



Git原理及使用

讲师：神锋



百家讲坛

About me

- 神锋
 - shenfeng@taobao.com
- Java中间件团队
- 技术大学其他课程
 - Java入门
 - Java深入



我们遇到的问题

- 异地开发的痛苦
- 没有网络的情况下开发？
- checkout过JBoss的源码吗？查看过log吗？
- 集权。。。



痛苦的合并

- 记录合并的版本号
- 向分支bugfix? 大分支合并? 多分支合并?
- 大量“非人为”的冲突
- 增量保存的缺陷



到处都是.svn

- 目录的复制和粘贴



Git的特点

- 分布式
- 本地仓库
- 保存全量内容(snapshot)
 - 极大的降低了合并的难度
- 只有一份.git
 - 不污染子目录
- 给力的命令行



git的对象模型

- 指纹字符串——SHA-1哈希值
 - 长度为40的十六进制字符串
- Blob
- Tree
- Commit



使用前的准备

- 下载 & 安装
- 配置个人信息
 - `$ git config --global user.name "shenfeng"`
 - `$ git config --global user.email "shenfeng@taobao.com"`



创建个人版本库

- `git init`



.git与工作目录

- .git
 - 保存了git仓库的信息
 - 只有一份
- 工作目录
 - 干活的目录



一次提交动作

- 添加或修改文件
- `git status`
 - 查看索引状态
- `git add <file>`
 - 将变更添加到索引
- `git commit`
 - 提交变更



优秀的帮助系统

- 恰到好处的命令提示
- `git help <command>`
 - 详尽的帮助文档



索引的操作

- 索引作为一个暂存区，能使你“变更的”和“需要提交的”分开
 - 甚至支持一个文件只提交一部分
- `git add <file>`
 - 将变更添加到索引
 - 新增的文件
 - 修改的文件
- `git reset HEAD <file>`
 - 将变更从索引中移除，HEAD为版本库“最年轻”的版本



查看差异

- `git diff`
 - 查看工作目录与HEAD的差异
- `git diff --cached`
 - 查看索引与HEAD的差异



启用色彩

- `git config --global color.ui true`
 - 打开全部配色



撤销变更

- `git checkout -- <file>`
 - 将工作目录的文件还原为HEAD
- `git reset HEAD^`
 - 将HEAD版本撤销，HEAD^为当前最新版本的上一个版本



方便的commit

- `git commit -m <msg>`
 - 直接添加注释信息并提交
- 注释技巧
 - 第一行少量概要文字——标题
 - 第二行空行
 - 第三行开始详细描述——正文
- `git commit -a`
 - 自动将所有变更和删除的文件添加到索引并提交，但不包括新增文件



设置忽略选项

- 添加.gitignore文件
 - *.class
 - target/



为指令添加别名

- `git config alias.<alias> <command>`
 - 将<alias>作为命令<command>的别名
- 或者直接修改`.git/config`文件
- 也可以添加`--global`参数使别名全局有效
 - 将修改`~/.gitconfig`文件



善用分支

- 小步快走的开发方式
 - 需要面对两个问题：
 - 开发的临时状态
 - 合并的复杂度
- Git的分支是简单且有效的



一个功能（特性）开发的流程

- git branch
 - 查看分支
- git branch <branchName>
 - 创建分支
- git checkout <branchName>
 - 切换分支
 - git checkout -b <branchName>
 - 创建并切换分支



一个功能（特性）开发的流程2

- 添加、修改文件
- 提交变更
- 切换到合并的目标分支
 - 如master分支
- `git merge <branchName>`
 - 将<branchName>的所有变更合并到当前分支



fast-forward

- 合并分支可能会产生fast-forward情况
- --no-ff参数
 - 强制不使用fast-forward，比如需要保留分支的记录



删除分支

- `git branch -d <branchName>`
 - 删除分支（如果需要）
- 如果合并不完全则会删除失败
- `git diff ... <branchName>`
 - 当前分支与目标分支的交集与目标分支的差异
- 使用-D参数强制删除分支



强大的diff

- 工作目录的文件与索引
- 分支与分支
- 版本与版本
- 其他...
 - `git help diff`



解决冲突

- 冲突的产生
 - 两个合并的分支修改了同一处内容
- 两个分支的维护者共同编辑冲突文件
 - 记得删除冲突标记
- 将冲突文件添加索引并提交



多用户协作



传输协议

- file
 - /path/to/repo.git/
 - file:///path/to/repo.git/
- ssh
 - ssh://[user@]host.xz[:port]/path/to/repo.git/
- http[s]
 - http[s]://host.xz[:port]/path/to/repo.git/
- git
 - git://host.xz[:port]/path/to/repo.git/



获取副本

- `git clone <repo.git>`
 - 可以认为是创建了一个远程分支



多用户协作流程

- 个人开发流程
 - 开分支、完成功能、合并等
- `git pull <remote>`
 - 获取远端更新
- `git push <remote>`
 - 将更新推送到远端，需要写权限



创建中心库

- 可以直接拷贝.git作为中心库



有中心库的开发流程

- 从中心库clone
- 个人开发流程
- 从中心库pull变更
- push变更到中心库





git flow

- A successful Git branching model
 - <http://nvie.com/posts/a-successful-git-branching-model/>



临时仓库

- 正在开发某功能，忽然一个紧急BUG需要修复
- `git stash`
 - 把工作目录的变更压入栈中，并且工作目录还原到HEAD
- `git stash list`
 - 查看栈中有哪些存储
- `git stash pop`
 - 弹出最新压入的变更



查找BUG的利器

- `git blame <file>`
 - 查看指定文件每一行的最后版本及修改者
- `git bisect`
 - 帮助你快速定位到有问题的版本



一些学习资源

- Git Community Book 中文版
 - <http://gitbook.liuhui998.com/>
- Pro Git
 - <http://progit.org/book/zh/>
- Why Git is Better than X
 - <http://zh-cn.whygitisbetterthanx.com/>



Q & A



百家讲坛

希望你带来了帮助
Bye~

