

## 《开源软件开发技术》2020 年作业 (2)

班级 大数据 1902 学号 2019011191 姓名 郭浩

### 1. 开源开发有哪些人员角色?

维护者  
贡献者  
修订者

### 2. 创建和开展开源项目需要注意的问题有哪些?

清晰的依赖性和安装说明  
至少有一个简要的文档指南  
修改日志和仓库中的标签  
关于支持的语言、运行时、工具版本的信息和项目的成熟度  
一个可以让用户提问和交流的邮件列表

### 3. 开源开发为什么要进行版本控制, 如何进行版本控制?

可以随时回滚到之前的版本  
协同开发时不会覆盖别人的代码  
留下修改记录, 以便随时查看  
发版时可以方便的管理不同的版本

版本控制系统发展至今有几种不同的模式:

#### Local VCS

本地使用 复制/粘贴 的方式进行管理, 缺点是无法协同开发

#### Centralized VCS (Lock, 悲观锁)

中央集中式版本控制系统团队共用仓库, 当某人需要编辑文件时, 进行锁定, 以免其他人同时编辑时造成冲突。缺点是虽然避免了冲突, 但不是很方便。其他人需要排队才能编辑文件, 如果有人编辑了很久或是忘记解锁就会造成其他人长时间等待的情况。

#### Centralized VCS (Merge, 乐观锁)

中央集中式版本控制系统团队共用仓库, 不采用悲观锁方式来避免冲突, 而是事后发现如果别人也修改相同文件(冲突), 再进行手动修改解决。有很多 VCS 属于这种类型, 如: CVS, Subversion, Perforce 等

中央集中式版本控制系统的共同问题是, 做任何操作都需要和服务器同步, 如果服务器宕机则会造成无法继续工作的窘迫。

#### Distributed VCS

分布式版本控制系统, 本地也拥有完整的代码仓库, 就不会出现上述集中式管理的问题, 即使没有网络, 依然可以 commit 和看 log, 也无需担心服务器同步问题。如: Git, Mercurial,

Bazaar 等就属于分布式版本控制系统。缺点是功能比较复杂，上手需要一定的学习时间。

4. 注册 Github 网站，填写个人资料（个人介绍，位置，个人网站等）。在 Github 创建一个仓库（repository），把本作业的文档上传到这个仓库中。将你的 Github 地址写到下面作为答案供检查。

<https://github.com/>\_\_\_\_\_