

# Demystifying FOSS Licenses

by  
Harsh Gupta

Which of the following are you allowed to do, and why:

- You come across project X on github, it does exactly what you want but the project has no license, can you use it?
- You come across project X licensed under GNU GPL v3, you want to use it in your project Y licensed under BSD, can you use it?
- You come across project X licensed under BSD, you want to use it in your project Y licensed under GNU GPL v3, can you use it?
- You come across project X licensed under GNU GPL, you want to use it use it in your project Y licensed under BSD, instead of directly using X, you read its source and implement some of the idea in your project Y. Can you do it?

## Few more scenarios:

- There is a popular FOSS project, you really like the project but not its license, so you forked the project and changed its license. Can you do it?
- You are the original author and maintainer of an open source library, over the years the library has got popular and has attracted 100s of contributors, now you think your original license was not right, can you change the license with just a commit?

Lets find out the answers.

# Disclaimer

- I am not a lawyer and this is not legal advise
- When in doubt contact your lawyer
- The presentation is simplified representation of the matter. I have given preference to applicability over accuracy
- I focused on some basics which are hard to understand by yourself

# Copyright

You probably know that following are not allowed by law:

- Copying a DVD of a latest movie and selling it
- Scanning chapters of the latest book by JK Rowling and publishing them online
- “Pirating” Windows 7

- Copyright gives the creator of an **original literary, artistic, dramatic or musical work** the **exclusive right to create copies, offer to public or create derivatives, adaptations or translations** of that work.

In general Copyright law is damn complicated, here are few things you need to know:

- Software is copyrightable
- There is no need of registration of copyright, you get copyright on a work by act of creating it.
- Unless the person owning copyright over a work gives you explicit permission, you are not allowed to do any of things which are exclusive rights of the copyright holder, ie., copy it, modify it, distribute it or adapt it.\*
  - Example: You are not allowed to use code published on github which doesn't have any license
- Copyright applies to the “expression of an idea” and not the idea itself.\*
  - Example: If you create a the first word processing system, then the copyright will apply to only the code of the word processing system and not the idea of creating word processing systems itself. So, someone else can independently come up with their implementation of word processing system without infringing on the copyright of the original.

\*There are important exceptions



# What is a license?

- A license is a permission to do something which wouldn't otherwise be allowed to do. As copyright holder of a software you can dictate the terms under which the permission is given.
- Irrespective of the license the original author retains the copyright over their work
  - You cannot relicense someone else's code, even if it is open source
- Proprietary Licenses often explicitly state that you are not allowed to create copies, make modifications etc. Example: EULA (End User License Agreement) which you accept when you install Microsoft Office.
- Free and Open Source Software (FOSS) don't have such restrictions but might have other conditions. There are many FOSS licenses but I'll only talk about two i.e., BSD license and GNU GPL family

## GNU GPL: GNU General Public License

- A family of licenses which allows you to:
  - Use the software in whatever way you want
  - Study the software and make modifications to it
  - Make copies of the software and distribute them
  - Copy and distribute any modifications you make

On condition that any derivatives of the software are also distributed under the same license as the original software. (called “Copyleft” condition).

- This means any modifications to a GPL software or combination of a GPL software with a non GPL software has to be released under GPL.
- Popular software licensed under a license form GNU GPL family: Linux kernel, GCC, VLC media player, FFmpeg, pandoc, GNU Utils

- BSD was originally used for “Berkley Software Distribution” operating system.
- For a software licensed under BSD, you can make copies of the software, modify it, distribute it, including distribution of the modifications.
- There is no requirement that modification needs to be in BSD, hence BSD licensed software can also be combined with proprietary software. Such licenses are called “permissive” licenses
- Examples of BSD Licensed software: SciPy, Django, lot of tools in the python ecosystem

# License Compatibility

- When you using a software *A* in software *B*, you need to comply with the license of *A* while releasing *B*.
- If you cannot do so, then the licenses of *A* and *B* are not compatible.
- *A* is compatible with *B* does not imply *B* is compatible with *A*
  - Example: You can use a BSD licensed library in a GPL software, but not the other way round.
- Say you use two FOSS libraries *lib1* and *lib2* in a project. You need to comply with the terms of both the licenses.

Back to original questions

- Ques: You come across project X licensed under GNU GPL, you want to use it in your project Y licensed under BSD, can you use it?

- Ques: You come across project X licensed under GNU GPL, you want to use it in your project Y licensed under BSD, can you use it?
- Ans: No, unless you relicense your project under GPL.

- Ques: You come across project X licensed under BSD, you want to use it in your project Y licensed under GNU GPL, can you use it?



- Ques: You come across project X licensed under BSD, you want to use it in your project Y licensed under GNU GPL, can you use it?
- Ans: Yes, BSD allows you to do so

- Ques: You come across project X licensed under GNU GPL, you want to use it use it in your project Y licensed under BSD, instead of directly using X, you read its source and implement some of the idea in your project Y. Can you do it?

- Ques: You come across project X licensed under GNU GPL, you want to use it use it in your project Y licensed under BSD, instead of directly using X, you read its source and implement some of the idea in your project Y. Can you do it?
- Ans: Maybe. The distinction between idea and expression isn't strictly defined. Ultimately court will decide if you are allowed to do this.

- Ques: There is a popular FOSS project, you really like the project but not its license, so you forked the project and changed its license. Can you do it?

- Ques: There is a popular FOSS project, you really like the project but not its license, so you forked the project and changed its license. Can you do it?
- Ans: No, because irrespective of the FOSS License, the contributors hold the copyright for their code and only they can change its license.

(If this was not the case, the conditions of GPL would have been irrelevant)

- Ques: You are the original author and maintainer of an open source library, over the years the library has got popular and has attracted 100s of contributors, now you think your original license was not right, can you just change the license with a commit?

- Ques: You are the original author and maintainer of an open source library, over the years the library has got popular and has attracted 100s of contributors, now you think your original license was not right, can you just change the license with a commit?
- Ans: No. Same reason as the previous one, contributors hold the copyright for their contributions not you. In general, you need to explicitly take permission of every contributor about the license change. Or you need to have an agreement with asking them to assign their copyright on the contributions to you.
- Corollary: You cannot violate the license on your own code.

# Things to think about before choosing a license

- What is the aim of your software? Is it a library or something to be only used by end user?
- Do you have some kind of business model in mind?
- Which software do you want to build upon? Are the licenses of these software compatible?
- Which use cases do you want to allow?
- What communities do want to engage?



# Things I didn't talk about

- Patents
- Public domain and fair use

# Resource

- TL;DR Legal: Simple explanations of different FOSS Licenses <https://tldrlegal.com/>
- List of GPL compatible Licenses:  
<https://www.gnu.org/licenses/license-list.en.html>
- CopyrightX: MOOC on Copyright Law by Harvard Law School <http://copyx.org>

Thank You