# Git远程操作详解

Git参考手册：<http://gitref.org/zh/remotes/>

# Git与SVN的区别

1、GIT是分布式的，SVN不是，这是GIT和其它非分布式的版本控制系统，最核心的区别；

GIT跟SVN一样有自己的集中式版本库或服务器。但，GIT更倾向于被使用于分布式模式，也就是每个开发人员从中心版本库/服务器上chect out代码后会在自己的机器上克隆一个自己的版本库。  
2、Git直接记录快照，而非差异比较

Git 和其他版本控制系统的主要差别在于，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。Git 并不保存这些前后变化的差异数据。实际上，Git 更像是把变化的文件作快照后，记录在一个微型的文件系统中。每次提交更新时，它会纵览一遍所有文件的指纹信息并对文件作一快照，然后保存一个指向这次快照 的索引。为提高性能，若文件没有变化，Git 不会再次保存，而只对上次保存的快照作一链接。

3、近乎所有操作都是本地执行

在 Git 中的绝大多数操作都只需要访问本地文件和资源，不用连网。但如果用 CVCS 的话，差不多所有操作都需要连接网络。因为 Git 在本地磁盘上就保存着所有当前项目的历史更新，所以处理起来速度飞快。

# Git常用命令

git clone #工程克隆

git status #查看文件缓存状态

git add #添加至暂存区

git reset #取消缓存修改(还原)

git diff #文件比较

git checkout #放弃文件修改

git commit #提交本地仓库

git log #查看提交日志

远程操作命令:

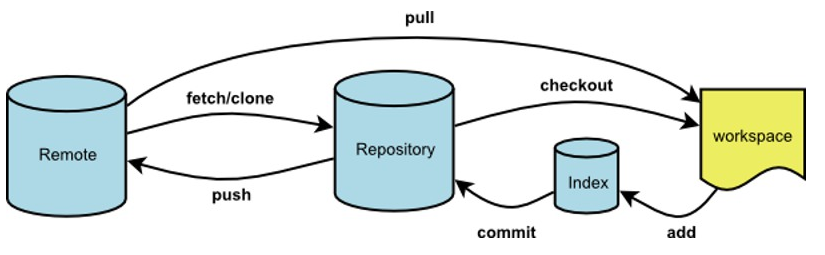
git remote #列出远程别名

git fetch #从远程仓库下载代码

git pull #从远程仓库下载代码并合并 相当于(git fetch + git merge)

git push #向远程仓库推送代码

关联关系图如下：

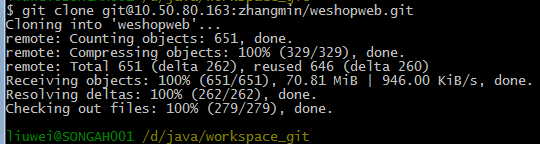


# 基本命令操作流程（以微店weshopweb为例）

## 一、git clone

第一步，通过Git Bash客户端，进入到本地workspace目录，如：

cd /d/java/workspace\_git 然后执行 git clone 命令

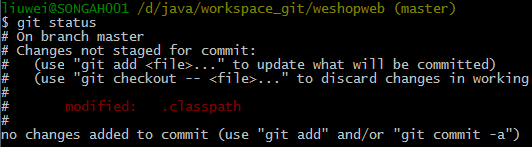




## 二、git status

查看你的文件在工作目录与缓存的状态

比如：修改了.classpath文件，执行git status 如下：

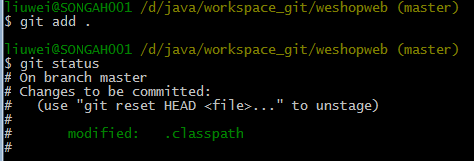


git status –s 以简短的结果输出

## 三、git add

添加文件到缓存

执行git add .后再使用git status命令查看结果如下：

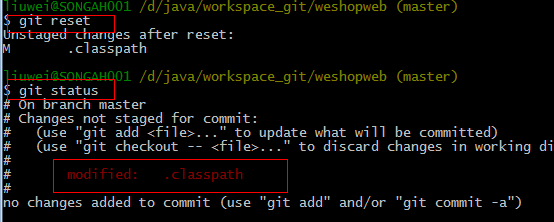


git add . 指添加当前目录下所有修改的文件，添加具体某一个执行git add .classpath

## 四、git reset

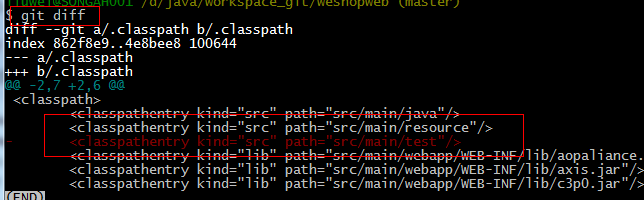
取消已缓存的内容

如果缓存后需要取消，然后重新修改提交，可以执行git reset命令，将文件返回到未缓存的状态



## 五、git diff

查看尚未缓存的改动



显示删除了红色表示的行

#### git diff --cached #查看已缓存的改动

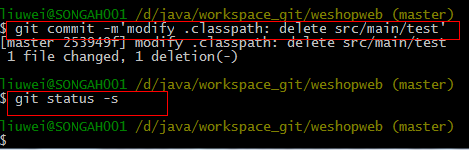
#### git diff HEAD 查看已缓存的与未缓存的所有改动

#### git diff --stat 显示摘要而非整个 diff

## 六、git commit –m’描述’

记录缓存内容的快照

现在你使用 git add 命令将想要快照的内容写入了缓存， 执行 git commit 就将它实际存储快照了。

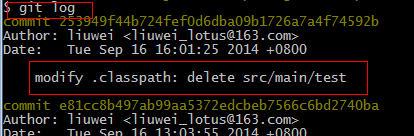


执行完git commit之后用git status查看，发现查询不到修改的文件了

#### git commit –am’描述’  自动将在提交前将已记录、修改的文件放入缓存区

## 七、git log

查看提交日志，按q键退出



git log .classpath #查看某个文件的提交日志

git log --pretty=oneline readme.txt #查看readme.txt文件的历史变动，简易版本

git log --author #只寻找某个特定作者的提交'

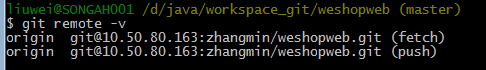
git log --grep 根据提交注释过滤提交记录

git log --oneline --graph 拓扑图形式显示日志

git log --oneline --decorate --graph 可以查看标签

## 七、git remote

列出远程别名



其中origin为默认的远程别名

git remote -v # 列出远程别名

git remote add github git@github.com:schacon/hw.git #添加远程

git remote rm github #删除远程

## 八、git fetch

从远程仓库下载数据及新分支

git fetch origin #更新远程仓库数据到当前分支

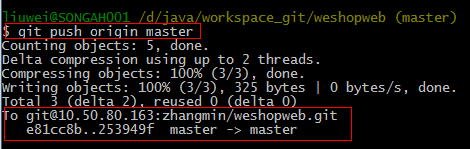
## 九、git pull

从远程仓库提取数据并尝试合并到当前分支

该命令就是在 git fetch 之后紧接着 git merge 远端分支到你所在的任意分支

## 十、git push

推送你的新分支与数据到某个远端仓库



git push -u origin master #推送你的master分支与数据到远端仓库

git push origin master （省略了<dst>，等价于“git push origin master:master”）

git push origin branch1 #推送分支到远程

git push origin :branch1 （再origin repository里面查找branch1，删除它。用一个空的去更新它，就相当于删除了）