

Licensing your work: How & Why

Protecting yourself and your users

Nathan Fish

Why licensing matters to you

- Let people enjoy your creations without legal issues
- Prevent your work from being misused or plagiarized
- Avoid lawyers and get work done

Copyright

- Copyright is automatic & implicit
 - Any intellectual work, the moment it's written down
- License depends on local law
- Author's intended license is unclear
- When publishing, include an explicit license
 - No confusion & international

Public Domain

- Works in the Public Domain have no copyright
 - Can be freely used in any way
 - Sherlock Holmes
- Not all countries allow authors to choose public domain – only government can!
- Use Creative Commons 0 (CC0) instead

Software patents

- A few countries allow software to be patented as well
- Patents are independent of copyright
 - Software released under a Free license can still have patents!
 - Allows “Patent treachery” - FSF
- Many Free licenses explicitly block patenting
 - GPL 2.0+

Definitions

Free/Libre Software: “Four Freedoms” checklist:

- 0 Run the program as you wish, for any purpose
 - 1 Study the program & modify as you wish
 - 2 Share copies to help others
 - 3 Share your modified copies with others
- 1 & 3 require source code

Definitions (2)

- Proprietary: Non-free – Cannot be used, copied, or modified freely (Fails any of the 4 Freedoms)
- Copyleft: Derivative works must copy license
 - Generally implies a permissive license
 - *Copyright* is used to take freedom away from users
 - *Copyleft* is designed to protect user's freedom
 - Free licenses may or may not be copyleft

Definitions (3)

- Open Source: A development model based on collaboration through access to source code
- Private Software: Software you develop and never release
 - If you modify Free, copyleft software and use it privately, you aren't required to give source to anyone

Commercial Software

- Commercial Software: Software developed as part of a business venture
 - Does not mean proprietary!
 - Apache, Linux kernel – Funded largely by companies
 - Red Hat, Ubuntu - commercial GNU/Linux distributions

Open vs Free

- The “Open Source” movement is about getting better software for everybody
 - A software development model
 - Access to source code enables mass collaboration
- Better software is a great goal – IF it serves the user
 - Open Source DRM software possible

Open vs Free (2)

- Free Software is about giving users control over their own computers
 - An ethical choice
 - Collaboration & better software is a great side effect
- These goals are often compatible, but are not the same
 - Most Open Source Software is also Free
 - Difference is the attitude of the community & developers

GPL-based licenses

- General Public License
 - Iconic & most popular Free Software license
- Lesser GPL (For shared libraries)
 - Allows linking libraries into non-free programs
- Affero GPL
 - Required to give source to remote / network users
 - Prevents Service as a Software Substitute (SaaS)

Licenses – Docs, Art, etc

- Creative Commons: Actually a set / toolkit of licenses
 - Lets you toggle a combination of options
 - You must specify which CC license, eg “CC-BY-SA 3.0” = Attribution & Share Alike (copyleft)
 - Use for music, art, etc – not documentation or code
- GNU Free Documentation License
 - Tutorials, manuals, textbooks, etc

Examples of Free Software

- Linux kernel: GPL 2.0 (copyleft)
- Apache: Apache 2.0 license (not copyleft)
- Firefox: MPL 2.0 & others
- VLC: GPL

Notes

- Originally presented at KWLUG, January 05, 2014
- Thanks to the Free Software Foundation and the GNU Project for their excellent resources on licensing, and their work to protect Free Software
- This presentation is released under CC-BY-SA 3.0
- Nathan Fish, 2015