Open Source Software

Lecturer: Taghreed Alreffaee

Lecture #6

Content

- Proprietary Software licenses.
- Open-Source Software Licenses.
- Copy left License.
- GPL License.
- Permissive License.
- MIT License
- Choosing the Right License.
- Business Model
- How do open source businesses make money

- **Proprietary Software licenses:**

Proprietary software is a restrictive software license that restricts the user to copy, modify or redistribute in any way. It is the most restrictive license that is out there and is generally used for commercial purposes.

- **Open-Source Software Licenses:**

Open Source Software licenses are those licenses where the owner or the creator of the software has copyright over his work but chooses to ease down some restrictions on his work with respect to the opening, modifying, or redistributing the software. Based on the degree of restrictiveness there are two main types of open source software licenses.

- When a person gets copyright of the software, his main intention is to prevent other people from distributing the software. Copyleft is exactly the opposite of copyright. When a person gets a copyleft, he is imposing on the user who modifies anything in the original software that they must keep their version of the software open-source so that it is publically available.

Most licenses fit into two categories:

- **copy left licenses** .
- **permissive licenses**.

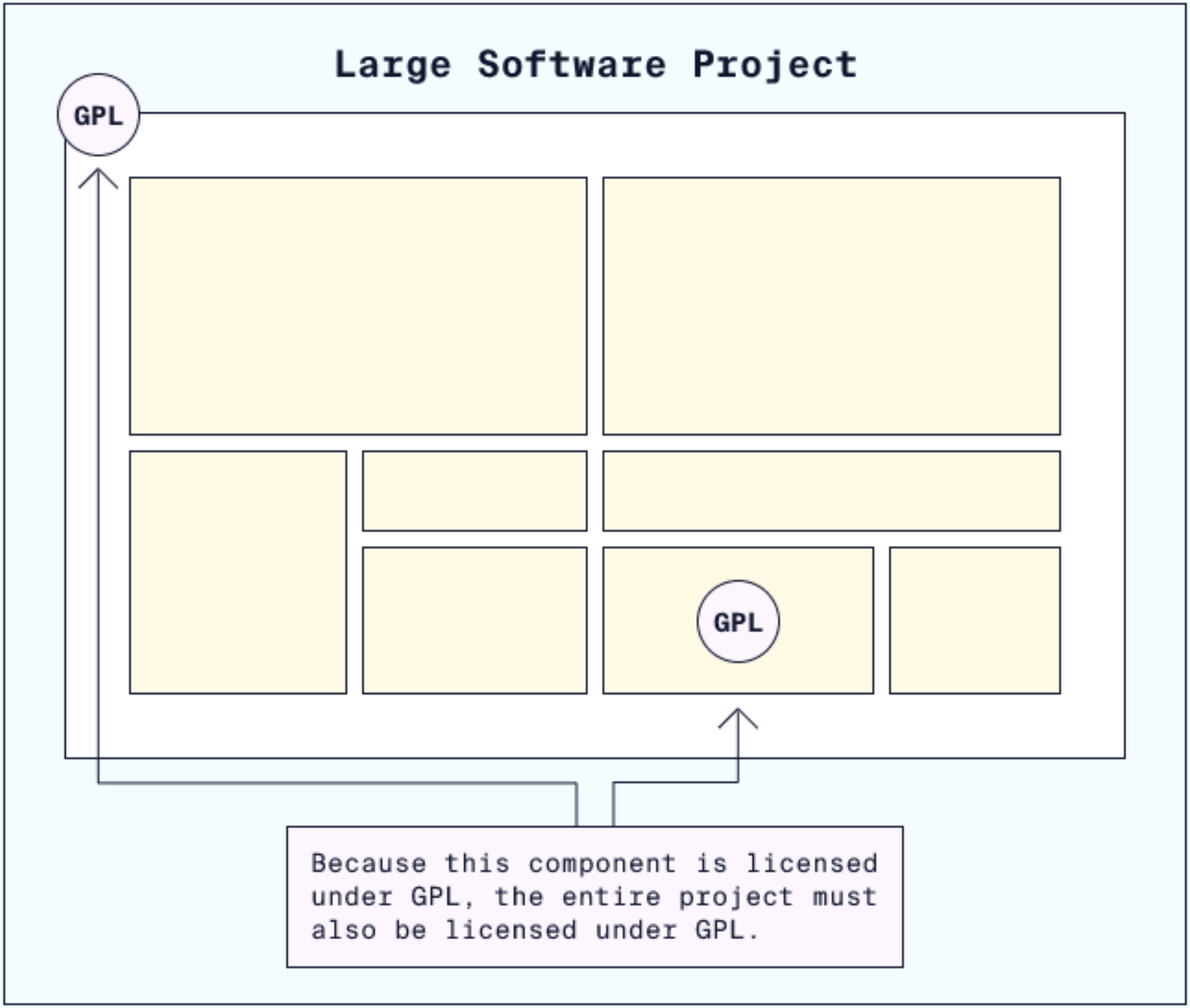
Depending on how you want other people to use your project.

Copyleft Licenses

- **A copyleft License** requires that any modified version of an open source project also be released under the same license as the original project.
- For example, if you released an open source project with a copyleft license, anyone who modifies your code must then release their code, including any of the changes they made, under the same license as your project. any developer who modifies or adds to the original software is generally required to share the source code of those changes when distributing software that includes copyleft-licensed code.
- The idea of copyleft licenses is fundamental to open source software. In fact, the first copyleft license was invented by the developers of the GNU operating system, the foundational project of the free software movement!

GPL

- A GPL(General Public License) is the most popular copyleft license. This license is based on the four freedoms.
- The original copyleft license (and oldest open source license in general) is the **GNU General Public License, or GPL**. GPL was created in 1989 by the same developers who built the GNU operating system, and is still one of the most popular open source licenses in use today. A 2015 analysis by GitHub showed that about 25% of GitHub projects used GPL version 2 or version 3.
- GPL is known as **strong copyleft license**. This means that any software that uses a project licensed under GPL must also be licensed under GPL. Because GPL is an open source license, this means that any project using even a bit of GPL-licensed software must also be open source


Large Software Project
GPL
GPL
Because this component is licensed under GPL, the entire project must also be licensed under GPL.

Permissive Licenses

- The other general category of open source licenses is **permissive licenses**. Permissive licenses are different from copyleft licenses in that they do not put restrictions on people modifying or redistributing a project.
- When one modifies a copyleft software, one is obligated to keep their new altered code under open source so it can be publically available. Whereas, when one modifies a permissive software. The user has the freedom to use the altered code however one wishes to, one also has the freedom to make it proprietary software.

- Typically, software under a permissive license can be modified, copied, added to, subtracted from, etc. without any obligation to share those updates. Developers can take the permissive-licensed software, make it their own through changes or additions, keep their new version to themselves, or share it if they choose to. This is a major feature if you’re looking to create proprietary software that you can sell and keep secret from competitors, and one of the main reasons why permissive licenses are popular.

- ✓ MIT licenses are the most popular and commonly used permissive software licenses.
- ✓ This license gives users of software permission to reuse the source code for any purpose, sometimes even if their code is part of proprietary software.
- ✓ As long as users include the original copy of the MIT license in their distribution, they have the freedom to make any changes or modifications to the code to suit their own needs and they can also use it for commercial purposes.
- ✓ In the case of permissive licenses, the owner of the original source code is not liable for any claims or liabilities.
- ✓ Other popular permissive licenses include: Apache license, Berkeley Source Distribution License(BSD)

- ✓ the MIT license does not impose many restrictions: anyone can copy, modify, or redistribute MIT-licensed code however they want.
- ✓ It permits redistribution with software other with code that has another license.
- ✓ It allows redistribution of the derivative works with non-MIT license.
- ✓ It is OSS Licenses and Because of its simplicity, the MIT license has become a popular choice, with 44% of GitHub projects using the license in 2015

Choosing the Right License

Now that you have all this information, how do you choose the open source license? There are a few considerations to make:

- First, you should ask yourself if you would be comfortable with someone else creating or distributing a proprietary version of your project. If you are, then a permissive license might be right for you. If you're not, then a copyleft license would probably be better suited.
- At this point, you may be wondering if you even need to choose a license at all! Many people make the assumption that if they don't add an open source license to their project, then people are free to use their code however they want. Actually, the opposite is true: if you add a license to your project, you are legally preventing people from using or modifying your code.

- When it comes to choosing an individual license within one of these categories, it can help to look at other projects in the same ecosystem to see what licenses they use. And remember, if any of your project's dependencies are licensed under GPL, you must also license your project under GPL or replace that dependency.

- If you do not want others to modify your code, then it is okay to omit an open source license on your project. Otherwise, it is important to include a license specifying what others can and cannot do with your code.

- All original code that we write falls under copyright, meaning that nobody else can copy or distribute someone else's original work. By adding open source licenses to our projects, we give others explicit legal permission to use and modify our code.
- If you're interested in using a copyleft license, you have to decide if you'd like your open source code to be used in proprietary software (likely made by private companies). If so, a weak copyleft license will be a good fit. Otherwise, use a strong copyleft license.

What is a business model?

A business model is an outline for how your company plans to make money. In general, a business model explains four things:

- What product or service a company will sell.
- How it intends to market that product or service.
- What kind of expenses the company will face.
- How the company expects to turn a profit

How do open source businesses make money

- An open source business can not charge a licensing fee. There are many different dynamic ways in which a company can earn revenue and ensure a feasible, profitable and sustainable business model. Here are three of the most frequent revenue streams they explored and adopted:-

1. Consulting services.
2. Charge for additional or fringe services.
3. Sale of additional proprietary products

Thanks for listening