

# git仓库的创建与获取

成文时间：2020-09-17

参考地址：[GIT--fast-version-contro](#)

阅读软件：Typora (Catfish主题)

联系作者：[qiao\\_jinming@foxmail.com](mailto:qiao_jinming@foxmail.com)

文章作者：乔金明

## 1. 初始化仓库

- 本地获取git仓库有两种方式，一种是将本地目录转换为git仓库，另一种是从其他服务器克隆一个已经存在的git仓库
- 初始化本地的仓库，进入需要初始化的项目目录下，执行：

```
git init
```

- 执行完初始化命令后，那么这个目录下将会生成一个.git/隐藏目录，这个隐藏子目录包含git仓库中所有的必须文件，此时整个目录完成了初始化，但是目录内部还没有任何文件加入到git仓库
- .git/目录中包含哪些文件
  - description：描述文件，仅供GitWeb程序使用，这里不做过多介绍
  - config：配置文件包含项目特有的配置选项
  - info/：info目录包含全局排除文件，用来放置不希望记录在.gitignore文件中的模式
  - hooks/：钩子目录用来放置客户端或者服务端的钩子脚本，钩子指的是在特定的事件触发后被调用的脚本
  - HEAD：重要，记录文件指向目前的分支
  - index：重要，记录文件保存暂存区信息
  - objects/：重要，存储所有的数据内容
  - refs/：重要，存储指向数据的提交对象的指针，例如：分支、远程仓库、标签等
- 文件夹内存在文件时可以进行版本控制，空文件夹进行版本控制没有意义，通过以下可以开始对指定文件进行追踪

```
touch README.md
git add README.md
git commit -m 'commit'
```

- git初始化的其他常用操作

```
# 初始化时只打印错误和警告信息，其他信息不打印
git init -q|--quiet

# 初始化一个裸仓库，通常用于远端的中心仓库
git init --bare

# 使用自己命名的分支名字，如果不指定默认的分支名字为master
git init [-b <branch-name>|--initial-branch=<branch-name>]

# 创建一个指定项目位置的仓库
git init [directory]
```

- 创建一个基本仓库与裸仓库的区别
  - 创建一个基本仓库会在项目目录下生成一个.git/文件夹，而裸仓库创建的项目下直接就是.git/文件夹下的内容，是不包含工作数据的，一般裸仓库文件夹目录以.git结尾命名
  - 因为裸仓库内不包含工作区，所以不能够在这个目录下执行一般使用的git命令，另外正常创建的仓库会记录所有的引用指针（refs）更新
  - 在功能上的差异，也是根本差异，从裸仓库克隆下的仓库可以正常执行push仓库，但是一般初始化的仓库却不行，所以裸仓库一般作为远端中心仓库而存在的
  - 可以理解为先修改再提交，如果有仓库push了修改，此时远端仓库工作区不会改变，这样提交关系就本末倒置了，所以裸仓库可以接受push但是不能直接提交变更
- 为什么git需要一个远程的中心仓库
  - 在介绍git的优点时，其中之一就是git区别于集中式版本控制，这里远程中心仓库说法实际上并不冲突，因为各个开发者为了开发协作的方便，需要有一个总的控制中心，这样对于合并更新都更容易操作，当然如果愿意，各个开发者也可以在本地上架服务器，部署自己的项目供其他人克隆提交

## 2. 克隆仓库

- 克隆仓库支持的协议有ssh，git，http[s]，ftp[s]，本地协议

```
git clone xxx
```

- 克隆仓库的其他常用命令

```
# 项目克隆到本地后进行重命名  
git clone xxx [name]
```

- 各个协议优缺点
  - 本地协议
    - 优点：简单，并且可以直接使用现有的文件权限与网络，通过共享文件系统并设置好读写权限即可使用
    - 缺点：不方便从多个位置访问，数据没有压缩，速度较慢，所有人都有访问的权限，容易损坏仓库
  - http协议
    - 优点：使用比ssh协议更简单也可以想ssh一样提供授权与加密，使用广泛，一般的防火墙都支持此类端口数据
    - 缺点：部署麻烦，每次访问都需要账号密码（可以直接配置到git配置文件中，避免不断输入）
  - ssh协议
    - 优点：ssh协议一般服务器均支持，部署容易，相关管理工具也很多，传输数据需要进行加密和授权，需要进行数据压缩传输效率很高效
    - 缺点：ssh不支持匿名访问，使用者必须通过ssh访问主机，不利于开源项目
  - git协议
    - 优点：包含在git里的一个特殊守护进程，端口9418，是git使用的协议中最快的，与ssh具有相同的传输机制，但是省去了加密和授权的开销
    - 缺点：缺乏授权机制，所以一般不把git协议作为唯一手段，git协议是最难架设的，一般防火墙不会开放9418端口