

Git pull 使用

emacsun

Github支持协作代码开发管理,会经常遇到需要更新别人的代码或者在不同的电脑上更新自己的代码。那么使用git pull命令即可更新代码。git pull命令的作用是，取回远程主机某个分支的更新，再与本地的指定分支合并。git pull可以接受很多参数，本文总结一些常见的用法。

1 直接从远程主分之更新代码

```
git pull
```

以上相当于 `git pull origin master`

2 指定更新的分支

```
git pull forkName branchName
```

从 `forkName` 更新 `branchName`。具体来讲：

```
git pull <远程主机名> <远程分支名>:<本地分支名>
```

例如

```
git pull origin next:master
```

取回 `origin` 主机的 `next` 分支，并与 `master` 合并。上面命令表示，取回`origin/next`分支，再与当前分支合并。实质上，这等同于先做`git fetch`，再做`git merge`。

```
git fetch origin
```

```
git merge origin/master
```



在某些场合，Git会自动在本地分支与远程分支之间，建立一种追踪关系(tracking)。比如，在git clone的时候，所有本地分支默认与远程主机的同名分支，建立追踪关系，也就是说，本地的master分支自动”追踪” origin/master分支。

Git也允许手动建立追踪关系。

```
git branch --set-upstream master origin/next
```

上面命令指定master分支追踪origin/next分支。

如果当前分支与远程分支存在追踪关系，git pull就可以省略远程分支名。

```
git pull origin
```

上面命令表示，本地的当前分支自动与对应的origin主机”追踪分支”(remote-tracking branch)进行合并。

如果当前分支只有一个追踪分支，连远程主机名都可以省略。即

```
git pull
```

这与第一节的 git pull 不谋而合。