**1.Git工作流程**

Git的相关空间主要有四个：Workspace：工作区，Index / Stage：暂存区，Repository：仓库区（或本地仓库），Remote：远程仓库，一般步骤如下：

（1）克隆（clone）远程资源库，在本地生成同名仓库。然后可以在克隆的资源上添加或修改文件。

（2）如果远程资源仓库有增加新内容，可以拉取（pull）更新资源到工作区。

（3）提交时先将文件从工作区加到（add）暂存区，然后提交（commit）到本地仓库中，提交后如果需要修改可以通过checkout将文件恢复到工作区，最后从本地仓库推送（push）到远程仓库。

下图是我看到觉得比较好理解的图：



**2.Git分支原理**

Git 保存的不是文件差异或者变化量，而是一系列文件快照。新建一个提交对象前，Git 会计算子目录的校验和，然后在 Git 仓库中将这些目录保存为树对象。之后 Git 创建的提交对象包含着指向这个树对象的指针。

每次提交新更新，当前分支指向最后一个提交更新，而最后一个提交对象则存在一个指针指向前一次的提交更新。

如下图，版本更新时，Git会记录每个版本下，所有目录下文件的校验和，有需要时可以通过指针找到该版本对应的树对象，也就可以恢复到之前版本的内容



而 Git 分支的实现与项目复杂度无关，它可以很快完成分支的创建和切换。同时，因为每次提交时都记录了之前版本信息，要合并分支时，只需要寻找恰当的合并基础，即共同的旧版本，就可以实现分支的合并。

**3.分支的恰当合并**

项目存在两个分支：主分支master，开发分支develop，同时存在三种短期分支，短期分支完成开发后会被合并进develop或master，develop分支上可以进行修改，而合并进master的作为最终版本。可以使用git merge来合并分支。

