**Git操作 - rebase命令详解**

在 Git 中合并来自不同分支的修改主要有两种方法：merge 和 rebase。

在之前的文章当中我们介绍了git merge命令的用法，通过git merge命令我们可以对两个分支进行合并操作。这样我们可以很方便地进行协同开发了，每个人都在自己的分支下开发代码，开发完毕之后再合并到master分支，通过这种方式可以保证大家的代码不会陷入混乱。

但是这样也会有隐含的缺陷：

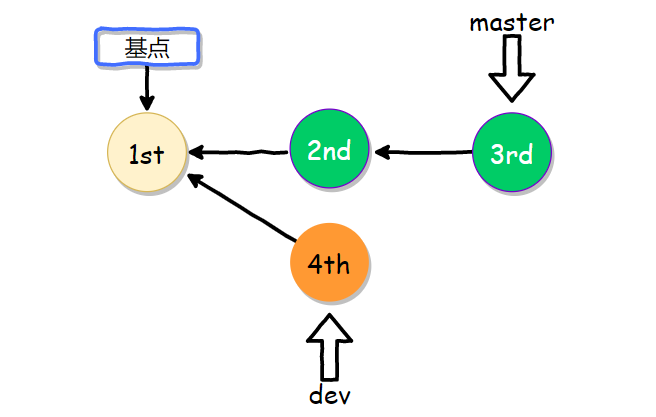
* 一个是提交历史记录的不完整，回退的时候可选择性减少。
* 一个是历史提交记录非常的混乱，会让你眼花缭乱，欲仙欲死。

为了解决这些头疼的问题，我们就需要用到git rebase操作。

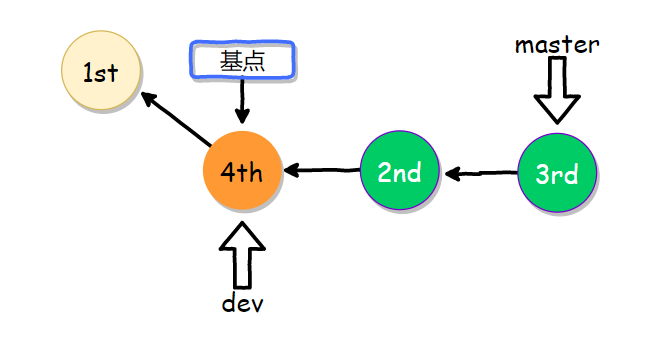
**1、git rebase命令简介**

rebase有两个意思：**变基**、**衍合**，即变换分支参考的基准点。默认情况下，一个分支会以该分支上的第一次提交作为基点。

如下图所示：master分支默认以提交1st作为基点：



如果以提交4th作为master分支的基点，master分支就会变为：



这个变化基准点的过程就称之为变基（rebase）。

**接下来详细介绍一下rebase命令**：

官方对于rebase的描述为：“git-rebase: Forward-port local commits to the updated upstream head”— git doc

翻译一下，就是将你在某个分支上的所有提交记录，移花接木到另一个分支上。

这边需要强调一个概念：reapply，使用rebase并不是简单地像你进行剪切复制一样，rebase命令会依次地，将你所要操作分支上的所有提交，应用到目标分支上。

也就是说，实际上在执行rebase命令的时候，有两个隐含的注意点：

1. 在重放之前提交的时候，Git会创建新的提交。也就是说即使你重放的提交与之前的一模一样，Git也会将他当做新的独立的提交进行处理。
2. git rebase命令并不会删除旧的提交。也就是说，你在对某个分支执行了rebase操作之后，老的提交仍然会存放在.git文件夹的objects目录下。

**2、merge与rebase的区别**

git rebase 命令与git merge 命令的功能十分相似，不过二者的工作方式有着显著的差异。

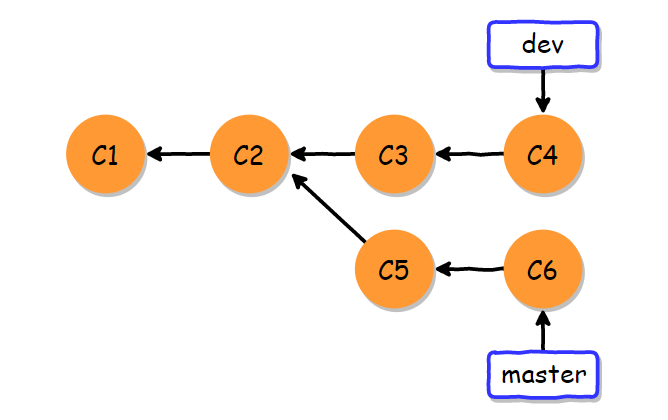
比如：将A和B两分支进行合并：

* 在A分支上执行git merge B，表示的是将B分支**合并到**A分支上；
* 而在A分支上执行git rebase B，则表示将A分支通过变基**合并到**B分支上；（看不懂，往下看）

**（1）采用merge合并分支**

现在有两个分支master分支和dev分支，如果想要将dev分支**合并到**master分支上。根据三方合并原则，需要在c4、c6和它们的公共父提交节点c2的基础上进行合并。

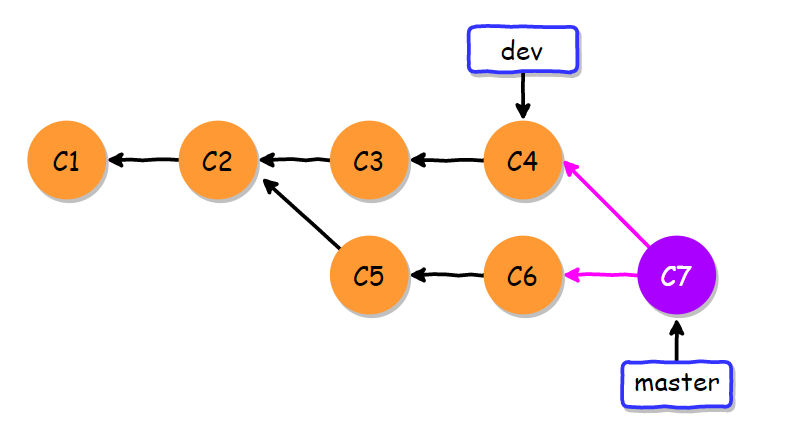
如下图：



合并后生成一个新的提交c7，该提交有两个父节点c4和c6。

具体的合并方式为：如果没有冲突git就会自动采用Fast-forward方式进行合并，如果有冲突就进行典型合并，解决冲突后再进行手动合并。

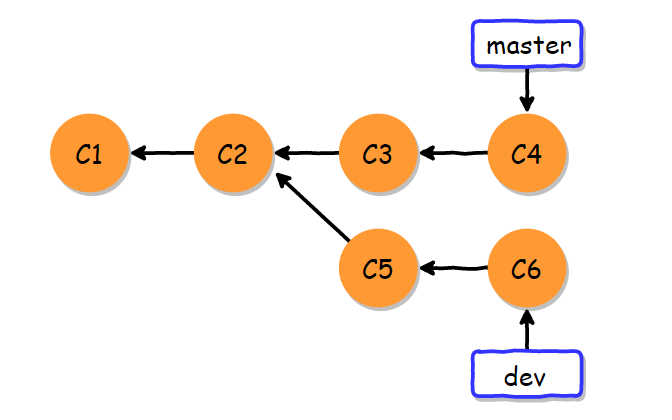
如下图：



**（2）采用rebase合并分支**

我们希望dev分支变基合并到master分支上，所以首先切换到dev分支（注意这里与采用merge方法时所在的分支相反）：

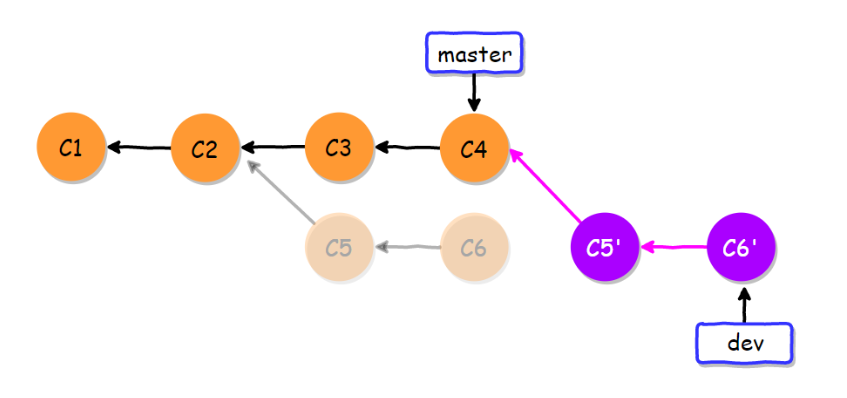
如下图：



执行命令：

|  |
| --- |
| Bash # 切换分支 $ git checkout dev  # 再进行rebase合并 $ git rebase master |

合并后的结果，如下图：

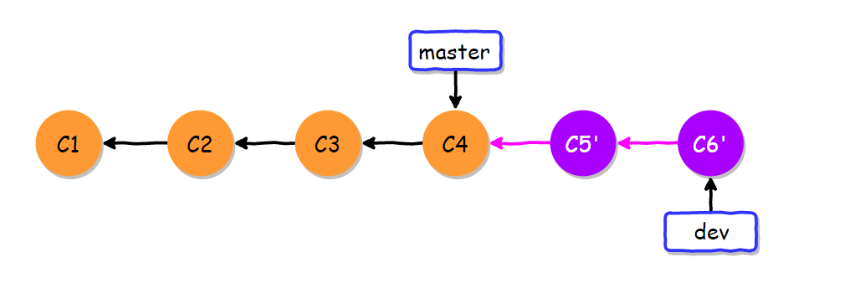


**注意**：

被合并的master分支保持不动，而合并它的dev分支，将自己的提交作为补丁（patch）一个个应用（applying）到master指向的分支上。

在这个过程中Git会自动创建c5'和c6'提交。原来的c5和c6提交就没用了，会被git gc回收（就是不显示在历史记录中了，但是数据还在，应该用git reflog命令可以看到）。

合并后分支Git版本库中的提交记录变成了一条直线，如下图：



**（3）总结rebase命令**

基于以上表述，我们可以得出git rebase命令的工作流程：

* 在对特定分支进行rebase操作之后，其等效于创建了新的提交。
* 并且老的提交也没有被销毁，只是简单地不能再被访问或者使用。
* 之前分支的章节我们曾经提及，分支只是一个执行提交的指针。因此如果没有分支或者Tag指向某个提交，该提交将无法再被访问使用，但是该提交会一直存在于你的文件系统中，占用着你的磁盘存储。

**git rebase master命令的意思**：

该指令翻译过来就是：重新定位当前工作的分支，以master 分支做为新的基准点，即使用 master 分支所指向的提交，作为我新的基准点。

* git merge 要合并的分支 命令：切换的主要的分支，把其他的分支合并过来。
* git rebase 变基的目标分支 命令：切换到被合并的分支上，把该分支上的提交变基到目标分支上。

|  |
| --- |
| 参考：   * [https://blog.csdn.net/qq\_43425487/article/details/105620270](https://links.jianshu.com/go?to=https%3A%2F%2Fblog.csdn.net%2Fqq_43425487%2Farticle%2Fdetails%2F105620270) * [https://blog.csdn.net/u012925172/article/details/84855995](https://links.jianshu.com/go?to=https%3A%2F%2Fblog.csdn.net%2Fu012925172%2Farticle%2Fdetails%2F84855995) |

作者：繁华似锦Fighting 链接：https://www.jianshu.com/p/9e5505cf1afe 来源：简书 著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。