

# git merge中冲突解决的摘要

混帐

最后更新时间：2018-06-22来自Joe

如何解决git合并冲突。如果您使用git和git merge分支，则无法避免冲突及其解决问题的的工作。

这一次，我们总结了如何解决冲突。

## 目录

大约1 git合并

2 git merge中的冲突

如何解决3 Git中的冲突

3.1 1检查冲突位置

3.2纠正必要的部件

3.3第3阶段修复索引并重新启动

4取消git merge

5参考链接

## 关于git merge

git merge将特定提交的历史记录合并到当前分支中。在实际使用中，您经常指定分支名称。分支名称是该分支的第一次提交的“引用”，因此实际指定的是提交。

```
// ブランチ topic を現在のブランチにマージする
$ git merge topic
```

具体而言，例如，将主题分支与下面的主分支“G”合并

```
      A---B---C topic
      /
      D---E---F---G master

// masterに、topicブランチをマージする
$ git merge topic
```

因此，作为“H”的结果生成新的提交。A，B，C的变化被合并到主分支中。

```
// 結果
```

```
      A---B---C topic
     /         \
    D---E---F---G---H master
```

有关git merge命令的基本规范，请参阅此文章。

- [git合并的操作规范和主要选项的总结](#)

## git合并中的冲突

通过合并合并更改时，如果更改了同一文件的重复部分，则更改将发生冲突并导致称为“冲突”的条件。

```
// マージの実行
$ git merge branch-B

// 出力: コンフリクトによる中断を伝える
Auto-merging index.php
CONFLICT (content): Merge conflict in index.php
Automatic merge failed; fix conflicts and then commit the result.
```

在这两个历史中，我应该采用哪些代码更改？由于Git本身无法确定问题，因此开发人员必须指定或调整以下代码。

如果在运行git merge时发生冲突，则会暂停git merge进程，让开发人员完成其余的工作。这是Git中的一个特殊状态，它将处于“合并中断”的状态。此时，开发人员将选择一个。

- 通过运行git commit来解决代码冲突并提交合并提交
- 停止合并和恢复

除了执行上述操作之外，其他Git操作是不可能的。

## 如何解决Git中的冲突

我们来试试吧。

## 1 确认冲突位置

在与git merge发生冲突之后，您可以通过运行“git status”找出冲突的位置。

```
// ステータスを確認
$ git status

// 出力
On branch branch-A
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:   index.php

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

Unmerged path列出发生冲突的文件。

## 2 纠正必要的部件

如果您发现哪个文件存在冲突，请直接打开该文件并进行修复。

运行git merge时，冲突由“<<<< HEAD""===="">>>>主题”在工作树的文件中突出显示，所以“>在文件中如果您搜索>>>等系列，您将立即找到问题部分。

```
<div class="content-area front">

<<<<<<< HEAD
  <h1>This is added from branch-A</h1>
=====
  <h1>This is added from branch-B</h1>
>>>>>>> branch-B
```

如果你不习惯它，你会感到有点紧张，但没有什么可担心的。HEAD是当前的分支。下面是指定为“git merge branch-B”的分支。

Git写入工作树中的文件。在编辑器中，删除不必要的部分并整齐地排列它们。

## 3 修改到阶段索引，重新提交

索引仍会在合并之前记录文件的状态。Git添加已完成修复冲突的文件。

```
$ git add index.php
```

再次检查git状态。

```
$ git status

// 出力
On branch branch-B
All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)

Changes to be committed:

  modified:   index.php
```

Git说：“所有冲突都已解决，但你仍在合并。”

您注意到您已解决冲突并运行git add以将修复程序转换为索引。之后，如果执行git commit，则完成合并提交以解决冲突。

```
$ git commit

// エディタでメッセージを編集・・・
Merge branch 'branch-B' into branch-A

# Conflicts:
#   index.php
#
# It looks like you may be committing a merge.
# If this is not correct, please remove the file
#   .git/MERGE_HEAD
# and try again.
```

成功地，已创建合并提交。

```
* 2787f68 (HEAD -> refs/heads/branch-A) Merge branch 'branch-B' into branch-A
|\
| * 9e2f7d0 (refs/heads/branch-B) fix something again.
* | 733fc68 fix something.
|/
* 1034887 (refs/heads/master) Update something again.
* 9031694 Update something.
```

在上一次成功的git commit中，git merge过程完成了。

在上面，git add git commit是手动执行的，但**如果执行“git merge --continue”，git add和git commit将自动执行。**结果是一样的。

## 取消git merge

运行git merge后，您可以取消它并取消它，因为很难解决冲突。

```
$ git merge --abort
```

这就是解决git合并冲突的问题。