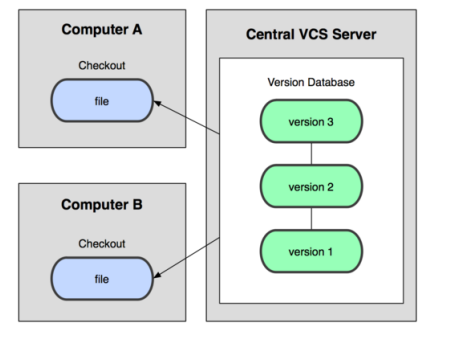
集中化版本控制



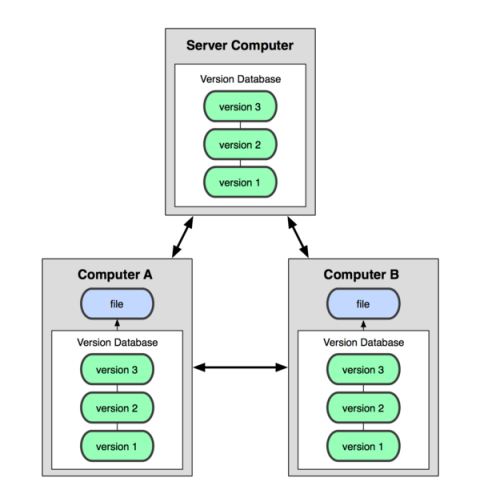
集中式的版本控制系统（Centralized Version Control Systems CVCS），有一个单一的中央管理服务器。

缺点：

中央服务器的单点故障，导致版本数据丢失。

CVS(**Cocurrent Version System**)是一个集中式版本管理系统，大家修改代码后就直接commit到CVS服务器上。跟服务器网络断开之后，不能进行版本查看，管理等等。

分布式版本控制系统（Distributed Version Control System）

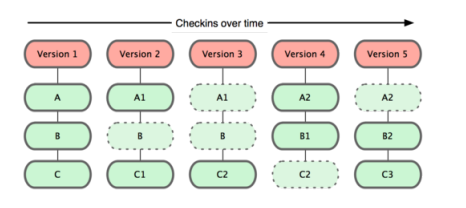
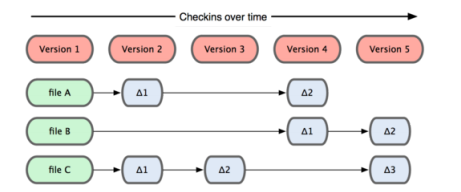


Git是所谓的分布式版本管理系统，并没有集中在某一台服务器上。所以，每个人都可以在本机进行版本控制。然而做项目，总要把大家的成果综合起来，放在一个安全的地方，交给某个人管理。

Git的特点：

1. 直接快照，而非差异

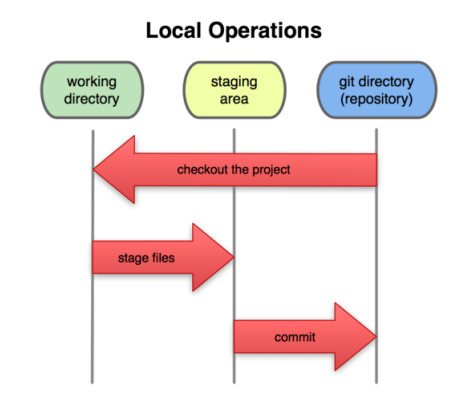
Git 只关心文件数据的整体是否发生变化，而大多数其他系统则只关心文件内容的具体差异



1. 几乎所有的操作都在本地执行
2. 时刻保证数据的完整性

Checksum SHA -1

三种状态：已提交（committed），已修改（modified）和已暂存（staged）



Git 安装

$ apt-get install git-core

Git 配置

用户信息 git config –global user.name/user.email

文本编译器 git config --global core.editor emacs

差异分析工具 git config --global merge.tool vimdiff

查看配置信息 git config – list

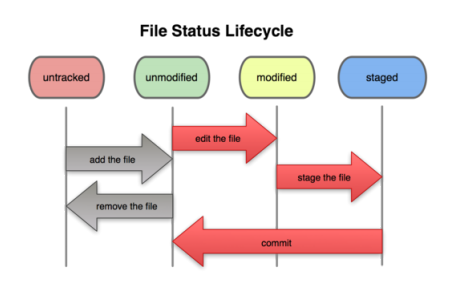
获取帮助 git <verb>--help

Git仓库

git init 初始化一本地仓库

git clone git://git.com/username/test.git 从远程仓库clone。

Git文件状态



状态转换

git add .

git commit –m ‘\*\*\*\*’

.ignore文件，能够实现在提交的时候，自动过滤掉满足条件的文件。

查看已经暂存和未暂存文件的差异

git diff

删除文件

git rm 🡪git commit

查看提交历史

git log

撤销操作

修改上一次的提交(有些文件没有提交或提交信息错误)

git commit –amend

取消已暂存的文件

当add之后，就会将其纳入暂存。撤销已暂存，git reset HEAD file\_name

取消对文件的修改

git checkout -- fine\_name

远程仓库

查看远程仓库

git remote [-v]

git remote show remote-name

添加远程仓库

git remote add myremote git://github.com/user/test.git

拉取远程仓库

git fetch/pull myremote

推送到远程仓库

git push [remote-name] [branch-name]

远程仓库重命名

git remote rename my-remote your-remote\

删除远程仓库

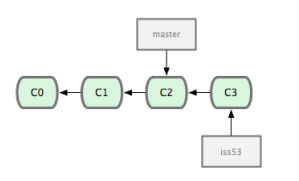
git remote rm your-remote

分支

创建分支 git branch branch-name

切换分支 git checkout branch-name

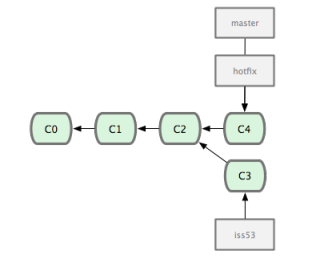
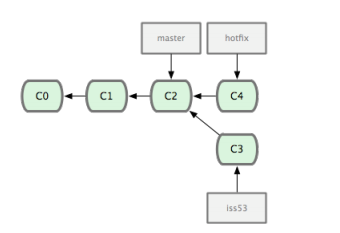
创建并切换 git checkout –b branch-name



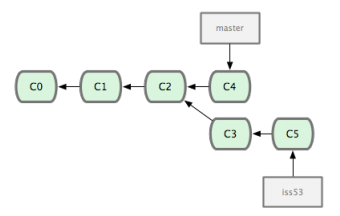
合并分支 将feature分支合并到dev上

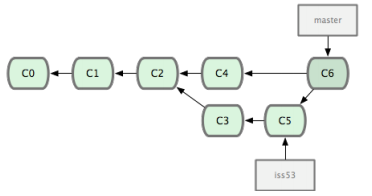
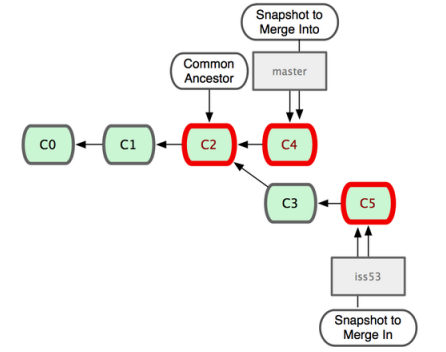
git checkout dev

git merge feature



删除分支：git branch –d branch-name





分支冲突：



查看当前可用分支 git branch [-v]