

## 解决冲突

阅读: 598660



人生不如意之事十之八九，合并分支往往也不是一帆风顺的。

准备新的 `feature1` 分支，继续我们的新分支开发：

```
$ git checkout -b feature1
Switched to a new branch 'feature1'
```

修改 `readme.txt` 最后一行，改为：

```
Creating a new branch is quick AND simple.
```

在 `feature1` 分支上提交：

```
$ git add readme.txt

$ git commit -m "AND simple"
[feature1 14096d0] AND simple
1 file changed, 1 insertion(+), 1 deletion(-)
```

切换到 `master` 分支：

```
$ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
(use "git push" to publish your local commits)
```

Git还会自动提示我们当前 `master` 分支比远程的 `master` 分支要超前1个提交。

在 `master` 分支上把 `readme.txt` 文件的最后一行改为：

Creating a new branch is quick & simple.

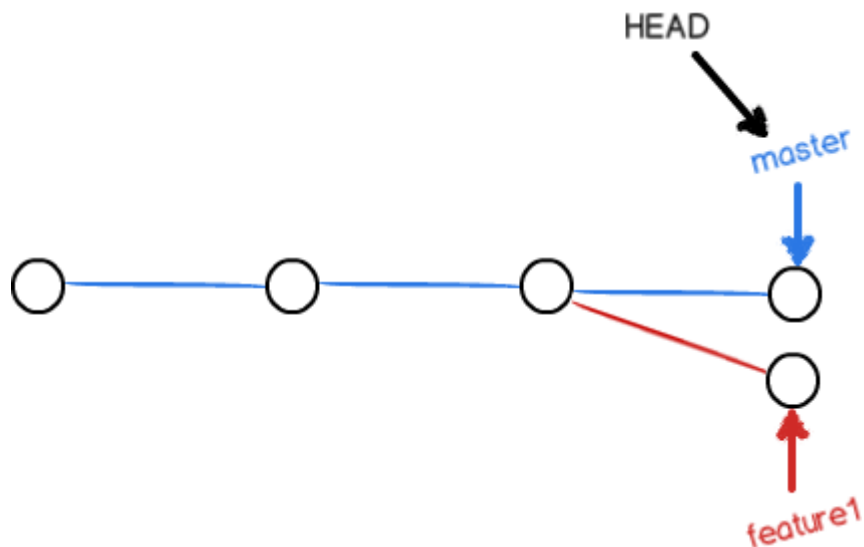
目录



提交：

```
$ git add readme.txt
$ git commit -m "& simple"
[master 5dc6824] & simple
1 file changed, 1 insertion(+), 1 deletion(-)
```

现在，`master` 分支和 `feature1` 分支各自都分别有新的提交，变成了这样：



这种情况下，Git无法执行“快速合并”，只能试图把各自的修改合并起来，但这种合并就可能会有冲突，我们试试看：

```
$ git merge feature1
Auto-merging readme.txt
CONFLICT (content): Merge conflict in readme.txt
Automatic merge failed; fix conflicts and then commit the result.
```

果然冲突了！Git告诉我们，`readme.txt` 文件存在冲突，必须手动解决冲突后再提交。`git status` 也可以告诉我们冲突的文件：

```
$ git status
On branch master
Your branch is ahead of 'origin/master' by 2 commits.
(use "git push" to publish your local commits)
```

目录  



```
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)
```

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

```
    both modified:   readme.txt
```

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

我们可以直接查看readme.txt的内容：

```
Git is a distributed version control system.
Git is free software distributed under the GPL.
Git has a mutable index called stage.
Git tracks changes of files.
<<<<<< HEAD
Creating a new branch is quick & simple.
=====
Creating a new branch is quick AND simple.
>>>>>> feature1
```

Git用<<<<<<，=====，>>>>>>标记出不同分支的内容，我们修改如下后保存：

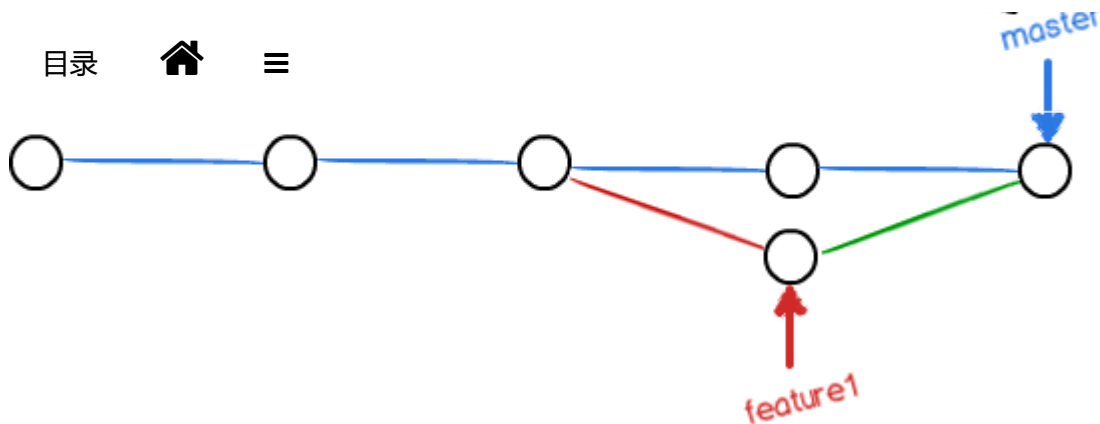
```
Creating a new branch is quick and simple.
```

再提交：

```
$ git add readme.txt
$ git commit -m "conflict fixed"
[master cf810e4] conflict fixed
```

现在，master分支和feature1分支变成了下图所示：





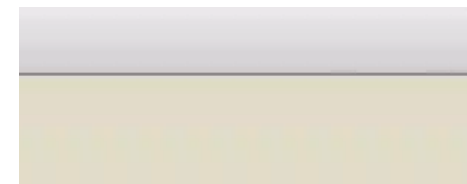
用带参数的 `git log` 也可以看到分支的合并情况：

```
$ git log --graph --pretty=oneline --abbrev-commit
*   cf810e4 (HEAD -> master) conflict fixed
| \
|  * 14096d0 (feature1) AND simple
* | 5dc6824 & simple
| /
* b17d20e branch test
* d46f35e (origin/master) remove test.txt
* b84166e add test.txt
* 519219b git tracks changes
* e43a48b understand how stage works
* 1094adb append GPL
* e475afc add distributed
* eaadf4e wrote a readme file
```

最后，删除 `feature1` 分支：

```
$ git branch -d feature1
Deleted branch feature1 (was 14096d0).
```

工作完成。





0:00 / 1:30

## 小结

当Git无法自动合并分支时，就必须首先解决冲突。解决冲突后，再提交，合并完成。

解决冲突就是把Git合并失败的文件手动编辑为我们希望的内容，再提交。

用 `git log --graph` 命令可以看到分支合并图。

感觉本站内容不错，读后有收获？

还可以分享给朋友

🐦 分享到微博

⏪ 上一页

下一页 ⏩

## 评论



### 关于删除分支

旧雨衣Iridescent created at 2018-1-26 15:02, Last updated at 2天前

解决冲突以后必须删除分支吗？我在测试过程中解决完冲突以后没有删除该分支，然后下次继续在分支上修改内容，合并到主分支上时又出现了冲突，可是我没有在主分支上修改内容啊。。。

然后我做对比测试，在解决冲突以后删除该分支，下次重新生成一个新的分支修改内容，再合并到主分支上就没有出现冲突。



### 为伊消得人憔悴

Created at 2018-1-27 0:40, Last updated at 2018-1-27 0:40

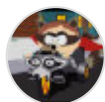
应该不需要的 我照着题主的方法试了一遍没有问题呀



### 张燕茹10900

Created at 2018-8-1 10:12, Last updated at 2018-8-1 10:12

你这个原因应该是，解决冲突只解决的主分支master的。分支应该没有变化，还是冲突前的内容。所以再次提交会报错



### 长宁特产

Created at 2-24 21:16, Last updated at 2-24 21:16

这么解释吧，主分支必须落后于分支一次操作才行。

现在解决冲突的方式，就是更新一次master，再把分支删掉。

如果拟更新了master，继续保持分支，再merge分支feature1，出现的局面就是本章节一开始出现的，master和分支feature1各走各路一次

秀秀58937

Created at 2天前, Last updated at 2天前

我试了，冲突后，在master上修改，再add,commit,其实不管feature1和master内容是否一致，都认为解决了。感觉只是告诉你冲突，只要你手动解决成你要的内容，就认为冲突管理结束了，然后就是合并成功。其实master还是master,featur1还是feature1

秀秀58937

Created at 2天前, Last updated at 2天前

即使解决冲突后，master再怎么修改add,commit,都是默认已经和feature1合并的。除非feature1的内容修改后再add,commit.想了下作者的初衷应该就是告诉你合并的文件有冲突，你知道改就好了，具体怎么改作者是不管的。所以不管master怎么改，都认为已经是合并完毕的。

### 全部讨论

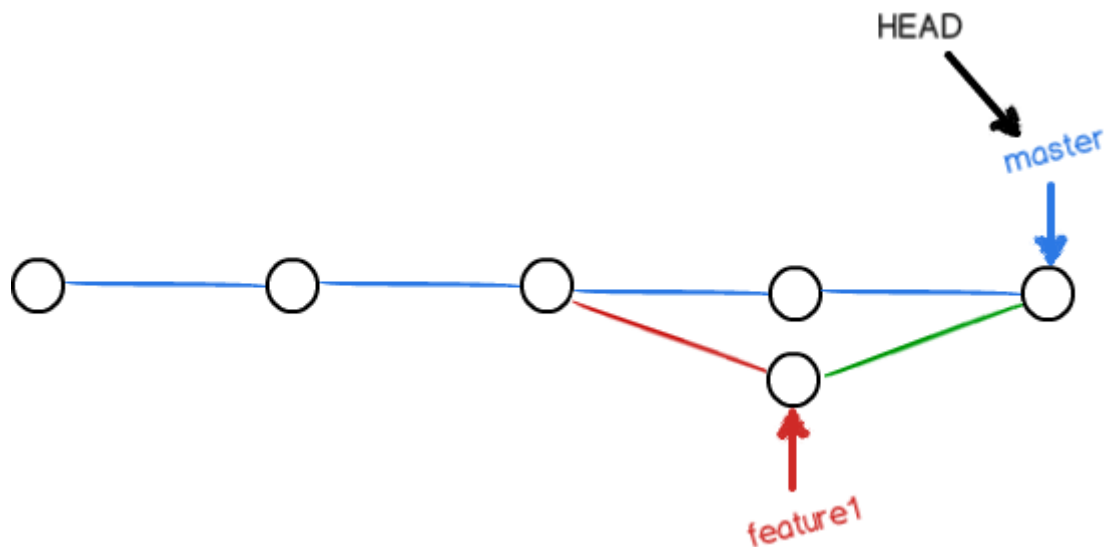
↩ 回复



### 图中错误疑问

图灵不灵 created at 1-22 17:22, Last updated at 2-24 22:16

在master上解决完冲突，提交到版本库后，feature1和master的图中，feature1到master的最新commit间应该没有连线吧？个人理解，不知道对不对





图灵五灵

Created at 1-22 17:41, Last updated at 1-22 17:41

其实有连线问题不大，因为feature1的指针还是之前的commit，但是连线感觉有些误解

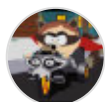


廖雪峰

Created at 1-22 21:52, Last updated at 1-22 21:52

有的，那个连线是master指的commit，它有两个前序节点

feature1的指针不一定会移到master指的commit，它可以自主独立发展



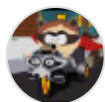
长宁特产

Created at 2-24 21:08, Last updated at 2-24 21:08

git checkout feature1你就懂啦

那张图是相对主线master而言的，它的当前状态有两个，一个来自master一个来自feature1，更确切的来说绿线应该往左。

如果不删除feature1分支的话，这条分支倒是也可以继续往下走



长宁特产

Created at 2-24 22:16, Last updated at 2-24 22:16

“它的当前状态有两个”，当前就是“之前的意思”，链路逻辑就是这样的，所以冲突解决后merge的时候不会报错

≡ 全部讨论

↩ 回复



git冲突问题

笨二CHINA created at 2018-8-20 11:53, Last updated at 1-11 10:46

按照上述进行操作。报错。

```
$ git commit readme.txt -m "conflict fixed"
fatal: cannot do a partial commit during a merge.
```

主分支 未解决



LIANG-

Created at 2018-9-3 23:35, Last updated at 2018-9-3 23:35

老铁，你输入的命令出现问题，再检查一遍....

胡二刀1988

Created at 2018-12-26 21:54, Last updated at 2018-12-26 21:54

```
$ git commit readme.txt -m "conflict fixed"
fatal: cannot do a partial commit during a merge.
```

```
H@H MINGW64 ~/learngit (master|MERGING)
$ git commit -m "conflict fixed"
[master 3c65847] conflict fixed
```

处在(master|MERGING)状态的时候,不可以使用部分提交,也就是不能使用指定文件名这种提交方式.  
提示是这样的,你可以再试试

舟渺江雨疏

Created at 1-11 10:46, Last updated at 1-11 10:46

在已经解决合并冲突并希望再次提交的时候,是不能使用部分提交的  
这里应该直接使用 `git commit -m "conflict fixed"`

全部讨论

回复

这样解决了分支冲突，但是文件还是有问题的啊明非知是非 created at 2018-11-25 22:25, Last updated at 2018-12-14 0:36

```
$ cat readme.txt
```

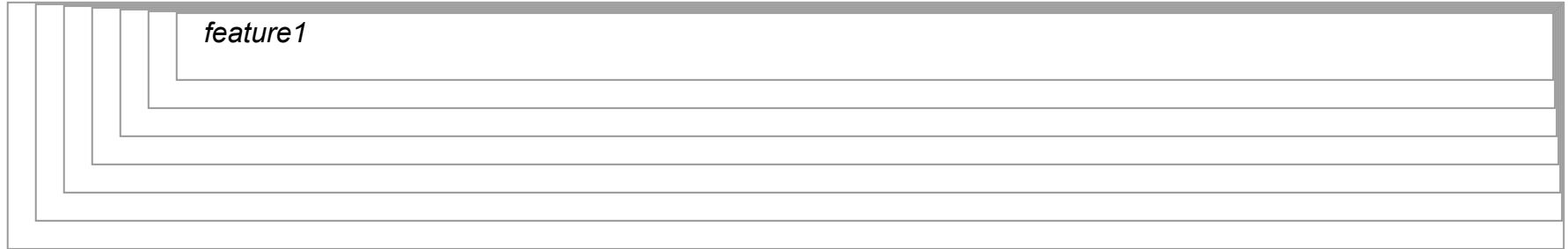
Git is a distributed version control system.

Git is a free software distributed under the GPL

Git is a free software distributed under the GPL.  
It has a mutable index called stage.  
Git tracks changes.  
<<<<<<< HEAD

# Creating a new branch is quick and simple.

Creating a new branch is quick AND simple.



小疯疯888

Created at 2018-12-14 0:36, Last updated at 2018-12-14 0:36

这个有卵用啊，放着UI工具不用，蛋疼啊。

≡ 全部讨论

↩ 回复



mac下xcode软件直接打开文件，不显示冲突标记，

我有疫苗 created at 2018-12-10 14:33, Last updated at 2018-12-10 14:33

用xcode打开文件显示的只有master的内容，并没有任何的冲突标记，但是git提示文件有冲突。

git checkout feature1

error: 您对下列文件的本地修改将被检出操作覆盖：

README.md

请在切换分支前提交或贮藏您的修改。

终止中

MFWdeMacBook-Pro:test mfw\$ git branch

dev

feature1

[全部讨论](#)[回复](#)

### git log 单独输入无法退出

体贴的模型 created at 2018-4-26 12:29, Last updated at 2018-11-16 16:07

我在linux系统下输入git log,发现太长,结尾只是<end>这样,baidu搜索退出方法,是ctrl+c,linux下没用,其实是q,就可以了不知道是不是完全的对



Lifangchao668

Created at 2018-7-20 17:20, Last updated at 2018-7-20 17:20

我windows的git Bash也是这样，发现退不出，我就重新开了个终端，看了你这个才知道Q可以退出，原来git Bash跟linux有点像啊！谢了



榭寄君

Created at 2018-11-16 16:07, Last updated at 2018-11-16 16:07

什么叫跟linux有点像。git原本就是为了linux开发的

[全部讨论](#)[回复](#)

### 文中漏写了一步操作

外城 义凡 created at 2018-11-8 11:12, Last updated at 2018-11-8 11:12

文中漏掉了一步操作：feature1分支提交到本地仓库后，还需要合并到master分支  
否则master分支的本地仓库和远程仓库不会有差别

[全部讨论](#)[回复](#)



## 分支提交内容冲突

不会TC的猫 created at 2018-9-17 17:25, Last updated at 2018-9-17 17:25



- 1.经过自己的测试，在合并出现问题之后，master分支的装填是merging,代表正在合并过程中
- 2.此时，两个文件已经是一个完整体了，虽然还是有冲突内容，修改完冲突内容之后，提交，此时两个文件已经合并成功了

≡ 全部讨论

↩ 回复



## 在合并发生冲突之后再次git add readme.ad git commit -m 'fixed commit'能合并分支

欧皇寿命极长 created at 2018-7-22 20:57, Last updated at 2018-8-18 17:54

在合并发生冲突之后没有去修改文件，直接再次git add readme.ad git commit -m 'fixed commit'就合并成功了，这样是对的吗。



## 诺伊佩拉的眼泪

Created at 2018-8-18 17:54, Last updated at 2018-8-18 17:54

没有修改冲突的话，文件里会有提醒的，真实情况下代码会无法运行的。  
当然add和commit是可以正常提交，git不会判断代码是不是可以正常运行

≡ 全部讨论

↩ 回复



## 大面积实验后，发现即便两个库修改的完全一样，也不能合并成功；删除分支后，无论如何回溯，都找不到该分支了，这样正常吗？

济南自媒体 created at 2017-12-6 18:52, Last updated at 2018-8-13 22:29

大面积实验后，发现即便两个库修改的完全一样，也不能合并成功；删除分支后，无论如何回溯，都找不到该分支了，这样正常吗？

```
oixqiu@oixQiuAcerWin10 MINGW64 /i/qiuData/gitRepository1 (dev)
```

```
$ git branch
```

- dev master

```
oixqiu@oixQiuAcerWin10 MINGW64 /i/qiuData/gitRepository1 (dev)
```

```
$ git add test.txt
```

```
git commit -m "Add branch ."
```

```
[dev 938b04e] Add branch .
```

1 file changed, 2 insertions(+), 1 deletion(-)

oixqiu@oixQiuAcerWin10 MINGW64 /i/qiuData/gitRepository1 (dev)

```
$ cat test.txt
```

Hello git .

Test branch .

```
oixqiu@oixQiuAcerWin10 MINGW64 /i/qiuData/gitRepository1 (dev)
```

© Nitron

▼ Read More



廖雪峰

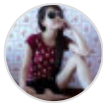
Created at 2017-12-6 19:07, Last updated at 2017-12-6 19:07

提问的时候，贴要点，这种篇幅，没人会看的。

DeppWangXQ

Created at 2018-2-2 23:09, Last updated at 2018-2-2 23:09

廖大大说话就是言简意赅



欧阳筱苒

Created at 2018-2-27 17:05, Last updated at 2018-2-27 17:05

oixqiu@oixQiuAcerWin10 MINGW64 /i/qiuData/gitRepository1 (master)

[illegible]

你这句不知道 干什么

这个之前 代码都没问题 也没报错

这个之后就开始了



## 爱吃西瓜的酱



友情的链接

Created at 2018-8-13 22:29, Last updated at 2018-8-13 22:29



兄弟你这个确实长的过份了。。

≡ 全部讨论

↩ 回复

## 发表评论

登录后发表评论

廖雪峰的官方网站©2017 v641f3f8

Powered by [iTranswarp.js](#)

本网站运行在[阿里云](#)上并使用[阿里云CDN](#)加速。



友情链接: [中华诗词](#) - [阿里云](#) - [金山云](#) - [SICP](#) - [4clojure](#) - [linuxprobe](#)