

Open Source License

ningjiang@apache.org

2012-10

Why does the software
need a license?

What OS license did you
hear about it?

What is the license for a book?

- ⦿ A book is not licensed, it is sold.
- ⦿ The sale of a book is not the sale of the copyright, it is the sale of a copy.
- ⦿ The rights of the owner of a copy of a copyright work are the copyright statute.
- ⦿ buying a book does not give a the right to copy the book.

What is a license

- ⦿ Permission by the owner of property to take some act that the owner has the ability to control
- ⦿ What is a sublicense?
 - ⦿ A license granted by a licensee
 - ⦿ The right to grant a sublicense must be expressly stated in a license and will not be implied copyright patent

What is the license for a book?

- ⦿ Fair use gives you certain rights to copy portions of the book.
- ⦿ The sale of used books demonstrated the difference between the sale of a copyrighted work and a license for copyrighted work.
- ⦿ Software has become a sale of a license not a sale of the software.

What is the same between commercial and Open source licenses

- ⦿ Both are based on ownership of intellectual property
- ⦿ Both grant certain rights and retain others
- ⦿ Both are governed by the same laws
- ⦿ Both may include provisions which may be incompatible with the obligations of other license
- ⦿ License obligations can be incompatible, but the issue is whether the obligations are triggered.

What is different about open source licenses?

- ⦿ Different goals
- ⦿ Written by developers not lawyers
- ⦿ Encourages uncontrolled combination and reuse
- ⦿ Certain legal issues regarding contract formation remain open for open source licenses
- ⦿ Open source licenses have no acceptance procedures
- ⦿ Some open source license impose sharing obligations on users

Open Source is not
Development
Methodology

Open Source = License

The license determines
the rules the community
lives by.

Difference Licenses
create different
communities.

Free Software & Open Source

The license captures the
community philosophy

Making Sense of OSS License

- ⦿ Goals
- ⦿ Philosophy
- ⦿ Example Uses
- ⦿ Community Implications

Why Not Public Domain?

Why Not Public Domain?

- ⦿ Warranty
- ⦿ Attribution
- ⦿ Requirements on distribution
- ⦿ Requirements on derivative works
- ⦿ To ensure code remains open source

A license is chosen to
reach certain goals

What goals?

License Goals

- ⦿ Ensure certain code remains open source
- ⦿ Reserve control
- ⦿ Build a commercial coalition
- ⦿ Encourage third-party marketplace
- ⦿ Challenge competitors

The Free Software Definition

- ⦿ The freedom to **run** the program for any purpose
- ⦿ The freedom to **study** how the program works, and **change** it to make it do what you wish
- ⦿ The freedom to **redistribute** copies so you can help your neighbor
- ⦿ The freedom to **improve** the program, and **release** your improvements (and modified version in general) to the public, so that the whole community benefits

Open Source Definition

- ⦿ Free Redistribution
- ⦿ Program must include Source Code and must allow distribution in source code as well as compiled form
- ⦿ Must Allow Modifications and Derived Works
- ⦿ Integrity of the Author's Source Code
- ⦿ No Discrimination Against Persons or Groups
- ⦿ No Discrimination Against Fields of Endeavor
- ⦿ Distribution of License
- ⦿ License Must Not Be Specific to a Product
- ⦿ License Must Not Restrict Other Software
- ⦿ License Must Be Technology-Neutral

Free Distribution

- By constraining the license to require free redistribution, we eliminate the temptation to throw away many **long-term gains** in order to make a few short-term sales dollars. If we didn't do this, there would be lots of pressure for cooperators to defect.

Source Code

- We require access to un-obfuscated source code because you can't evolve programs without modifying them. Since our purpose is to make evolution easy, we require that modification be made easy.

Derivative Works

- ⦿ The mere ability to **read source** isn't enough to support independent peer review and rapid **evolutionary selection**. For rapid evolution to happen, people need to be able to **experiment with and redistribute modifications**.

Integrity of Author's Source code

- ⦿ Encouraging lots of **improvement** is a good thing, but users have a right to know who is responsible for the software they are using. Authors and maintainers have reciprocal right to know what they're being asked to support and protect their reputations.

No Discrimination

- ⦿ The major intention of this clause is to **prohibit license traps** that prevent open source from being used commercially. We want commercial users to join our community, not feel **excluded** from it.

Distribution of License

- ④ This clause is intended to forbid closing up software by indirect means such as requiring a non-disclosure agreement.

License Not Specific to Product

- ⌚ This clause foreclose yet another class of license traps.

No Restriction of Other Software

- ⦿ Distributors of open source software have the right to make their own choices about their own software.

Technology Neutral

- This provision is aimed specifically at licenses which require an explicit gesture of assent in order to establish a contract between licensor and licensee

Most Common License

- ⦿ Apache License, 2.0
- ⦿ BSD license
- ⦿ GNU General Public License (GPL)
- ⦿ GNU Library or “Lesser” General Public License (LGPL)
- ⦿ MIT license
- ⦿ Mozilla Public License 1.1 (MPL)
- ⦿ Common Development and Distribution License
- ⦿ Eclipse public license
- ⦿ Artistic Licenses

Open Source License Families

- ⦿ Give Me Credit
 - ⦿ AL, BSD, MIT
- ⦿ Give Me Fixes
 - ⦿ MPL, EPL, LGPL
- ⦿ Give Me Everything
 - ⦿ GPL

Give Me Credit

- ⦿ Software “Commons”
- ⦿ Derivatives can sub-license
- ⦿ May have some conditions
 - ⦿ No Warranty
 - ⦿ Credit to original authors required
- ⦿ Apache (AL), BSD, MIT

Give Me Credit

- ⦿ Universal donor
- ⦿ Limited control by any one entity
- ⦿ Little value in direct commercial licensing
- ⦿ Allows for commercial product development
- ⦿ Allows for competing services

Give Me Fixes

- ⦿ File or derivative based conditions
- ⦿ Original author may have special rights
- ⦿ Differentiate between source and binary
- ⦿ Larger works can be under a different license
- ⦿ Mozilla(MPL), Eclipse(EPL/CPL), LGPL

Give Me Fixes

- ⦿ Single entity control
- ⦿ Still very “business friendly”
- ⦿ Encourage incorporation of code into larger work
- ⦿ Ensures direct development benefits all
- ⦿ Tends toward libraries or platforms

Give Me Everything

- ⦿ Copyleft
- ⦿ Derivative works remain under the license
- ⦿ Linked works may also remain under the license
- ⦿ Ensures all “down stream” have the same right
- ⦿ GPL

Give Me Everything

- ⦿ All direct development is contributed back
- ⦿ Contributors assured code remains open source
- ⦿ Encourages a full free software economy
- ⦿ Copyright holder retains much control
- ⦿ Limits commercial adoption
- ⦿ Dual-license business strategy

License Scope

- ⦿ Commercial Friendly
 - ⦿ AL > MPL > GPL
- ⦿ Free Software
 - ⦿ GPL > MPL > AL

Apache License

Apache License

- ⦿ Permissive License (Give me Credit)
- ⦿ Requires Attribution via a NOTICE file
- ⦿ No Trademark Grant
- ⦿ Grant of Patent License

Apache License Philosophy

- ➊ To allow the maximum use of our software for any purpose and by all people.

wrap up

- ⦿ What is license for software?
- ⦿ what is open source license?
- ⦿ open source license family
- ⦿ Apache license

Reference

- ⦿ <http://www.slideshare.net/markradcliffe/introduction-to-open-source-licensing>
- ⦿ <http://www.slideshare.net/jaaronfarr/making-sense-of-open-source-licenses>