

Robin Hu的专栏

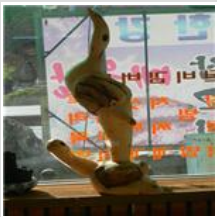
Let's practice improving every day

目录视图

摘要视图

RSS 订阅

个人资料



hudashi

恒

访问：2589883次

积分：23604

等级：BLOG > ?

排名：第146名

原创：226篇 转载：237篇

译文：177篇 评论：269条

文章搜索

文章分类

- Android基础 (80)
- 开发专题 (34)
- 深入研究Android (33)
- Android的UI开发 (17)
- 数据适配器 (22)
- Android绘画 (9)
- Android网络 (9)
- 问题小汇 (24)
- 电话簿 (18)
- 数据持久化存储 (24)
- 数据监听 (4)
- Android内存 (15)
- Android调试 (8)
- Android其他 (23)
- JAVA基础 (18)
- JAVA线程相关 (20)
- JAVA集合容器 (33)

学院APP首次下载，可得50C币！

欢迎来帮助开源“进步”

当讲师？爱学习？投票攒课吧

认识Atlassian Datacenter

公告】有奖试读—增长黑客

git rebase简介(基本篇)

标签：git merge conflict c patch

2012-06-14 22:09

48117人阅读

评论(7)

收藏

举报

分类：版本控制 (44)

原文：

[http://gitbook.liuhui998.com/4\\_2.html](http://gitbook.liuhui998.com/4_2.html)

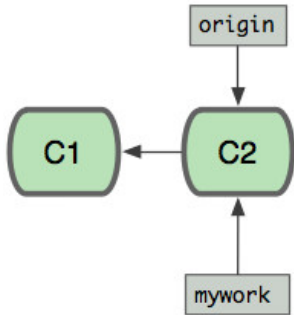
一、基本

git rebase用于把一个分支的修改合并到当前分支。

假设你现在基于远程分支"origin"，创建一个叫"mywork"的分支。

\$ git checkout -b mywork origin

假设远程分支"origin"已经有了2个提交，如图



现在我们在这个分支做一些修改，然后生成两个提交(commit).

\$ vi file.txt

\$ git commit

\$ vi otherfile.txt

\$ git commit

...

但是与此同时，有些人也在"origin"分支上做了一些修改并且做了提交了. 这就意味着"origin"和"mywork"这两个分支各自"前进"了，它们之间"分叉"了。

JAVA执行器 (9)  
JAVA虚拟机 (7)  
J2SE (2)  
J2SE之lib (3)  
C和C++语言 (28)  
Linux基础 (32)  
Linux内核 (12)  
数学相关 (5)  
网络基础 (3)  
术语 (8)  
DOS批处理 (25)  
工具箱 (5)  
JDK命令 (7)  
我的DOS脚本库 (4)  
工具箱2 (9)  
Linux脚本集 (1)  
数字签名 (17)  
数据结构 (3)  
版本控制 (45)  
数据库 (2)  
设计模式 (7)  
JavaScript (1)  
其他 (17)  
网络编程 (2)

#### 阅读排行

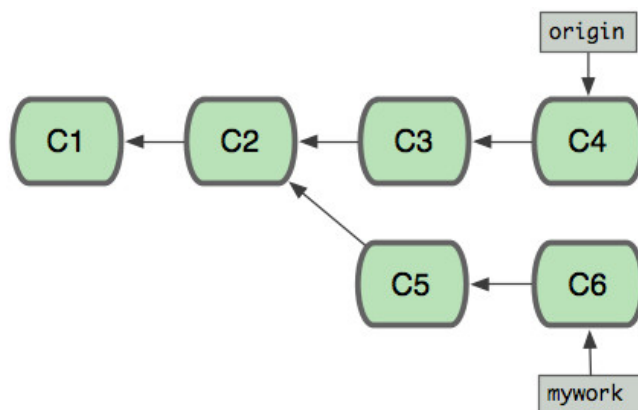
Git fetch和git pull的区别 (148132)  
git clone简介 (126796)  
如何更改linux文件的拥有 (93644)  
git merge简介 (76322)  
git reset简介 (75385)  
git checkout简介 (65451)  
git revert和git reset的区别 (49498)  
git rebase简介(基本篇) (48113)  
git commit简介 (47849)  
git分支的合并 (46002)

#### 评论排行

Git fetch和git pull的区别 (11)  
pthread\_cond\_signal和p (10)  
Android手机通过wifi进行 (9)  
Android手机通过wifi进行 (9)  
git clone简介 (9)  
ContentProvider简介 (7)  
git rebase简介(基本篇) (7)  
线程属性pthread\_attr\_t (7)  
Android中如何查看内存 (7)  
Android中图片占用内存 (6)

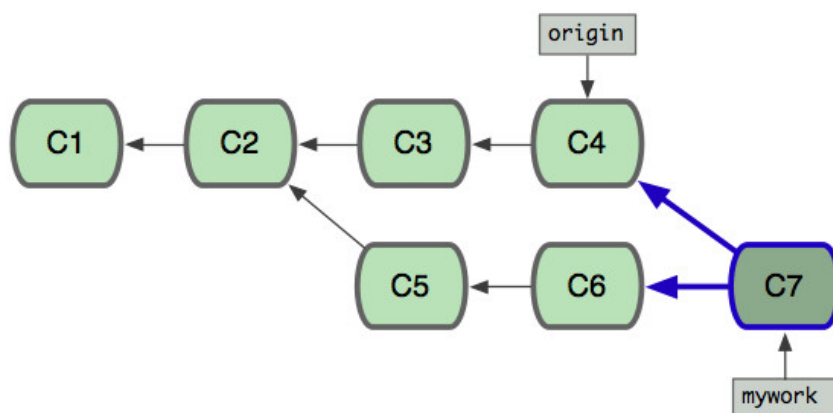
#### 推荐文章

\*一款炫酷Loading动画--加载失败



在这里，你可以用"pull"命令把"origin"分支上的修改拉下来并且和你的修改合并；结果看起来就像一个新的"合并的提交"(merge commit):

## git merge



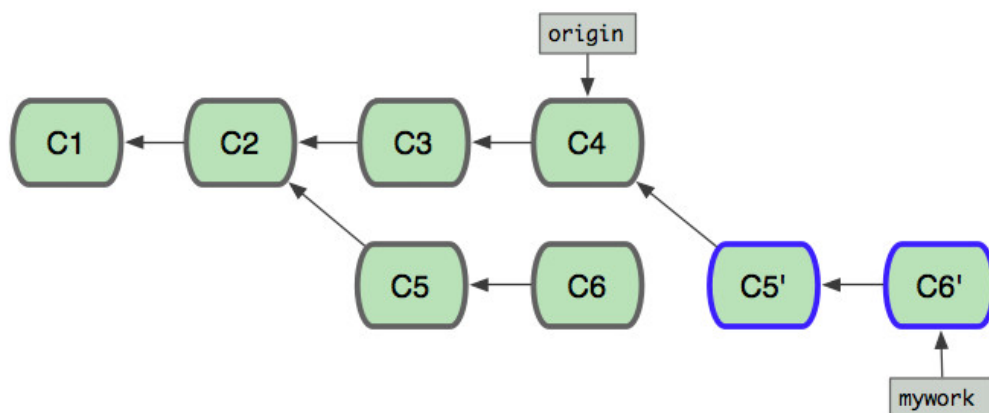
但是，如果你想让"mywork"分支历史看起来像没有经过任何合并一样，你也许可以用 git rebase:

```
$ git checkout mywork
```

```
$ git rebase origin
```

这些命令会把你的"mywork"分支里的每个提交(commit)取消掉，并且把它们临时保存为补丁(patch)(这些补丁放到".git/rebase"目录中),然后把"mywork"分支更新为最新的"origin"分支，最后把保存的这些补丁应用到"mywork"分支上。

## git rebase



当'mywork'分支更新之后，它会指向这些新创建的提交(commit),而那些老的提交会被丢弃。如果运行垃圾收集命令(pruning garbage collection), 这些被丢弃的提交就会删除。(请查看 git gc)

\*图解堆算法、链表、栈与队列  
(多图预警)

\*Eclipse快捷键 10个最有用的快捷键

\*基于fiddler插件开发的移动测试网络监控与分析

\*拉开大变革序幕(中): Docker演武场

\*没有躲过的坑--正则表达式截取字符串

## 最新评论

Logcat源码分析(一)

daemoncrafter: 学习了 感谢分享

Android Lint 检查规则的定制 (基  
hottest518: /(T o T )/~太高级,看  
不懂

Android手机通过wifi进行数据传  
a513658565: 楼主你好 我也在弄  
一个关于wifi项目, 能把项目代码  
给我看看么, 谢谢。  
qq:294106453@qq...

sql的limit

郝姬友: 这个实例5写错了吧, 应  
该将select \* from class limit 10 ,  
-1 改为s...

IntentService简介

BHGXB: 怎么说呢 整篇文章色  
彩运用倒是挺丰富

Android中单APK应用多进程

\_涅槃: 亲, 提出了问题, 有解决  
方案吗? 我就是要取到  
MainActivity中的成员变量。  
Binder及AID...

Android中单APK应用多进程

\_涅槃: 亲, 提出了问题, 有解决  
方案吗? 我就是要取到  
MainActivity中的成员变量。

Android中单APK应用多进程

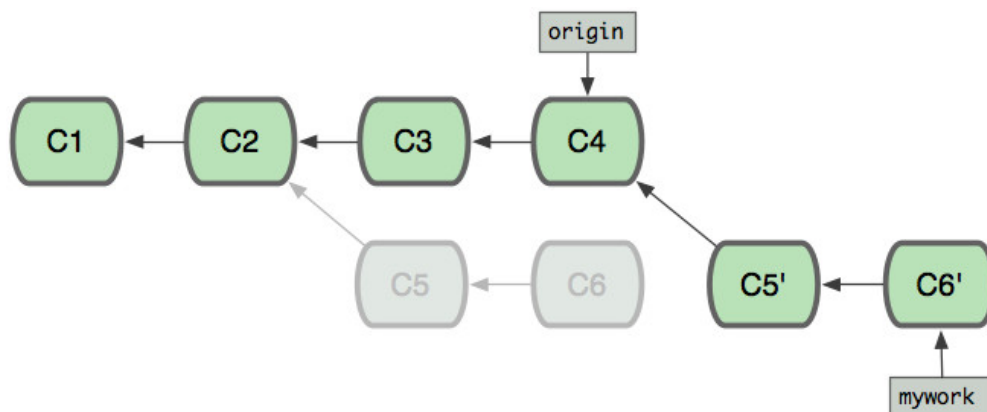
\_涅槃: 被这个鬼东西折磨了两天  
了, 头疼。。。

向上兼容和向下兼容

陈百万\_: 喷两句 貌似楼主根本没  
有理解上下兼容的逻辑上: 高版  
本下: 低版本向下兼容 指的是  
当前位置 ...

git rebase简介(基本篇)

Louis\_\_Xu: 赞 分析得很清楚



## 二、解决冲突

在rebase的过程中, 也许会出现冲突(conflict). 在这种情况下, Git会停止rebase并会让你去解决冲突; 在解决完冲突后, 用"git-add"命令去更新这些内容的索引(index), 然后, 你无需执行 git-commit, 只要执行:

```
$ git rebase --continue
```

这样git会继续应用(apply)余下的补丁。

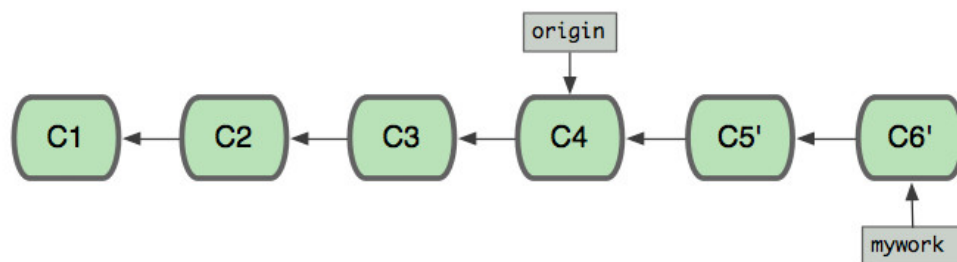
在任何时候, 你可以用--abort参数来终止rebase的行动, 并且"mywork"分支会回到rebase开始前的状态。

```
$ git rebase --abort
```

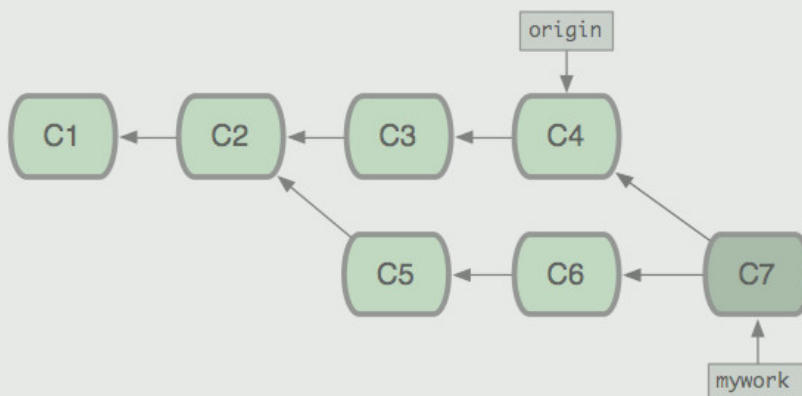
## 三、git rebase和git merge的区别

现在我们可以看一下用合并(merge)和用rebase所产生的历史的区别:

### git rebase



### git merge



当我们使用Git log来参看commit时, 其commit的顺序也有所不同。

假设C3提交于9:00AM, C5提交于10:00AM, C4提交于11:00AM, C6提交于12:00AM,

对于使用git merge来合并所看到的commit的顺序(从新到旧)是: C7, C6, C4, C5, C3, C2, C1

对于使用git rebase来合并所看到的commit的顺序(从新到旧)是: C7, C6', C5', C4, C3, C2, C1

因为C6'提交只是C6提交的克隆, C5'提交只是C5提交的克隆,

从用户的角度看使用git rebase来合并后所看到的commit的顺序(从新到旧)是: C7,C6,C5,C4,C3,C2,C1

另外, 我们在使用git pull命令的时候, 可以使用--rebase参数, 即git pull --rebase,这里表示把你的本地当前分支里的每个提交(commit)取消掉, 并且把它们临时 保存为补丁(patch)(这些补丁放到".git/rebase"目录中),然后把本地当前分支更新 为最新的"origin"分支, 最后把保存的这些补丁应用到本地当前分支上。关于git pull的更多内容请参考《git pull简介》

顶  
1

踩  
0

## 我的同类文章

### 版本控制 (44)

- git log简介
- git config基本篇
- git tag简介
- git add详解
- git rm简介
- 如何安装Git
- Git fetch和git pull的区别
- git stash简介
- git clone简介
- git reset简介

更多

## 主题推荐

git   color   合并   class   rgb

## 猜你在找

零基础实战HTML、XHTML、CSS3应用开发  
Excel高端应用:多条件约束报表自动统计与制作  
Java Swing、JDBC开发桌面级应用  
Office 365 用户管理  
使用maven构建spring mvc+mybatis应用

git rebase简介基本篇  
git rebase简介基本篇  
git rebase简介基本篇  
git rebase简介基本篇  
git rebase简介基本篇



访客机



ios就业



p2p理财



喝牛奶拉肚子



最新电子产品

## 查看评论

4楼 [Louis\\_\\_Xu](#) 2015-11-17 09:50发表



赞 分析得很清楚

3楼 [M漫游人](#) 2015-06-16 08:49发表




图里箭头的方向是不是都错了 啊。。。。

2楼 [小强就是我](#) 2015-04-17 16:50发表




大赞! 可以引申下, 如果git pull -r

Re: [hudashi](#) 2015-04-23 10:25发表

 回复小强就是我：已经引申git pull --rebase。谢谢你的建议

1楼 [WeAreYoung1990](#) 2015-03-19 11:22发表

 写得特别好！

您还没有登录,请[\[登录\]](#)或[\[注册\]](#)

\* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

- 全部主题
- Hadoop
- AWS
- 移动游戏
- Java
- Android
- iOS
- Swift
- 智能硬件
- Docker
- OpenStack
- VPN
- Spark
- ERP
- IE10
- Eclipse
- CRM
- JavaScript
- 数据库
- Ubuntu
- NFC
- WAP
- jQuery
- BI
- HTML5
- Spring
- Apache
- .NET
- API
- HTML
- SDK
- IIS
- Fedora
- XML
- LBS
- Unity
- 关闭
- Splashtop
- UML
- components
- Windows Mobile
- Rails
- QEMU
- coremail
- OPhone
- CouchBase
- 云计算
- iOS6
- Rackspace
- W
- Compuware
- 大数据
- aptech
- Perl
- Tornado
- Ruby
- Hibernate
- Angular
- Cloud Foundry
- Redis
- Scala
- Django
- Bootstrap



单机小游戏

