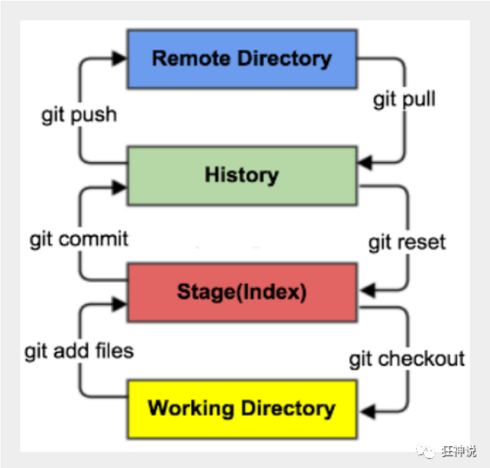
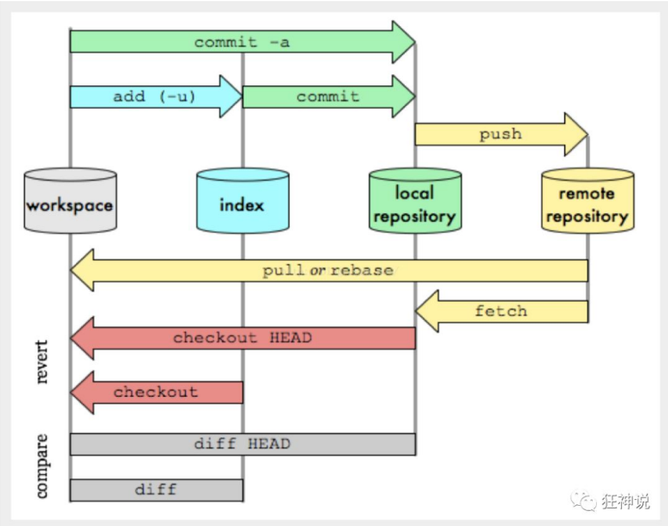
git

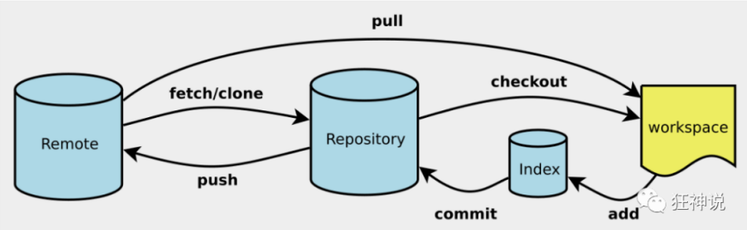
版本控制分类

- 集中式控制　SVN
- 分布式控制　git

git流程



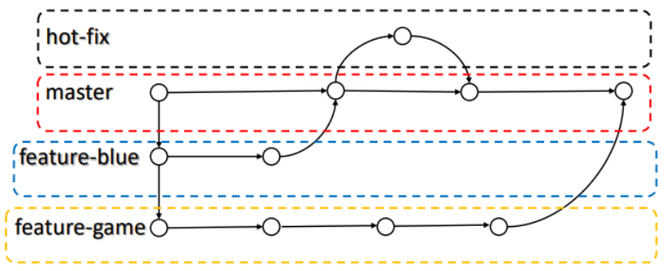




常用命令

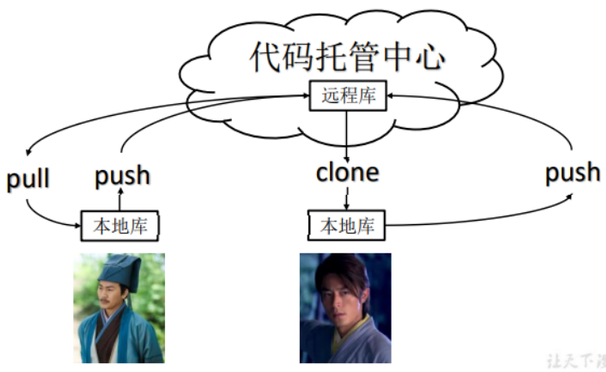
- git init　初始化本地库
- git status　查看本地库状态
- git add <file>
  - 从本地库提交到暂存区
  - 可以仅add一个文件
- git commit -m "version" <file>
  - 从暂存区提交到本地库中
  - -m 后附带版本更新信息
- git log　查看git历史　git reflog　查看简化log
- git reset --hard 版本号(可以为精简版版本号)　Git 切换版本，底层其实是移动的 HEAD 指针，

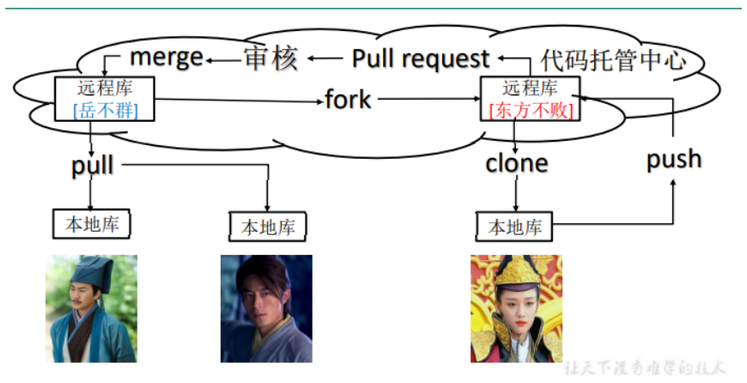
分支



- 底层实际为指针引用
  - 主要为2个指针：
  - head 指向 哪个分支
  - 分支指针指向哪个版本
- git branch -v　查看分支
- git branch 分支名　创建分支
- git checkout 分支名　切换分支　会将head指针指向分支
- git merge 分支名
  - 把指定的分支合并到当前分支上
    - 例如在master下
    - 输入 git merge hot-fix
    - 即将hot-fix内容合并到master下
  - 当合并的2个分支均有修改时，产生合并冲突
    - 需要人工修改代码，确定合并结果
    - 分支名后会多出merging符号
    - 1）编辑有冲突的文件，删除特殊符号，决定要使用的内容
    - 特殊符号：<<<<<<< HEAD 当前分支的代码 ======= 合并过来的代码 >>>>>>> hot-fix
    - 修改完成后需要git add + git commit
      - commit 后不能添加文件名
      - git会不知道commit 哪个分支下的文件

团队协作





- 本地库push到远程库
- 本地库什么都没有时，使用clone复制整份代码
- 本地库有代码时，使用pull从远程库更新本地库中代码
- 远程库间的复制，使用fork
- 远程库间的合并，需要先发送Pull request

Github

- git remote -v　查看当前所有别名
- git remote add 别名 远程地址　为远程地址赋予别名
- git push 别名 分支　推送本地分支上的内容到远程仓库　需要登录账号
- git pull 远程地址库别名 远程分支名　将远程仓库对于分支最新内容拉下来后与当前本地分支直接合并
- git clone 远程地址
  - 将远程仓库内容克隆到本地　无需登录账号
  - 自动化三件事 1、拉取代码 2、初始化本地仓库 3、创建别名
- github上需要加入团队，才能将自己修改的代码push到他人的仓库中
- 团队外协作
  - 搜索时可以在目标库前添加用户名，快速定位
  - 需要fork，将他人远程库代码复制到自己远程库代码中
  - 修改完成后发送pull request，在其中可以就修改内容进行沟通
- 使用SSH 进行免密登录　运行ssh-keygen 生成ssh公钥，然后复制进github settings中

Gitlab

- 局域网远程仓库
- 操作同github

Gitee

IDEA集成Git