

1. 为啥要用Git?

1. 在单人开发过程中需要进行版本管理
2. 在多人开发过程中不仅需要版本管理还需要多人协同控制

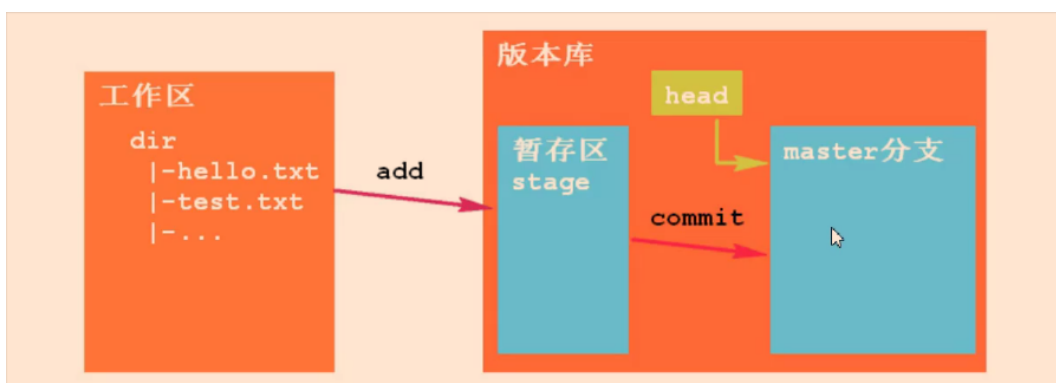
2. 介绍

1. Git是一个开源的分布式版本控制系统，用于敏捷高效地处理任何或大或小的项目。
2. Git的官网：<https://git-scm.com/>

3. 架构

1. 工作区：本地的项目目录。
2. 版本库：工作区中有一个隐藏的目录`.git`，这个目录不属于工作区，而是git的 版本库，是git管理的所有内容。
3. 暂存区：版本库中临时的存放区域，保存下一步要提交的文件。
4. 分支：版本库中包含若干分支，提交的文件存储在分支。

架构图



4. 仓库

1. 就是版本库加工作区

5. 本地仓库的基本命令

1. 创建仓库 `git init`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub
$ git init
Initialized empty Git repository in D:/git/GitHub/.git/
```

2. 查看仓库的状态 `git status`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        notes.txt

nothing added to commit but untracked files present (use "git add" to track)
```

3. 将文件从工作区保存在暂存区中 `git add filename`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git add notes.txt
```

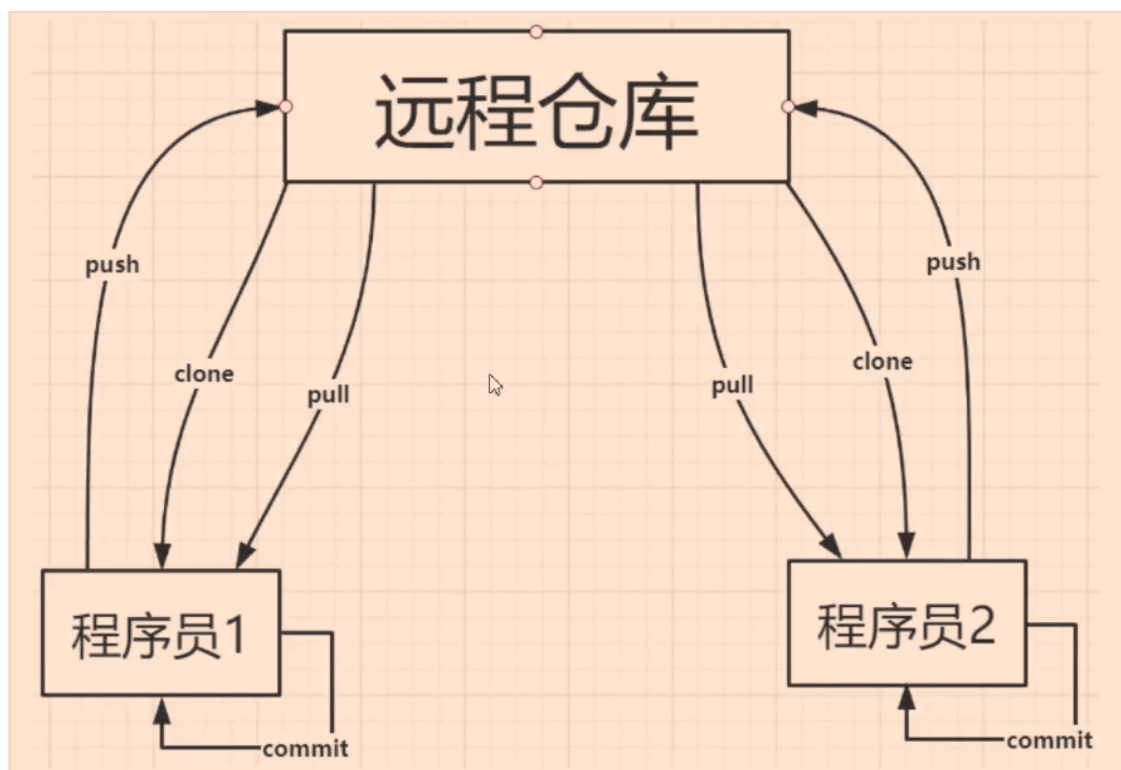
4. 提交文件形成正式版本 `git commit -m "这里写提交的描述信息"`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git commit -m "我的第一次提交"
[master (root-commit) 5292106] 我的第一次提交
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 notes.txt
```

6. 远程仓库

1. 当多人开发时需要共享代码、合并代码，此时需要一个 `远程仓库`。
2. 远程仓库的工作模式

远程仓库工作模式



3. 远程仓库选型：github、码云、公司内部的远程仓库等。

7. 远程仓库的基本命令

1. 关联远程仓库 `git remote add repositoryName url`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git remote add firstCommit https://github.com/Chen-Zhilin/hello_world.git
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$
```

2. 查看关联的地址 `git remote -v`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git remote -v
firstCommit      https://github.com/Chen-Zhilin/hello_world.git (fetch)
firstCommit      https://github.com/Chen-Zhilin/hello_world.git (push)
```

3. 将本地仓库提交到远程仓库 `git push repositoryName branchesName`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git push firstCommit master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 308 bytes | 308.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Chen-Zhilin/hello_world.git
d583793..050412d master -> master
```

4. 将远程仓库克隆到本地仓库 `git clone url`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git clone https://github.com/Chen-Zhilin/hello_world.git
Cloning into 'hello_world'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 6 (delta 0), reused 3 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 900 bytes | 1024 bytes/s, done.
```

5. 拉取远程仓库内容到本地 `git pull repositoryName branchesName`

```
Administrator@PC-20200418NUWM MINGW64 /d/git/GitHub (master)
$ git pull hello_world master
From hello_world
 * branch          master      -> FETCH_HEAD
Already up to date.
```