

CSDN新首页上线啦，邀请你来立即体验！(http://blog.csdn.net/)

立即体验

登录 (https://passport.csdn.net/account/login?ref=toolbar) 注册 (http://passport.csdn.net/account/mobileregister?ref=toolbar&action=mobileRegister)

博客 (http://blog.csdn.net/?ref=toolbar) 学院 (http://edu.csdn.net/?ref=toolbar)

下载 (http://download.csdn.net/?ref=toolbar) 更多

搜索

写博客 (http://write.blog.csdn.net/console/activities)

我的文章 (http://my.blog.csdn.net/console/activities)

# [Git高级教程(二)] 远程仓库版本回退方法

原创

2016年08月15日 11:08:30

标签：git (http://so.csdn.net/so/search/s.do?q=git&t=blog)

30760

## 1 简介

最近在使用git时遇到了远程分支需要版本回滚的情况，于是做了一下研究，写下这篇博客。

## 2 问题

- 如果提交了一个错误的版本，怎么回退版本？
- 如果提交了一个错误的版本到远程分支，怎么回退远程分支版本？
- 如果提交了一个错误的版本到公共远程分支，又该怎么回退版本？

## 3 本地分支版本回退的方法

如果你在本地做了错误提交，那么回退版本的方法很简单  
先用下面命令找到要回退的版本的commit id：

```
1 git reflog
```

接着回退版本：

```
1 git reset --hard 0bfafd
```

0bfafd就是你要回退的版本的commit id的前面几位

## 4 自己的远程分支版本回退的方法

如果你的错误提交已经推送到自己的远程分支了，那么就需要回滚远程分支了。  
首先要回退本地分支：

```
1 git reflog
2 git reset --hard 0bfafd
```

紧接着强制推送到远程分支：

```
1 git push -f
```



**fuchaosz** (http://blog.c...)

+ 关注

(http://blog.csdn.net/fuchaosz)

码云

0

(https://g

utm\_sour

原创

33

粉丝

170

喜欢

0

他的最新文章

更多文章 (http://blog.csdn.net/fuchaosz)

Android 神器 xposed 框架使用指南  
(http://blog.csdn.net/fuchaosz/articl  
e/details/53143216)

[Android编译(二)] 从谷歌官网下载an  
droid 6.0源码、编译并刷入nexus 6p  
手机 (http://blog.csdn.net/fuchaosz/  
article/details/52473660)

[技术随笔 (二)] win10 回滚 win7 注  
意事项 (http://blog.csdn.net/fuchaos  
z/article/details/52241223)



### 在线课程



数据科学家  
从入门到精通

数据科学家从入门到精通 course/detail/611?utm\_source=blog9)

(http://edu.csdn.net/hu  
iyiCourse/detail/611?utm\_source=blog9)



SDCC 2017  
前端技术实战线上峰会

(明日开播)SDCC 2017 course/series\_detail/之前端技术实战线上峰会 (http://edu.csdn.net/hu  
iyiCourse/series\_detail/72?utm\_source=blog9)

### 热门文章

本地分支回滚后，版本将落后远程分支，必须使用强制推送覆盖远程分支，否则无法推送到远程分支

## 5 公共远程分支版本回退的问题

看到这里，相信你已经能够回滚远程分支的版本了，那么你也许会问了，回滚公共远程分支和回滚自己的远程分支有区别吗？

答案是，当然有区别啦。

一个显而易见的问题：如果你回退公共远程分支，把别人的提交给丢掉了怎么办？

下面来分析:

假如你的远程master分支情况是这样的:

A1-A2-B1

其中A、B分别代表两个人，A1、A2、B1代表各自的提交。并且所有人的本地分支都已经更新到最新版本，和远程分支一致。

这个时候你发现A2这次提交有错误，你用reset回滚远程分支master到A1，那么理想状态是你的队友一拉代码git pull，他们的master分支也回滚了，然而现实却是，你的队友会看到下面的提示：

```
1 $ git status
2 On branch master
3 Your branch is ahead of 'origin/master' by 2 commits.
4   (use "git push" to publish your local commits)
5 nothing to commit, working directory clean
```



也就是说，你的队友的分支并没有主动回退，而是比远程分支超前了两次提交，因为远程分支回退了嘛。

(1) 这个时候，你大吼一声：兄弟们，老子回退版本了。如果你的队友都是神之队友，比如：Tony(腾讯CTO)，那么Tony会冷静的使用下面的命令来找出你回退版本后覆盖掉的他的提交，也就是B1那次提交：

```
1  git reflog
```

然后冷静的把自己的分支回退到那次提交，并且拉个分支：

```
1 git checkout tony_branch //先回到自己的分支
2 git reflog //接着看看当前的commit id,例如:0bbbbb
3 git reset --hard B1 //回到被覆盖的那次提交B1
4 git checkout -b tony_backup //拉个分支,用于保存之前因为回退版本被覆盖掉的提交B1
5 git checkout tony_branch //拉完分支,迅速回到自己分支
6 git reset --hard 0bbbbb //马上回到自己分支的最前端
```

通过上面一通敲，Tony暂时舒了一口气，还好，B1那次提交找回来了，这时tony\_backup分支最新的一次提交就是B1，接着Tony要把自己的本地master分支和远程master分支保持一致：

```
1 git reset --hard origin/master
```

[Android Studio系列(五)] Android Studio手动配置Gradle的方法 (<http://blog.csdn.net/fuchaosz/article/details/51567808>)

 42330

Ubuntu 16.04 安装QQ解决方案 (<http://blog.csdn.net/fuchaosz/article/details/51919607>)

40641

[Android 编译(一)] Ubuntu 16.04 LTS  
成功编译 Android 6.0 源码教程 (<http://blog.csdn.net/fuchaosz/article/details/51487585>)

33502

[Git高级教程(二)] 远程仓库版本回退方法  
(<http://blog.csdn.net/fuchaosz/article/details/52170105>)

 30740

Ubuntu 16.04 系列软件安装命令合集  
包括QQ、搜狗、迅雷、网易云音乐等  
 30548

执行上面这条命令后，Tony的master分支才真正的回滚了,也就是说你的回滚操作才能对Tony生效，这个时候Tony的本地maser是这样的：

A1

接着Tony要再次合并那个被丢掉的B1提交：

```
1 | git checkout master           //切换到master
2 | git merge tony_backup         //再合并一次带有B1的分支到master
```

好了，Tony终于长舒一口气，这个时候他的master分支是下面这样的：

A1 – B1

终于把丢掉的B1给找回来了，接着他push一下，你一拉也能同步。

同理对于所有队友也要这样做，但是如果该队友没有提交被你丢掉，那么他拉完代码git pull之后，只需要强制用远程master覆盖掉本地master就可以了：

```
1 | git reset --hard origin/master
```

(2) 然而很不幸的是，现实中，我们经常遇到的都是猪一样的队友，他们一看到下面提示：

```
1 | $ git status
2 | On branch master
3 | Your branch is ahead of 'origin/master' by 2 commits.
4 |   (use "git push" to publish your local commits)
5 | nothing to commit, working directory clean
```

就习惯性的git push一下，或者他们直接用的SourceTree这样的图形界面工具，一看到界面上显示的是推送的提示就直接点了推送按钮，卧槽，你辛辛苦苦回滚的版本就这样轻松的被你猪一样的队友给还原了，所以，只要有一个队友push之后，远程master又变成了：

A1 – A2 – B1

这就是分布式，每个人都有副本。这个时候你连接他的心都有了，怎么办呢？你不能指望每个人队友都是git高手，下面我们另外用另外一种方法来回退版本。

**注意：博主是在虚拟机中实验的，用于模拟两个人的操作，如果你在一个机器上，用同一个账号在不同的目录下克隆两份代码来实验的话，回退远程分支后，另外一个人是不会看到落后远程分支两次提交的，所以请务必使用虚拟机来模拟A、B两个人的操作**

## 6 公共远程分支版本回退的方法

使用git reset回退公共远程分支的版本后，需要其他所有人手动用远程master分支覆盖本地master分支，显然，这不是优雅的回退方法，下面我们使用另一个命令来回退版本：

```
1 git revert HEAD //撤销最近一次提交
2 git revert HEAD~1 //撤销上上次的提交，注意：数字从0开始
git revert 0ffaacc //撤销0ffaacc这次提交
```

git revert 命令意思是撤销某次提交。它会产生一个新的提交，虽然代码回退了，但是版本依然是向前的，所以，当你用revert回退之后，所有人pull之后，他们的代码也自动的回退了。

但是，要注意以下几点：

- 1. revert 是撤销一次提交，所以后面的commit id是你需要回滚到的版本的前一次提交
- 2. 使用revert HEAD是撤销最近的一次提交，如果你最近一次提交是用revert命令产生的，那么你再执行一次，就相当于撤销了上次的撤销操作，换句话说，你连续执行两次revert HEAD命令，就跟没执行是一样的
- 3. 使用revert HEAD~1 表示撤销最近2次提交，这个数字是从0开始的，如果你之前撤销过产生了commi id，那么也会计算在内的。
- 4. 如果使用 revert 撤销的不是最近一次提交，那么一定会有代码冲突，需要你合并代码，合并代码只需要把当前的代码全部去掉，保留之前版本的代码就可以了。

git revert 命令的好处就是不会丢掉别人的提交，即使你撤销后覆盖了别人的提交，他更新代码后，可以在本地用 reset 向前回滚，找到自己的代码，然后拉一下分支，再回来合并上去就可以找回被你覆盖的提交了。

## 7 revert 合并代码，解决冲突

使用revert命令，如果不是撤销的最近一次提交，那么一定会有冲突，如下所示：

```
1 <<<<<< HEAD
2 全部清空
3 第一次提交
4
5 全部清空
6 >>>>>> parent of c24cde7... 全部清空
```

解决冲突很简单，因为我们只想回到某次提交，因此需要把当前最新的代码去掉即可，也就是HEAD标记的代码：

```
1 <<<<<< HEAD
2 全部清空
3 第一次提交
4 =====
```

把上面部分代码去掉就可以了，然后再提交一次代码就可以解决冲突了。

## 8 继续扩展，简单粗暴的回滚方法

看到这里也许你已经觉得学会了远程仓库版本回滚方法了，但是实践中总是会遇到很多不按套路来的问题，考虑下面一种情况：

如果你们开发中，忽然发现前面很远的地方有一次错误的合并代码，把本来下一次才能发的功能的代码合并到了这一次来了，这个时候全体成员都觉得直接回滚比较快，因为他们都有备份，覆盖了无所谓，这个时候用reset的话对队友的要求比较高，用revert的话呢要大面积的解决冲突，也很麻烦呀，怎么办呢？

⚠

内容举报

⬆TOP

返回顶部

这个时候，可以使用简单粗暴的办法，直接从那个错误的提交的前一次拉取一份代码放到其他目录，然后将master代码全部删除，把那份新代码方进去，然后提交，果然简单粗暴啊，虽然这种方法不入流，但是，实践中发现很好使啊，所以，实践是检验真理的唯一标准。遇到问题还是要灵活应对。

如果你遇到问题，欢迎给我留言，我CSDN博客——“梧桐那时雨”。

## 9 总结

远程分支回滚的三种方法：

1. 自己的分支回滚直接用reset

2. 公共分支回滚用revert

3. 错的太远了直接将代码全部删掉，用正确代码替代

如果你遇到问题，欢迎给我留言，我CSDN博客——“梧桐那时雨”。

## 10 转载请注明来自“ 梧桐那时雨” 的博客：<http://blog.csdn.net/fuchaosz/article/details/52170105> (<http://blog.csdn.net/fuchaosz/article/details/52170105>)

Tips :

如果觉得这篇博客对你有帮助或者喜欢博主的写作风格，就关注一下博主或者给博主留个言呗，鼓励博主创作出更多优质博客，Thank you.

版权声明：本文为博主原创文章，未经博主允许不得转载。



16



- 

sinat\_29774479 (/sinat\_29774479) 3天前 10:13

(/sinat\_29774479) 失风趣

回复

15楼
- 

ggg (/ggg) 2017-11-08 21:57

(/ggg) 继续扩展，简单粗暴的回滚方法  
常用这个方法，很好使  
文章总结得很好，点个赞~

回复

14楼
- 

u013047584 (/u013047584) 2017-08-13 11:02

(/u013047584) 啊，博主超级给力！！请问博主您的博客是用MarkDown写的吗？

回复

13楼

查看 18 条热评

### 相关文章推荐