

《开源软件设计与开发》课程总结

姓名：范泽秋 学号：51195100007

1. 开源理解

开源软件的发展大致经历了三个阶段。第一阶段是上世纪 80 到 90 年代，这个阶段是以个人和大学的技术极客为主的小众市场；第二阶段主要是互联网时代，Linux 和 Apache Web 服务器在互联网业务占据了统治地位；第三阶段是 2007 年云计算出现后，商业模式和市场需求推动了开源业务出现新的变化，人们开始考虑用开源作为工具，满足商业和变现的诉求^[1]。

事实上与开源相关的有三个方面：商业软件（Business Software）、自由软件（Free Software）和开源软件（Open Source Software，此处为狭义的开源，以下简称 OSS）。它们之间的根本区别并不是在是否收费上，这是国内对开源的最大误解。这三者的根本区别在于 License。

商业软件用的是商业 License，以保障软件商的利益为第一位，基本没有考虑用户的利益。具体可以查阅手上的商业软件 License。

自由软件用的是 GPL，目的是在最大程度上保障用户的自由。这种自由表现在，用户除了可以自由使用软件以外，还可以自由分发和自由修改。但是因为 GPL 的派生性（就是说基于 GPL 软件派生的软件也要使用 GPL），用户对自由软件作修改和再分发后，用户也成了作者之一，所以 GPL 对用户自由的保障其实也保障了作者的权益。

而 OSS 类似于自由软件，但用的 License 更加宽松，并不局限于 GPL。只不过这些 License 有一个共同点就是开放源代码，赋予用户更多使用自由的同时，还赋予用户选择的自由。

既然如此，我们就可以很明白地看出来：开源软件并不排斥商业。不论是自由软件还是 OSS，都可以在其基础上通过提供增值服务取得商业利益，只要不违反其 License 就行。所以你完全可以把下载来的 Linux 刻成光盘拿出去卖，当然如果你修改了下载的内容，那么发行时要带上修改过的源代码。事实上，在不违反 License 的情况下，你可以用开源软件进行一切你能想到的商业活动。

但是在对待商业软件的态度上，自由软件与 OSS 还是有区别的。对于自由软件来说，商业软件是另一个世界的东西，二者是井水不犯河水，自由软件走的是独立自主的道路，不允许商业软件进入自由软件体系，也禁止自由软件成果进入商业软件中（因为商业软件肯定不是用 GPL，这违反了 GPL 的派生规定）。

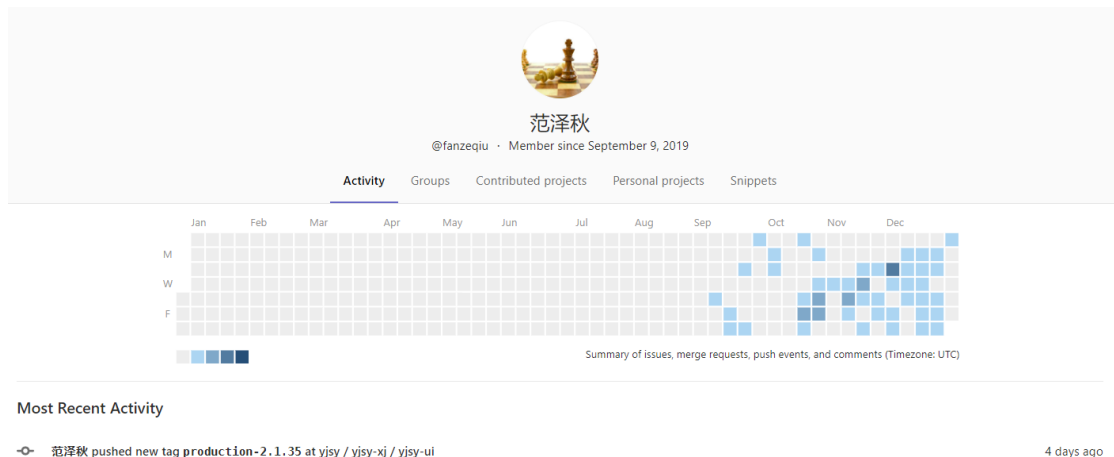
而对于 OSS 来说则要宽松得多，一般没有派生限制，很多开源 License 都允许其派生成果不开源，即允许这些成果进入商业软件。从这个意义上说，OSS 不但不是商业软件的敌人，反而可以说是一种必要的和有益的补充。

反观商业软件，由于开源软件的存在，它们失去了原有的垄断利益。这里的垄断不一定是指绝对垄断，还包括经济学意义上的“垄断竞争”。所谓垄断竞争是指在产品中增加不兼容特性，给用户增加替代壁垒，实现一定程度上的事实垄断。

垄断的好处就在于：用户没有别的选择，我做得再不好，卖得再贵，也能混得很好。但是开源软件的出现给了用户更多更好的选择，这就给它们带来了威胁，所以要 FUD 开源软件。

2. 开源贡献

本学期我参加了 **Apache ServiceComb** 的开源项目，囿于能力有限贡献很少。但是在实验室中积极参与了研究生系统的协同开发，学习了 **gitlab** 的使用，对常用的 **git** 命令可以熟练地运用。



3. 课程反馈

课堂上来了很多不同的老师，使这门课涉及到的点很多，但是可能是我个人的经验有限，而且更偏爱边做边学，所以听理论课有时会感觉收获较少。我个人本科时有使用 **github** 的经历，研究生入学以来也参与了实验室使用 **gitlab** 的协同开发，因此对开源项目的理解相对容易一些。但是即便如此，真正参与到一个开源项目之中是需要很多的时间精力的，特别是有些项目的上手不一定简单，甚至有些坑。

我在实验室参加协同开发时，有正在开发的学长指导我，加速了我学习的进度，当然这样的条件在很多开源项目中是几乎不现实的。对于部分同学而言，打开一个 **github** 中的项目，**git clone** 下来之后可能就不知道干什么了，有些项目的引导写的很好，直接按照文档部署项目即可；但还有些项目在部署时会出各种各样的问题，解决这些问题，也许对于一位经验丰富的相关开发者而言是很简单的事情，但是我的水平显然还有待提高。

对于课程的建议，希望能安排至少一次部署、开发、提交的案例分析。个人觉得案例是学习开发技术的最高效方式，直接打开 **ide** 执行相关命令行、看相关代码也是可以的。Talk is cheap. Show me the code.

4. 参考文献

[1] 张海涛.站在潮头看开源:趋势已定,未来已来[J].软件和集成电路,2019(06):12-15.