

SVN和Git差异

分享人：朱云华

专业：.net工程师，Android工程师

QQ：1481447401

邮箱：1481447401@qq.com

微博：未城未来

微信：朱云华

分享项目

- Git和SVN简介
- Git和SVN特点及其区别
- 更新提交及撤消操作
- 常用操作实践
- Git和SVN协作
- 权限管理

Git和SVN简介

SVN（Subversion），包括服务器和客户端两种，其中SVN服务器有2中运行方式：独立服务器和借助apache运行。本次主要讲SVN客户端的操作和Git操作的差异性。

具有直观的Gui界面操作，方便操作，容易理解，通常在Windows桌面的操作系统上使用，并且具有VS和Eclipse插件。

http://subclipse.tigris.org/update_1.0.x

Git和SVN简介

Git开源的分布式版本控制系统。通常使用命令行的操作，使用在Linux操作系统上。

也有Git的图形操作工具，Eclipse Egit项目提供了Git与Eclipse的集成。

<http://www.eclipse.org/egit/>

Git和SVN特点及其区别

1. 集中式和分布式

SVN数据集中式版本控制系统，中央版本库为一个唯一的URL，有服务器保存所有文件的修订版本机器操作记录，其他人都连在同一个服务器上操作，这样管理人员可以控制所有人及其所有文件夹的权限。服务器出了问题，就无法协同工作了

Git数据分布式控制系统，客户端不只是提取最新版本的文件快照，而是把原始的代码仓库完整地镜像下来。任何一处协同工作的服务器发生故障，都可以用任何一个镜像出来的本地仓库恢复。就可以在同一个项目中，分别和不同工作小组的人相互协作。并且每一次完全在本地完成的提交都最一次备份。同时**Git**也可以模拟集中式的工作模式，版本库在统一的服务器中，可以对版本库进行授权。

2. 版本库和工作区分离与一体

Svn的版本库是在服务器上，在工作区上进行操作，只有在**Commit**或者**Update**时，才版本库交互。其中工作区中的每一个文件夹都包含一个.svn文件，记录这这个文件夹中的文件更新记录。在使用**update**是进行更新内容。

Git的工作区和版本库是一体的，在根目录中的.git目录就是版本库本身。提供了版本库的操作历史。

更新和提交撤消操作

更新操作包括增、删、改三种操作，对于SVN来说，这三种操作都是在本地做修改，对SVN来讲是暂存在工作区中，而Git是产生一个索引文件，只有使用了Commit命令之后才做对服务器中版本库文件的修改，对于Commit之前的操作，都是以最后一次修改为准。

由于更新操作通常是需要互斥的操作，在使用操作的过程中，有必要对要修改的文件进行加锁操作。

增删的操作的Git命令：

Git add filename, rm filename

修改并加入注释：git add filename && git commit -m "a new file has been added"

增删的SVN操作：

在工作区中添加文件或者文件夹，并使用TortoiseSVN的Add菜单，将文件在工作区中添加，在使用Commit。

SVN操作对应的状态：normal,modified,conflicted,readonly,deleted,locked,added,ignored,non-versioned。

Git对应的操作状态：changed / new / deleted/ modified（Commit和非Commit两种状态下）

更新和提交撤消操作

回滚功能是比较常用的，当改错时，可以及时的回到以前操作的状态，Git的回滚和SVN的回滚又是不一样的。

回滚到某一次的修改：

SVN： 使用update to reversion 菜单,选择你需要revert到的ID； 或者使用show log菜单， 选择update item to reversion或者revert changes to this version， 将相应的版本代码下载下来， 然后在使用commit菜单提交到版本库。

Git 使用的命令有两种：

Git revert 是撤消某次提交，并保留之前的commit。

Git reset 是撤消某次提交，此次之后的修改都会退回到暂存区。

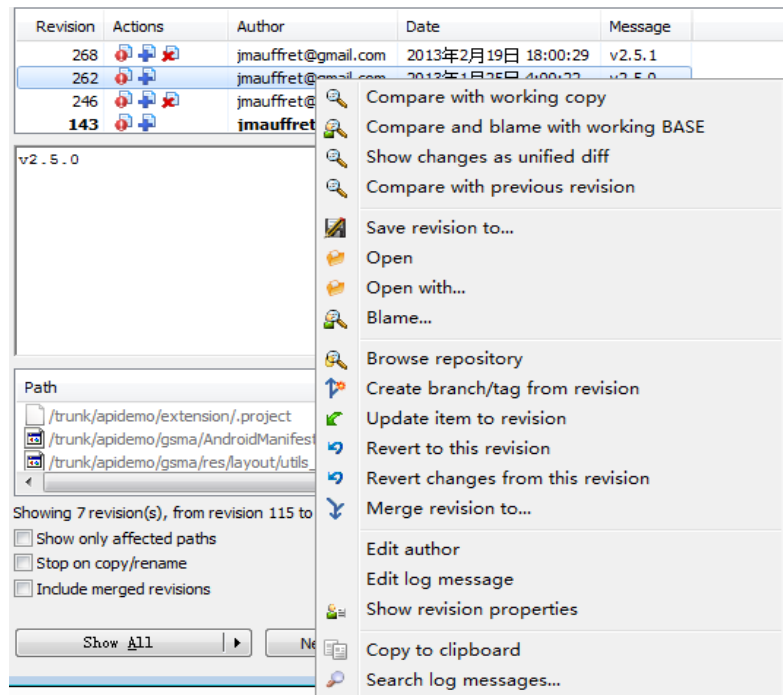
常用操作实践

比较项目	Git	SVN
添加文件	Git add filename && git commit -m "comment"	在文件夹中直接创建，使用add菜单添加，并用Commit提交
比较	Get diff SHA11..SHA12	Differ 或者compare菜单
提交	Commit - m "comment"	Commit菜单，并且要选择每一个要提交的项
状态	Git status	文件在文件夹中的颜色及图标显示
日志	Git log	Show log菜单，在显示的Log中能够针对历史Log做各种操作
修改提交的注释	Git commit -amend - m "aaaa"	在show log中，双击某个版本对应的修改记录，进行注释的重新填写及修改。
更新	Git clone	Check out 菜单/ SVN update 菜单

常用操作实践

SVN的操作多时图形化的，直接方便操作，一看就懂。

Git的操作是命令语句，显示针对文件历史版本的操作没有SVN那么简单
SVN不但能够根据菜单简单方便的操作，还能够使用客户端的属性来配置必要的操作条件。



Git和SVN协作

Svn的特点是中心化的管理，较少使用分支。这使得本地与服务器的连接非常重要。Git弥补了Svn的中心化的特点。Git-svn工具集合了Svn和Git优点，即能够进行本地代码备份和中心源码库的版本管理。

在Git包中有git-svn，如果环境没有的话，需要使用`sudo apt-get install git-svn` 进行安装，并把`/usr/lib/git-core`配置到Path环境变量中。使用方法上是将 `git`命令 操作为`git-svn`命令。

如从服务器的svn repository获取最新的更新：`git-svn rebase`。

检出操作：`git-svn colon svn_repository_url`

更多操作：<https://www.kernel.org/pub/software/scm/git/docs/git-svn.html>

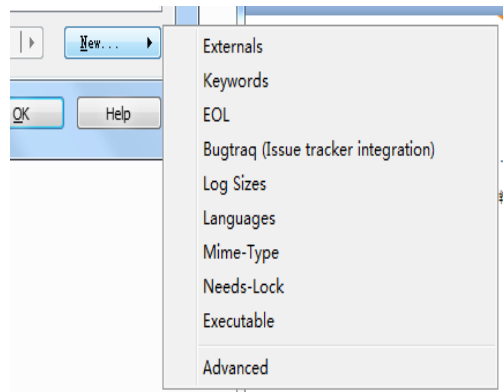
权限管理

SVN的权限比较详细，能够具体到授权用户对文件夹的操作权限，不能够对单个文件授权。

SVN的授权不仅仅在对用户操作的授权，同时可以在用户提交文件的注释中作出限制，比如输入的注释最大最小长度，及其注释中必须包含某些关键字。

Git的授权是非零即一式授权。

SVN注释权限配置方法可以在SVN服务器端的pre-commit.tmpl文件中，及其的SVN客户端中properties中设置。



SVN的注释权限设置