

# 极简教程：基于 git 控制 L<sup>A</sup>T<sub>E</sub>X 的 T<sub>E</sub>X 版本管理

Z. Fang

Chem. FDU

版本：1.00

更新：2020 年 8 月 14 日



# 1 Preface

本文的主要内容为如何基于 git 对  $\text{\TeX}$  进行版本控制。众所周知，`u1s1`<sup>1</sup>，`L $\text{\TeX}$`  十分好用，你甚至可以 2 个小时内完全掌握<sup>2</sup>。git 呢，就是 github 的那个 git，使用之前需要下载一个<sup>3</sup>（这不是废话吗）。行吧，不想听废话直接跳过这一部分好了。

如果只是怕写论文而不想建那么多个“最终版”的话，仅仅需要熟记以下几个命令：

- `$ git log`
- `$ git add`
- `$ git commit`
- `$ git tag`
- `$ git reset`

## 2 第 0 步，写你的 $\text{\TeX}$

这个就无所谓你用什么写啦，什么 `TexStudio`，`TexWork`，甚至 `txt` 写好再编译都一样的，快乐就行。

---

<sup>1</sup> “有一说一”

<sup>2</sup>大雾。指某本教程。

<sup>3</sup><https://git-scm.com/>

### 3 第一步，新建 git repository

在下载 git 后，右键空白处，“git bash here”，就可以进入命令行模式。新建的话直接输入

```
\$ git init %初始化git respiratory,i.e.,将当前目录定义为一个repository
```

**评论** 最好建一个文件夹，将写  $\text{\TeX}$  要用的所有资源放进去，然后直接在里面 Bash 就行啦。

### 4 第二步，将文件加入暂存区

将文件加入暂存区<sup>4</sup>的命令为：

```
\$ git add . %提交目录下的所有文件至暂存区
```

```
\$ git add 你的文件名.文件类型 %提交目录下的某文件至暂存区，e.g.: git add Hello.docx
```

### 5 第三步，暂存区文件提交至 repository

这一步需要的命令为：

```
\$ git commit -m "版本更新日志"
```

---

<sup>4</sup>不考虑 Remote 的话，git 分 workspace (你的  $\text{\TeX}$ )，index 和 repository

commit 之后就 vans<sup>5</sup>大吉了!

**注** 你还可以给你的文档加 tag:

```
\$ git tag %显示所有tag  
\$ git tag XXX %在当下的commit建一个tag  
\$ git tag XXX XXX %给指定commit建tag
```

## 6 回撤版本

写文章总会遇到不如意嘛，比如上个版本写的比较好..... 这个时候，你就需要回退版本了。使用

```
git reset --hard HEAD~ 回退一个版本（将版本退回上一个commit的状态）：  
git reset --hard HEAD^^ 回退两个版本：  
git reset --hard HEAD~100 回退100个版本：  
git reset --hard 版本号 回退到指定版本：
```

版本号可以通过 git log 查询。git log 就是你 git 的所有记录，你可以慢慢看这里面的日志。

## 7 致谢

谢谢你!

---

<sup>5</sup>即“万事”，玩谐音梗，我扣钱。

## 8 常见问题 FAQ

- 1). 有什么问题？  
你没有问题。

## 9 示例

略。或许此处应该有个示例吧。