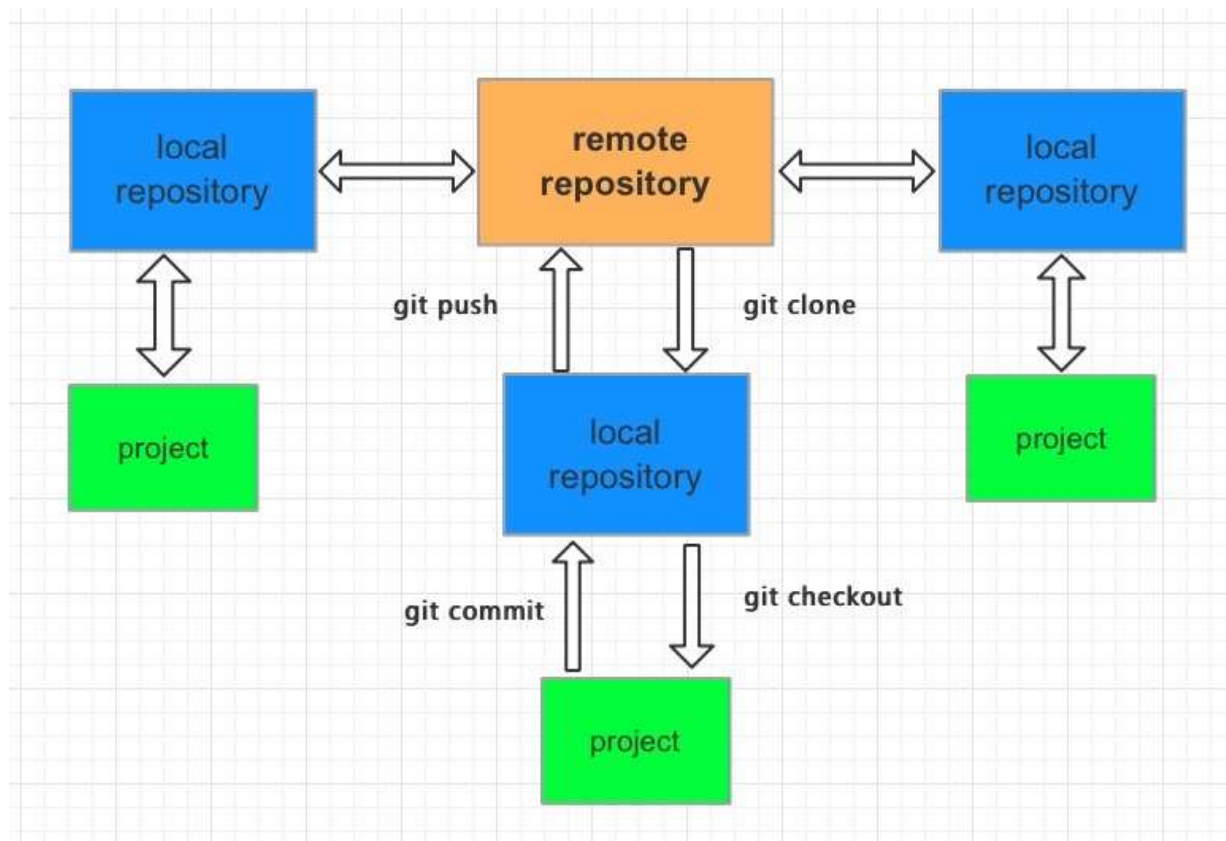


1、git简介

git是目前流行的分布式版本管理系统。它拥有两套版本库，本地库和远程库，在不进行合并和删除之类的操作时这两套版本库互不影响。也因此其近乎所有的操作都是本地执行，所以在断网的情况下任然可以提交代码，切换分支。git又使用了SHA-1哈希算法确保了在文件传输时变得不完整、磁盘损坏导致数据丢失时能立即察觉到。

git的基本工作流程：



- git clone: 将远程的Master分支代码克隆到本地仓库
- git checkout: 切出分支出来开发
- git add: 将文件加入库跟踪区
- git commit: 将库跟踪区改变的代码提交到本地代码库中
- git push: 将本地仓库中的代码提交到远程仓库

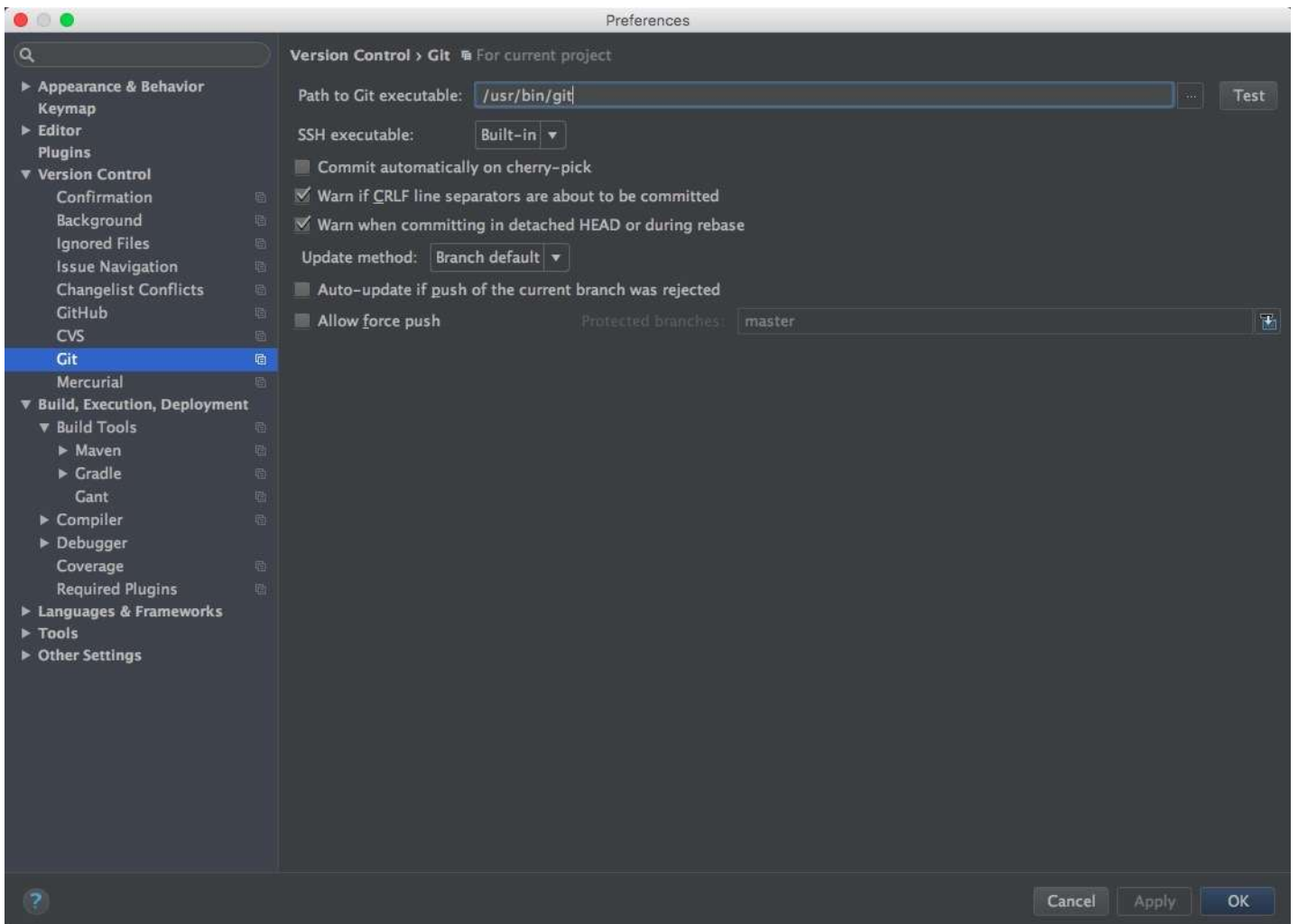
git 分支

- 主分支
 - master分支: 存放随时可供生产环境中的部署的代码
 - develop分支: 存放当前最新开发成果的分支，当代码足够稳定时可以合并到master分支上去。
- 辅助分支
 - feature分支: 开发新功能使用，最终合并到develop分支或抛弃掉
 - release分支: 做小的缺陷修正、准备发布版本所需的各项说明信息
 - hotfix分支: 代码的紧急修复工作

2、git在IntelliJ IDEA下的使用

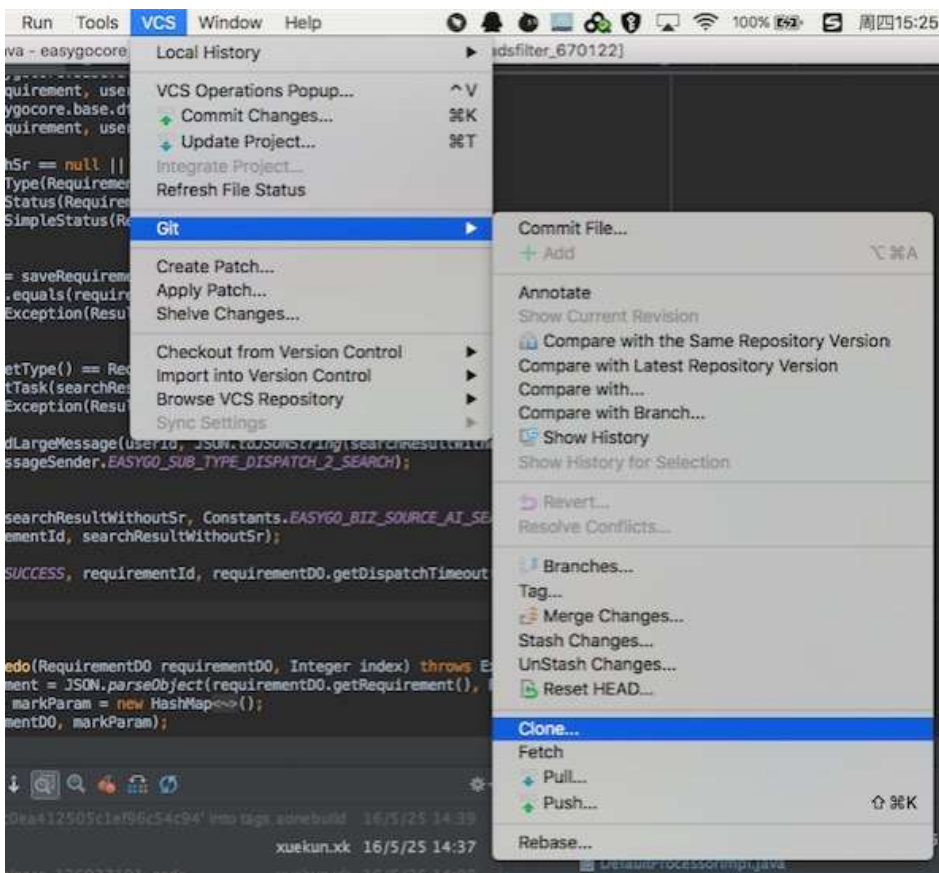
2.1、IntelliJ IDEA下配置git

- 本地安装好git，并配置合理的SSH key，[具体看这里](#)
- IntelliJ IDEA->Performance->Version Control->git 将自己安装git的可执行文件路径填入Path to Git executable，点击 Test测试一下

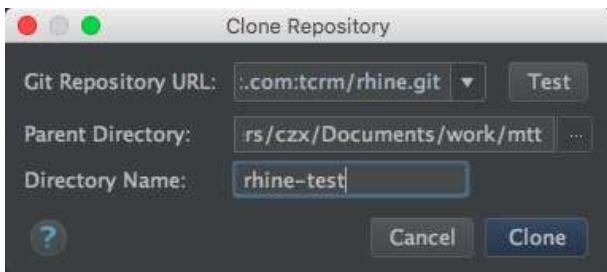


2.2、git clone

- VCS->Git->Clone

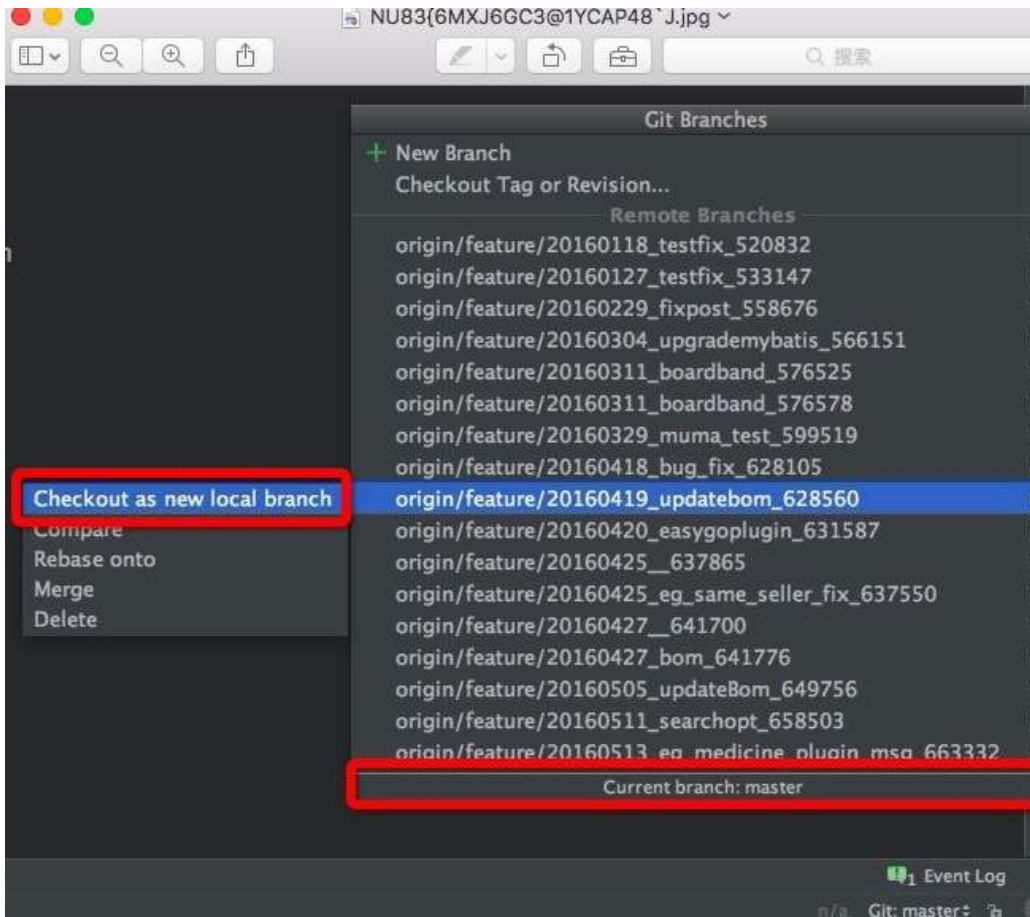


- 输入你的远程仓库地址,点击测试一下地址是否正确

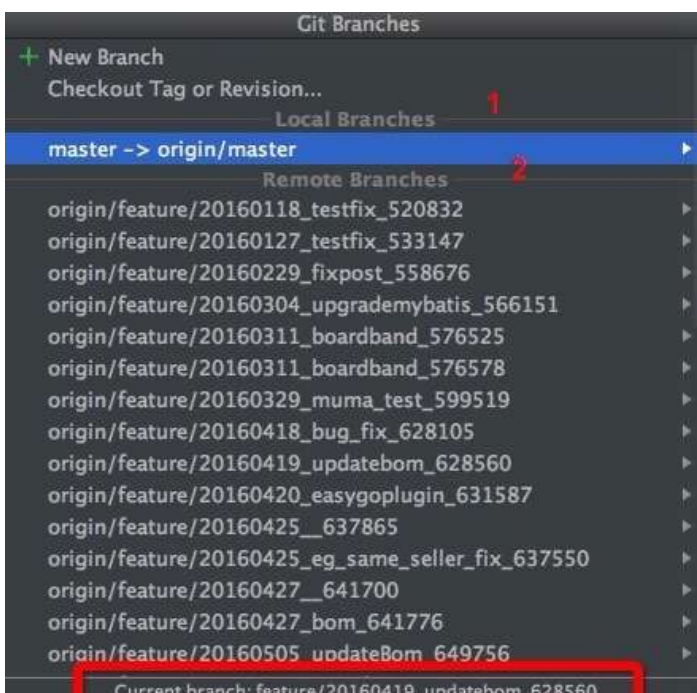


2.3、git checkout

- 在IntelliJ IDEA右下角有一个git的分支管理，点击。选择自己需要的分支，checkout出来



- checkout出来，会在底端显示当前的分支。其中1显示的为本地仓库中的版本，2为远程仓库中的版本

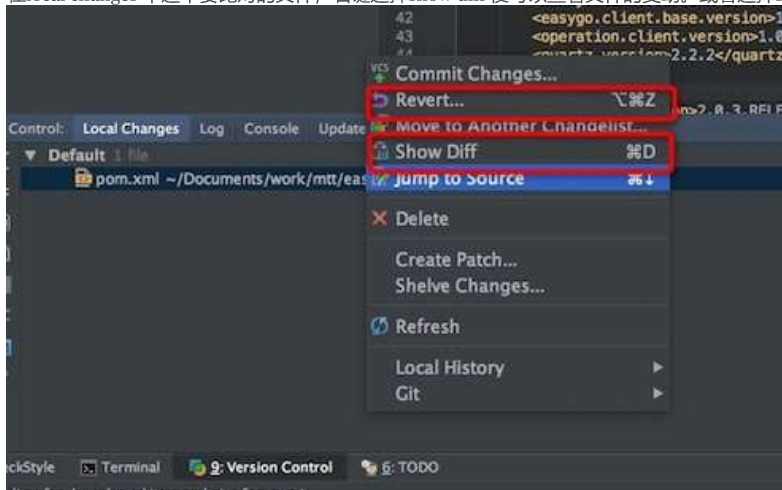


- 点击IDE的右上角的向下箭头的VCS，将分支的变更同步到本地



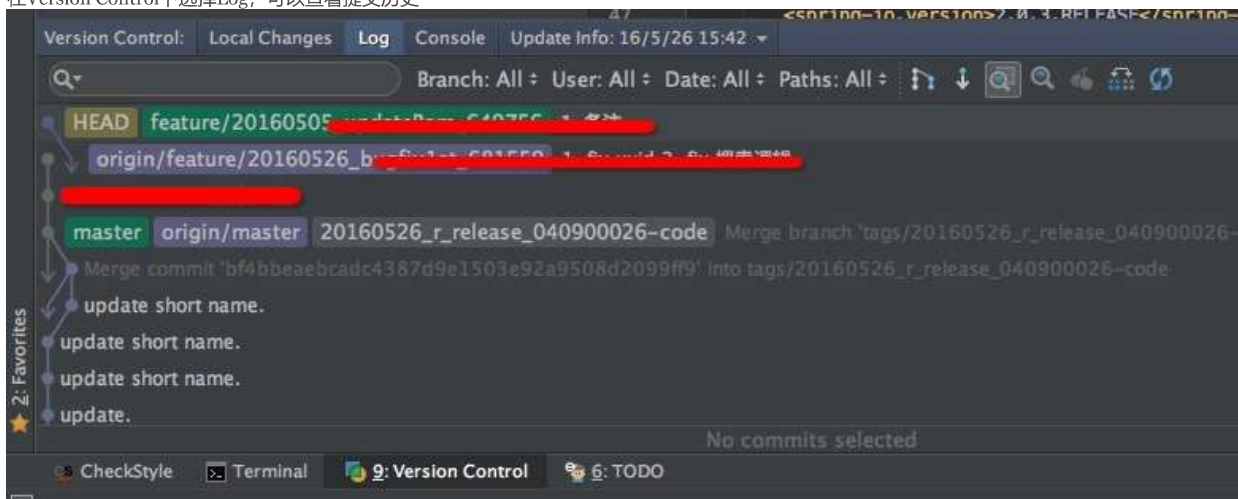
2.4、git diff

- 在local changes 中选中要比对的文件，右键选择show diff 便可以查看文件的变动。或者选择Revert放弃文件的改动



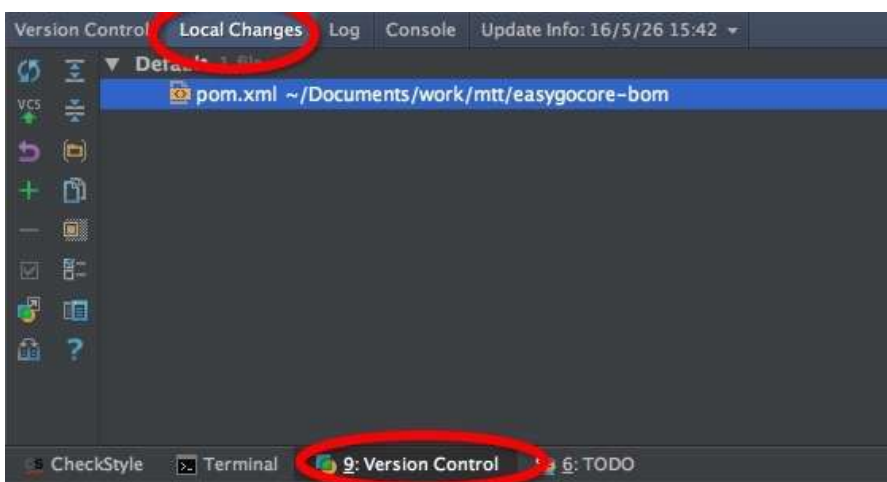
2.5、git log

- 在Version Control下选择Log，可以查看提交历史

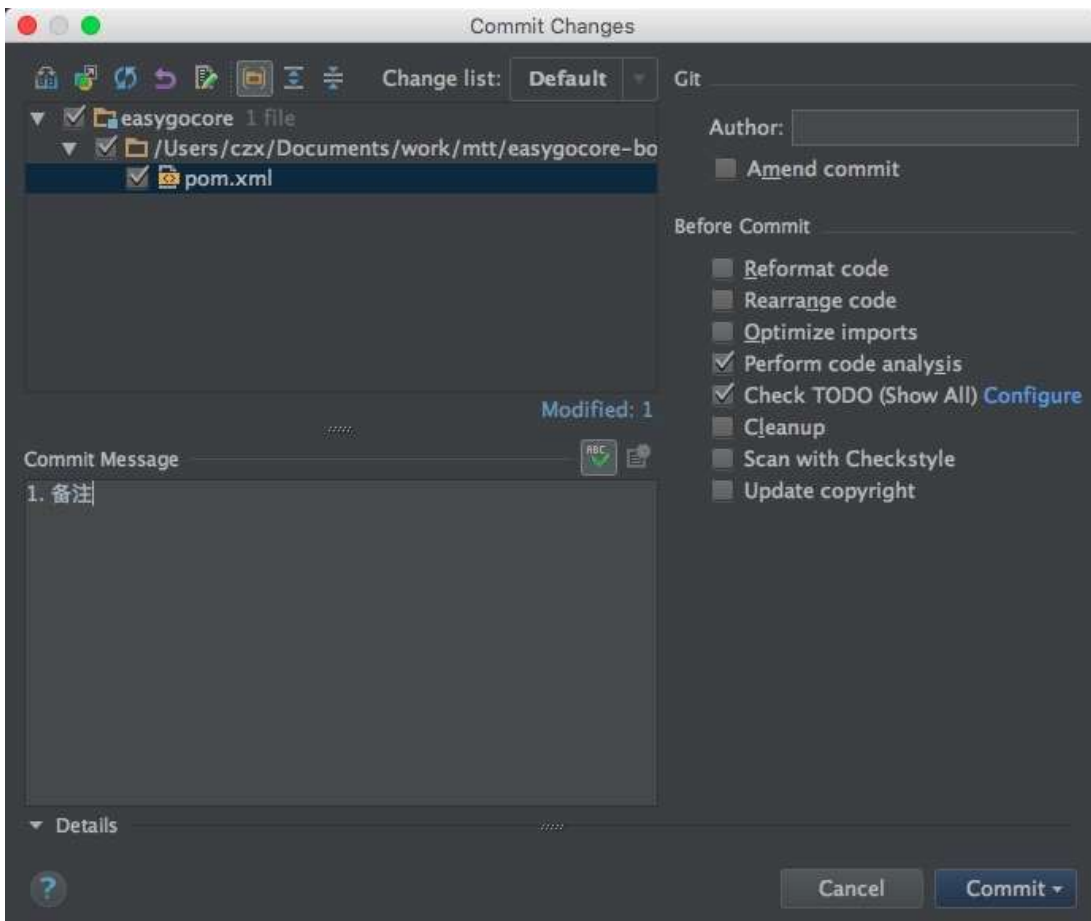


2.6、git commit

- 默认导入的工程已经git add加入库跟踪区了
- 随便修改一下pom.xml文件，其修改的文件会显示在Version Control中的local changes下

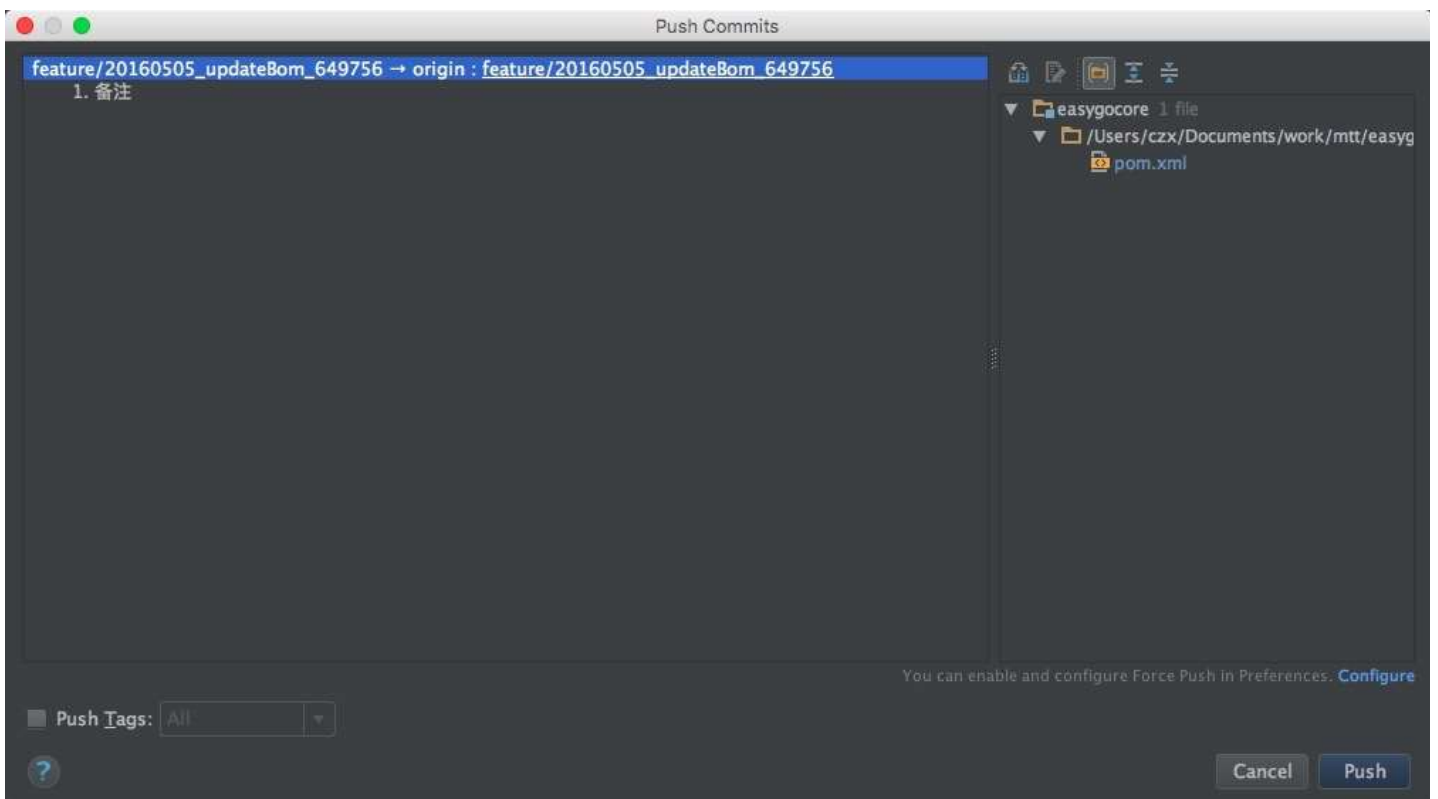


- 点击IDE右上角的向上箭头的VCS， git commit, 写上日志提交到本地代码库中



2.7、git push

- VCS->Git->Push 将本地代码提交到远程仓库



2.8、在Idea命令行使用git

mac下同时按alt+F12,进入idea命令行

常见的命令:

clone项目 git clone xxxxxx

检查项目状态 git status

切换分支并和远程的分支关联 git checkout -b xxx -t origin/xxx

拉最新更新 git pull

提交更新 git commit -am "备注"

合并分支到当前分支，首先切换到需要被合并的分支 git checkout xxx, 再合并 git merge yyyy

提交 git push