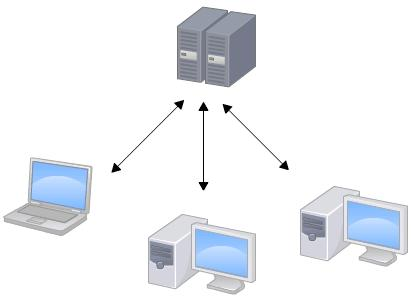
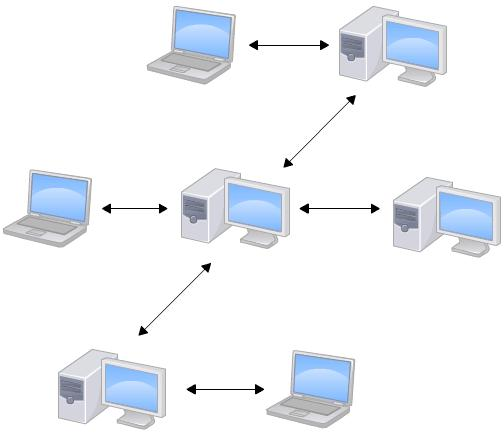
**git与svn**

git 是一款免费的、开源的分布式版本控制系统 svn 集中式版本管理系统



集中式



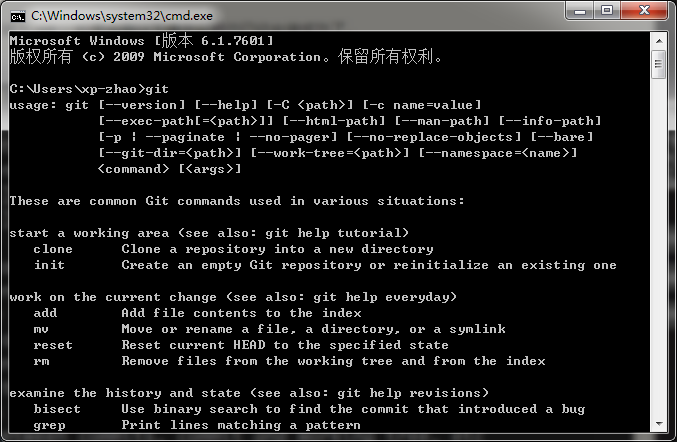
分布式

Git 是分布式的，SVN 是集中式的，好处是跟其他同事不会有太多的冲突，自己写的代码放在自己电脑上，一段时间后再提交、合并，也可以不用联网在本地提交,代码下载下来以后，在本地不需要联网就可以查看所有的 log。

**git安装**

window下载地址：https://git-for-windows.github.io/

验证是否安装成功



设置姓名和邮件地址

git config --global user.name "your name"

git config --global user.email "email@example.com"

tips: --global 参数表示这台机器上所有的 git 仓库都会使用这个配置

**git基础命令**

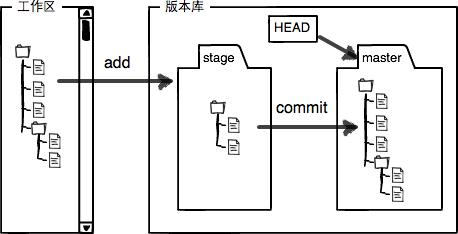
创建版本库

git init 初始化当前目录为 git 仓库

添加文件到版本库

git add <file> 添加文件到暂存区

git commit -m "提交说明"



查看仓库状态

git status (常用命令)

版本切换

git log 查看提交历史

git log -1 查看最后一次提交历史

git reflog 查看所有操作记录，包括删除的 commit 记录

git reset --hard HEAD^ 回退到上一版本 (HEAD 表当前版本， HEAD^ 表上一版本，HEAD^^ 表上两版本，HEAD~99 表上99版本)

git reset --hard commit\_id 切换到指定版本

分支管理

git branch <branch> 创建分支

git checkout <branch> 切换到指定分支

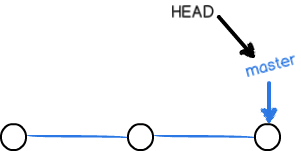
git checkout -b <branch> 创建并切换到该分支

git branch 查看现有分支

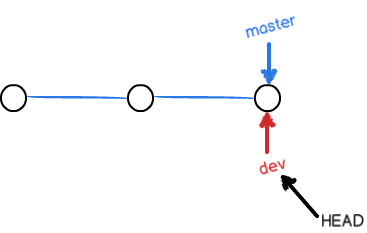
git branch -d <branch> 删除指定分支（-D 强制删除）

**分支详解**

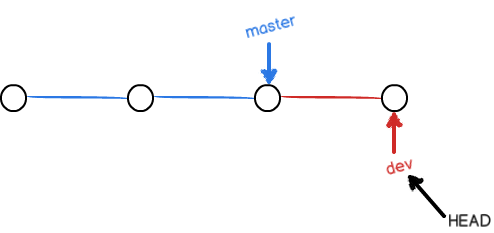
git 中用 master 指向最新的提交，用 HEAD 指向当前分支



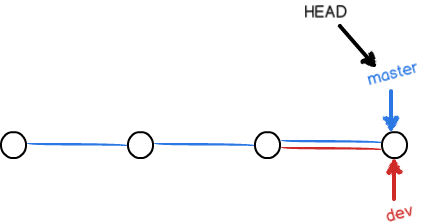
创建分支git checkout -b dev, git 新建一个指针 dev 指向跟 master 相同，并将 HEAD 指向 dev

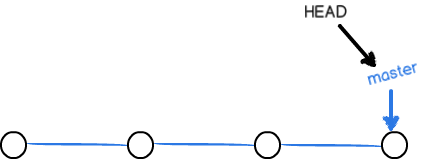


新分支的修改提交，dev 提交后指针向前移动一步，master 指向不变



分支的合并，切换到 master 分支，将 dev 分支合并到 master 分支



删除分支  
 将

**远程仓库**

本地项目添加到远程仓库

1. 在 github 上新建一个与本地仓库同名的空仓库
2. 在本地仓库下运行命令

git remote add origin git@github.com:xp-zhao/gitbook.git

1. 将本地内容推送到 github 远程仓库上(第一次 push 参数带 -u 管理远程仓库) git push origin master

从远程仓库克隆项目

1. 使用命令克隆仓库

git clone git@github.com:xp-zhao/gitbook.git

从远程仓库更新到本地仓库

git pull origin master