modern libraries use such inline functions very often
  - As a result this is just a theoretical option

GNU LGPL problems (6)

- According to Free Software Foundation (FSF) the above information is held also for Java applications using class libraries distributed under GNU LGPL license
  - See: David Turner. *The LGPL and Java*. http://www.gnu.org/licenses/lgpl-java.html

GNU Affero General Public License (GNU AGPL) (1)

- A strong copyleft license based on GNU GPL for the distribution of server software
  - Current version: http://www.gnu.org/licenses/agpl-3.0.html
  - The basic GNU GPL lets the subject software to be reached on a server without sharing the source code (e.g. through a web application)

GNU Affero General Public License (GNU AGPL) (2)

- It requires that the owner of the server shares the source code of the modified version running on the server
- It is mandatory to publish the source code of the modified version running on the server.

GNU Affero General Public License (GNU AGPL) (3)

- Usage:
  - Ghostscript http://www.ghostscript.com/Licensing.html
  - MongoDB Database Server and Tools https://www.mongodb.com/community/licensing
  - Oracle NoSQL Database Server (Community Edition) http://www.oracle.com/technetwork/database/database-technologies/nosqldb/overview/index.html

The Java license

- Oracle distributes JDK and JRE for Java SE platform under *Oracle Binary Code License Agreement* http://www.oracle.com/technetwork/java/javase/terms/license/
- The license lets JDK and JRE to be used for the development of commercial software (and also the distribution of JDK and JRE together with the software)
- Java SE OpenJDK implementation can be reached under GNU GPL v2.0 and *GNU Classpath Exception* licenses

Public domain (1)

- An intellectual property that is not under copyright
  - The copyright is expired or the author declares so
- All software available in source code format are non-copyleft license free software in the same time

Public domain (2)

- Usage:
  - MinGW Runtime http://www.mingw.org/license
  - SQLite https://www.sqlite.org/copyright.html
  - XZ Utils http://tukaani.org/xz/
  - X-12-ARIMA https://www.census.gov/srd/www/x13as/

Freeware, shareware

- ***Freeware***:
  - It has no officially accepted meaning
  - It is used for software that are free to be distributed but cannot be modified (source code is not available)
  - Not free software
- ***Shareware***:
  - Software that is free to be distributed but the use is typically possible under special conditions. The aim of them is to generate income for the author.
  - Not free software

How to choose a free and open-source license

- The license has to be selected carefully since later it cannot be changed
  - For those who have started using the software the right to run the software cannot be withdrawn
  - Changing the license has a meaning only if the new license is more permissive than the previous was
  - E.g. GNU GPL v3.0 declares *irrevocable*

Why to develop free and open-source software?

- A community can be created for the software
  - Makes it more easy for new technologies and standards to spread
- The community of the software can be used
  - They can test the software in order to find bugs
  - People who tried the software will become customers of the commercial version more likely
- The producer can sell hardware for the users

Commercial development of free and open-source software

- Examples of commercial software developers:
  - Adobe http://www.adobe.com/open-source.html
  - Apple http://www.apple.com/opensource/
  - Google https://developers.google.com/open-source/
    - e.g.: Android http://www.android.com/
  - HP: HP Linux Imaging and Printing http://hplipopensource.com/
  - Intel https://software.intel.com/en-us/oss
  - Oracle http://oss.oracle.com/
    - e.g.: MySQL http://www.mysql.com/
  - Red Hat https://www.redhat.com/en/open-source
  - …

GNU Project (1)
http://www.gnu.org/

- Richard M. Stallman announced in 1983 the aim of creating a free Unix-like operating system
  - *About the GNU Project – Initial Announcement* http://www.gnu.org/gnu/initial-announcement.html
- Name: GNU's Not Unix! (GNU)
- *The GNU Manifesto* (1985) http://www.gnu.org/gnu/manifesto.html

GNU Project (2)

- The development was started in 1984 by Stallman
- First steps: C compiler (GCC), text editor (Emacs)
- 1985 - FSF was founded by Stallman for financial support of the GNU Project
- 1989 - GNU GPL license introduced by Stallman
- GNU software: http://directory.fsf.org/wiki/GNU
  - Example: Bash, Gawk, GCC, GIMP, GTK+, GNOME, Make, GnuPG, Midnight Commander, R, …

GNU/Linux (1)

- For the early 90's the GNU operating system is almost done however the kernel is still missing
- 1991 - Linus Torvalds starts to work on the kernel known as Linux,
- In 1992 he changes to GNU GPL license
- According to FSF the proper name of software that are products of the linking of GNU software and Linux kernel is: GNU/Linux system
  - See different Linux distributions (e.g.: Debian, Fedora, Gentoo, Ubuntu, …)

GNU/Linux (2)

- A kernel developed under GNU Project: GNU Hurd http://www.gnu.org/software/hurd/hurd.html
  - The development started at 1990 and still cannot be used because of bugs
- See more:
  - *Linux and the GNU Project* http://www.gnu.org/gnu/linux-and-gnu.html
  - *GNU/Linux FAQ* http://www.gnu.org/gnu/gnu-linux-faq.html
  - *Free GNU/Linux distributions* http://www.gnu.org/distros/free-distros.html (only 11 distributions!)
    - E.g.: gNewSense (Ubuntu based distribution) http://www.gnewsense.org/

Free Software Foundation (FSF) http://www.fsf.org/

- Nonprofit company for the purpose of supporting free software, chair: Richard Stallman
  - Free Software Definition
  - Free software philosophic essays
  - Publishing software licenses (GNU GPL, GNU LGPL, GNU AGPL, GNU FDL)
  - Publishing free software related information
    - Example: *Free Software Directory* http://directory.fsf.org/
  - Managing copyright of free software
  - Supporting GNU project
  - Free software advertising campaigns

Some current FSF campaigns

- *Campaign for OpenDocument* http://www.fsf.org/campaigns/opendocument
- *DefectiveByDesign.org – We oppose DRM* http://www.defectivebydesign.org/
- *End Software Patents* http://endsoftpatents.org/
- *Free BIOS Campaign* http://www.fsf.org/campaigns/free-bios.html
- *PlayOgg* http://playogg.org/
- *Upgrade from Windows* http://www.upgradefromwindows.com/

Open-source business models: dual licensing (1)

- Widespread business solution. The producer provides the same software under more different licenses
  - Even more than two different licenses can be used. This is called ***multi-licensing***

Open-source business models: dual licensing (2)

- The use of more licenses in the same time:
  - For free and open source uses the software may be available with a proper free and open-source license
  - For commercial purposes the producer can provide the software with given conditions. This is typically done for financial benefits.

Open-source business models: dual licensing (3)

- Two different models:
  - The producer provides the same software with different licenses
    - This solution is used e.g. by Oracle for Berkeley DB products http://www.oracle.com/technetwork/products/berkeleydb/
  - The producer provides an enhanced version of the software for financial benefits.
    - This solution is used e.g. by Oracle for MySQL products http://www.mysql.com/

Open-source business models: dual licensing (4)

- The free and commercial versions are often distinguished by their names: *Community Edition* and *Enterprise Edition*
  - Example:
    - Liferay Portal (Liferay Inc.) https://www.liferay.com/downloads
    - MySQL (Oracle) https://www.mysql.com/downloads/
    - Neo4j (Neo Technology Inc.) http://neo4j.com/download/
    - RapidMiner Studio (RapidMiner) https://rapidminer.com/products/comparison/
    - …

Open-source business models: dual licensing (5)

- For closed source and other commercial software this is the only available solution because not all open-source licenses lets such usage
- See:
  - Mikko Välimäki. *Dual Licensing in Open Source Software Industry.* 2003. http://opensource.mit.edu/papers/valimaki.pdf
  - Karl Fogel. *Producing Open Source Software – How to Run a Successful Free Software Project.* http://producingoss.com/
    - See Chapter: *Dual Licensing Schemes* http://producingoss.com/en/dual-licensing.html

Open-source business models: dual licensing (6)

- Business opportunities in dual licensing:
  - If the user wants to use the software for purposes that is not permitted by the free and open source license the commercial version is needed to be bought
  - The producer provides support for the software as a standalone service

The use of other digital works

- For development not only software but other digital contents may be used
  - E.g.: Fonts, icons, music, …
- Since software are also digital contents, hereafter we use *other* (digital) work for non-software digital contents
- Contents published for widespread use are typically published with some sort of license like in the case of free and open source software

Use of software licenses for digital works

- The use of other works is often controlled by well known free and open-source software licenses
- In closed source developments contents under permissive software licenses can be used

GNU GPL as a license for other works (1)

- GNU GPL can be used for all contents for which the term „source code” can be interpreted
- See: *Frequently Asked Questions about the GNU Licenses – Can I use the GPL for something other than software?* http://www.gnu.org/licenses/gpl-faq.html#GPLOtherThanSoftware

GNU GPL as a license for other works (2)

- The use of it would make it impossible to use the works in closed source software
  - E.g. products using fonts with this license would also be subjects of GNU GPL
- This is one reason why *GPL Font Exception* Extension was added
  - See: David „Novalis” Turner. *Font Licensing*. April 25, 2005. http://www.fsf.org/blogs/licensing/20050425novalis

Licenses for digital works

- There are several standard licenses directly for other digital contents
  - Some licenses are for some given type of contents (e.g. fonts) while others can be used for different ones
  - Copyleft is also popular solution for licensing other works
  - *Creative Commons* licenses can be used for many different works (music, video, text)
- See: *Various Licenses and Comments about Them – Licenses for Other Works* http://www.gnu.org/licenses/license-list.html#OtherLicenses

Creative Commons licenses (1)

- For other works *Create Commons* licenses are particularly popular https://creativecommons.org/
  - Creative Commons has not only one license. There are several different ones.
  - Their developer is the Creative Commons nonprofit organization

Creative Commons licenses (2)

- Three layers:
  - ***Legal Code***
  - ***Commons Deed***
  - ***Machine readable***
    - *CC Rights Expression Language* (CC REL)
      https://wiki.creativecommons.org/wiki/CC_REL

Creative Commons licenses (3)

- See the licenses here: https://creativecommons.org/licenses/
  - Attribution (CC BY)
  - Attribution – (CC BY-SA)
  - Attribution – NoDerivs (CC BY-ND)
  - Attribution – NonCommercial (CC BY-NC)
  - Attribution – NonCommercial – ShareAlike (CC BY-NC-SA)
  - Attribution – NonCommercial – NoDerivs (CC BY-NC-ND)

Creative Commons licenses (4)

- For public domains:
  - CC0 http://creativecommons.org/choose/zero/
    (Publish works as public domain)
  - *Public Domain Mark*
    https://creativecommons.org/choose/mark/
    (For works already existing as public domain)

Creative Commons licenses (5)

- Attribution
  - You must give appropriate credit, provide a link to the license, and indicate if changes were made.
  - See: *Best practices for attribution* https://wiki.creativecommons.org/wiki/Best_practices_for_attribution

Creative Commons licenses (6)

- Tools and services for attribution:
  - flickr cc attribution bookmarklet maker http://cogdog.github.io/flickr-cc-helper/
  - ImageCodr.org http://www.imagecodr.org/
  - OpenAttribute http://openattribute.com/

Creative Commons licenses (5)

- ShareAlike
  - If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- NoDerivatives

  If you remix, transform, or build upon the material, you may not distribute the modified material.
- NonCommercial
  - You may not use the material for commercial purposes.

Creative Commons licenses (7)

- FSF about CC licenses:
  - CC BY: non-*copyleft* free license, recommended
  - CC BY-SA: *copyleft* free license, recommended
  - CC BY-ND, CC BY-NC, CC BY-NC-ND, CC BY-NC-SA: non-free licenses, not recommended
  - CC0: GNU GPL compatible license
- See: *Various Licenses and Comments about Them* http://www.gnu.org/licenses/license-list.html

Creative Commons licenses (8)

- More information:
  - *Creative Commons – Frequently Asked Questions* https://wiki.creativecommons.org/index.php/Frequently_Asked_Questions
  - Compatible Licenses https://creativecommons.org/compatiblelicenses/
    - ShareAlike! (CC BY-SA, CC BY-NC-SA) compatible licenses

Licenses for documentation

- The followings are often used for documentations:
  - *GNU Free Documentation License* (GNU FDL)
  - *Creative Commons*-licenses

GNU Free Documentation License (GNU FDL)

- Copyleft license by FSF for documentations
  - Current version: GNU FDL 1.3
    http://www.gnu.org/licenses/fdl-1.3.html
- Usage:
  - *The GNU C Library Reference Manual*
    https://www.gnu.org/software/libc/manual/
  - *Qt Documentation* http://doc.qt.io/
  - ...

Creative Commons-licensed documentation

- Example:
  - Scott Chacon, Ben Straub. *Pro Git*. 2nd edition. Apress, 2014. (license: CC BY-NC-SA 3.0) https://git-scm.com/book/en/v2
  - Daniel Stenberg. *http2 explained*. 2016. (license: CC BY 4.0) https://daniel.haxx.se/http2/
  - *PHP Manual* (license: CC BY 3.0) http://php.net/manual/
  - …

Software and services

- Solutions for supporting the usage of open-source software:
  - Free and open source software:
    - License Maven Plugin (programming language: Java; license: GNU GPL v3.0) http://www.mojohaus.org/license-maven-plugin/
    - FOSSology (programming language: PHP, C; license: GNU GPL v2.0) http://www.fossology.org/
    - ScanCode (programming language: Python; license: Apache License 2.0) https://github.com/nexB/scancode-toolkit/
  - Business services:
    - WhiteSource – Open Source Security & License Management http://www.whitesourcesoftware.com/

Further reading

- Heather J. Meeker. *The Open Source Alternative: Understanding Risks and Leveraging Opportunities*. Wiley, 2008.
- Van Lindberg. *Intellectual Property and Open Source: A Practical Guide to Protecting Code*. O'Reilly, 2008.