

Git submodule 使用技巧

emacsun

目录

1 简介	1
2 添加 submodule	1
3 删除 submodule	2
4 针对多个 submodule 的操作	2
5 参考	2

1 简介

在工程 A 中有时需要用到模块 b。而模块 b 是一个相对独立的模块，由别人维护，我们只需要定期更新模块 b 就可以了。如果把模块 b 也作为工程版本控制的一部分显然很麻烦，因为每次都从别人那里更新过来，再在当前工程下提交。git submodule 命令完美解决了这个问题。

2 添加 submodule

第一次接触 git submodule 是通过 Hugo 的一个主题 [docdock](#)。通常，在 Hugo 根目录下会有一个 theme 文件夹。这里放置了静态博客使用的主题，这些主题在 github 上有自己的仓库。而我的博客也有自己的仓库。一种使用主题的方法是把主题.zip 包下载下来，放到 hugo 根目录，同博客一起更新管理。但是这样就容易漏掉了主题所在的 github 上仓库的更新。并且主题的 github 仓



库更新时，我还是要下载.zip 包，并在我的仓库里提交这个主题的更新。为了方便 theme 中的主题同 github 上主题同步，可以选择使用 git submodule.

具体操作为：

1. 在 hugo 根目录下添加 submodule

```
$ git submodule add https://github.com/vjeantet/hugo-theme-docdock.git themes/docdock
```

这时会在根目录下生成一个 .gitmodules 文件。

1. submodule 有更新时，只需要

```
$ git submodule init
```

```
$ git submodule update
```

即可更新 submodule.

3 删除 submodule

删除 submodule 的步骤：

1. 删除 .gitmodules 文件。
2. 执行 `git rm -cached` 把子模块从 git 中删除。

4 针对多个 submodule 的操作

见[这里](#)

5 参考

1. [git submodule 的使用](#)
2. [Hugo 主题 docdock](#)