

Git使用之——冲突解决一 (git merge conflict)

置顶 2013年11月02日 00:10:17 [young-hz](#) 阅读数：90433 [更多](#)

Git冲突是在多用户协同工作下出现，在一些情况下Git可以智能自动合并，但有时需要用户手动合并。

(本文链接：<http://blog.csdn.net/u012150179/article/details/14047183>)

一 . Git智能自动合并

1. 多成员修改不同文件

这和之前所述的“非快进式推送”一致，按照其方式解决 (<http://blog.csdn.net/u012150179/article/details/36044515>) 。

2. 多成员修改相同文件不同区域

3. 同时修改文件名和文件内容

解决：

上述三种方式都可以用同一中方式解决，也就是在执行push之前先进行pull操作，即

```
git pull
```

这里

```
git pull = git fetch + git merge refs/remotes/origin/master
```

二．用户手动合并

1. 文件合并（修改同一文件的同一区域）

现在模拟这种情况的发生。（注：我这里是首先搭建了一个Git 服务器，搭建方法可见<http://blog.csdn.net/u012150179/article/details/17029731>）这里新建了从share版本库clone内容的两个用户user1和user2。并同时修改文件conflict_test文件的第二行，在user1执行push后，user2执行push会发生如下情形：

```
Administrator@SQ4U20E0QRJPEH1 ~/remotproject/user2/share (master)
$ git push origin master
To gm@192.168.7.7:repo/share
! [rejected]        master -> master (fetch first)
error: failed to push some refs to 'gm@192.168.7.7:repo/share'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
```

图1

在log信息中提示使用git pull，那尝试git pull，出现一下结果：

```
Administrator@SQ4U20E0QRJPEH1 ~/remotproject/user2/share (master)
$ git pull
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), done.
From 192.168.7.7:repo/share
   58e3457..26d0e91  master      -> origin/master
Auto-merging conflict_test1
CONFLICT (content): Merge conflict in conflict_test1
Automatic merge failed; fix conflicts and then commit the result.
```

图2

得到的信息是：“自动merge失败，需要修复conflict后才能commit。”也就是此时已经出现了在上面“Git智能自动合并”的情形之外的情况，现在Git已经不能执行自动合并，此时需要用户手动合并解决问题。

可以查看版本库状态：

```

Administrator@SQ4U20E0QRJPEH1 ~/remotproject/user2/share (master|MERC
git status
On branch master
Your branch and 'origin/master' have diverged,
and have 1 and 1 different commit each, respectively.
(use "git pull" to merge the remote branch into yours)

You have unmerged paths.
(fix conflicts and run "git commit")

Unmerged paths:
(use "git add <file>..." to mark resolution)

        both modified:      conflict_test1

no changes added to commit (use "git add" and/or "git commit -a")

```

图3

问题解决方法：

这里比较好的方式是借助mergetool，输入

git mergetool

得到

```

Administrator@SQ4U20E0QRJPEH1 ~/remotproject/user2/share (master|MERCING)
$ git mergetool
Merging:
conflict_test1

Normal merge conflict for 'conflict_test1':
  {local}: modified file
  {remote}: modified file
Hit return to start merge resolution tool (kdiff3):

```

图4，回车

出现kdiff3图形界面，其中A区为修改前，B区为本地（local）版本库修改，C区为远程版本库（remote）信息，在最下方点击右键选择需要使用的版本库，修改后保存退出即可。

这里就完成了conflict的手动合并。

之后：

git commit -m ""

git push origin master

即可。查看提交日志：

```
Administrator@SQ4U20EQRJPEH1 ~/remotproject/user2/share (master)
$ git log --graph --oneline
*   e4d6238 fix conflict.
|  \
|   * 26d0e91 Mosify the second line.
|   * 29a7d84 Modify the second line.
|  /
| * 58e3457 Say hello to user1
| * e54dbd6 Add sth to conflict_test.
| * 021e5b7 Merge branch 'master' of 192.168.7.7:repo/share
|  \
|   * c099d76 Add user2.txt
|   * 7197ce3 Add conflict_test.
|   * 6c25924 Add user2.txt
|  /
| * 16d70d1 Initial commit.
```

图5

可以发现最新提交是前两次提交的合并。

2. 树合并（不同成员对同一文件重命名）

关于树冲突，出现的原因是因为同时对一个文件进行了重命名。也可以使用mergetool修复冲突，但是更直接的方法是直接使用git rm删除想删除的文件，使用git add将需要的文件加到暂存区进而commit。

在文件合并和树合并中还有一个很有用的命令是：

git ls-files -s

可查看暂存区文件。