

# Git常用命令

冯哲\*

西安电子科技大学计算机学院

2018年9月17日

## 目录

<b>1</b>	<b>GIT空间状态关系图示</b>	<b>4</b>
1.1	代码提交与工作流程图 . . . . .	4
1.2	五种状态撤销更改图 . . . . .	4
<b>2</b>	<b>创建版本库</b>	<b>5</b>
2.1	初始化版本库 . . . . .	5
2.2	添加文件到版本库（本地仓库） . . . . .	5
<b>3</b>	<b>版本控制</b>	<b>5</b>
3.1	<b>GIT RESET用法解析</b> . . . . .	5
3.2	版本回退 . . . . .	7
3.2.1	查看提交历史 . . . . .	7
3.2.2	版本回退 . . . . .	7
3.2.3	版本前置 . . . . .	7
3.3	撤销修改 . . . . .	7

---

\*电子邮件: 1194585271@qq.com

3.3.1	丢弃工作区的修改 . . . . .	7
3.3.2	丢弃已添加到暂存区的修改 . . . . .	7
3.4	删除文件 . . . . .	7
3.4.1	删除工作区文件 . . . . .	7
3.4.2	确认删除则提交版本库 . . . . .	7
3.4.3	误删则暂存区替换工作区版本 . . . . .	8
<b>4</b>	<b>远程仓库</b>	<b>8</b>
4.1	添加远程库 . . . . .	8
4.2	推送最新修改 . . . . .	8
4.3	从远程库克隆 . . . . .	8
<b>5</b>	<b>分支管理</b>	<b>8</b>
5.1	分支的管理与合并 . . . . .	8
5.1.1	查看分支 . . . . .	8
5.1.2	创建分支 . . . . .	9
5.1.3	切换分支 . . . . .	9
5.1.4	创建+切换分支 . . . . .	9
5.1.5	合并某分支到当前分支 . . . . .	9
5.1.6	删除分支 . . . . .	9
5.2	修复bug 临时存储 . . . . .	9
5.3	多人协作 . . . . .	9
5.3.1	查看远程库信息 . . . . .	9
5.3.2	本地推送分支 . . . . .	10
5.3.3	本地创建和远程分支对应的分支 . . . . .	10
5.3.4	建立本地分支和远程分支的关联 . . . . .	10

5.3.5	远程抓取分支 . . . . .	10
5.4	REBASE . . . . .	10
<b>6</b>	<b>标签管理</b>	<b>11</b>
6.1	创建标签 . . . . .	11
6.2	查看所有标签 . . . . .	11
6.3	操作标签 . . . . .	11
6.3.1	推送本地标签到远程 . . . . .	11
6.3.2	删除本地标签 . . . . .	11
6.3.3	删除远程标签 . . . . .	11

1 GIT空间状态关系图示

Working Area. Stage. Local Repository. Remote Repository
Origin. Modified or Untracked. Staged. Committed. Pushed

1.1 代码提交与工作流程图

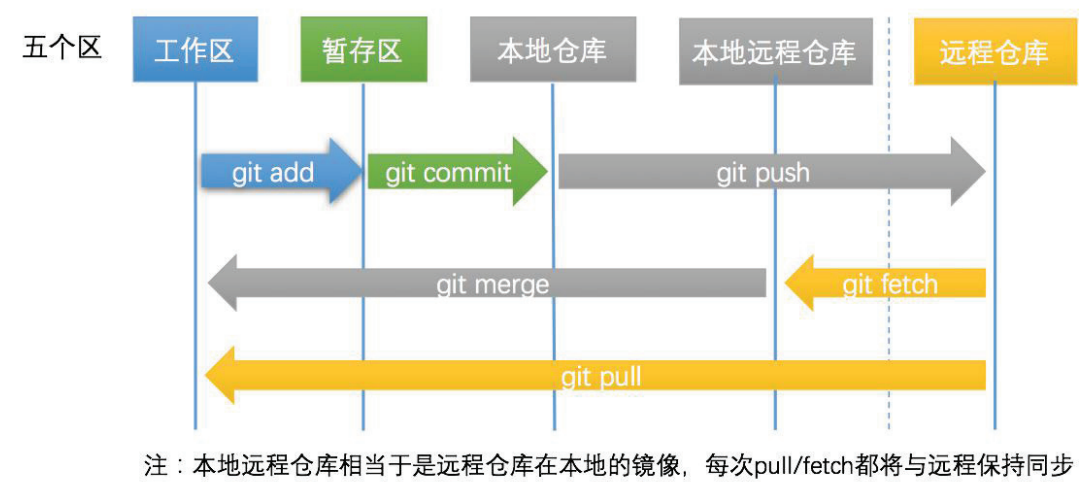


图 1: 代码提交与工作流程图

1.2 五种状态撤销更改图

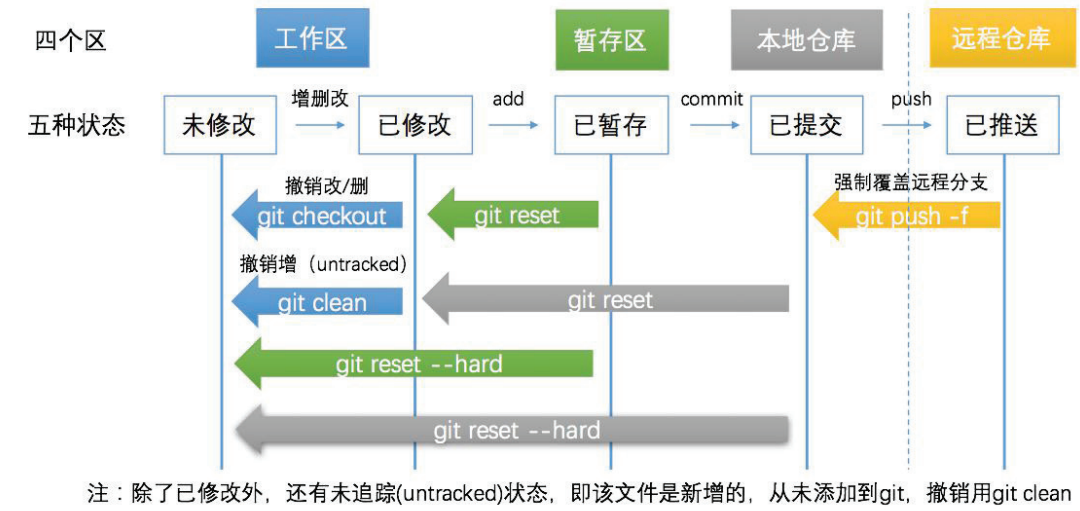


图 2: 五种状态撤销更改图

## 2 创建版本库

### 2.1 初始化版本库

```
$git init
```

### 2.2 添加文件到版本库（本地仓库）

```
$git add <files> or git add . (completely add)  
$git commit -m <message>
```

## 3 版本控制

### 3.1 GIT RESET用法解析

```
$git reset [--soft|--mixed|--hard] <commit>
```

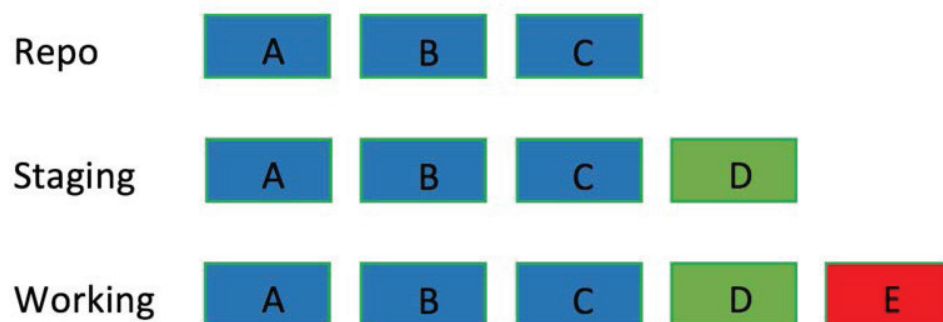


图 3: 原始状态

```
$git reset [--soft] <B commit>
```

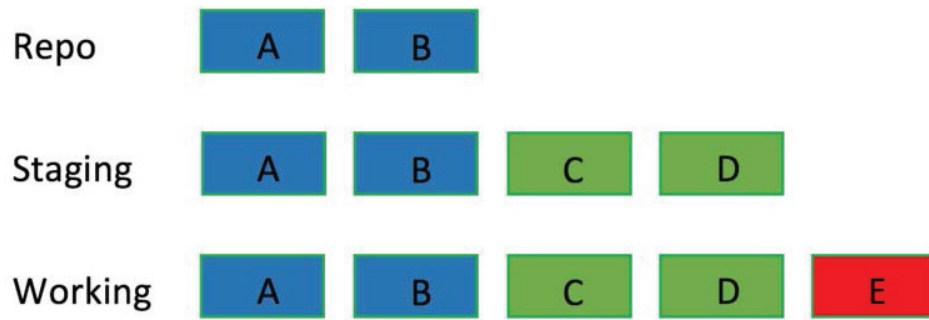


图 4: After git reset --soft B

```
$git reset [--mixed] <B commit>
```

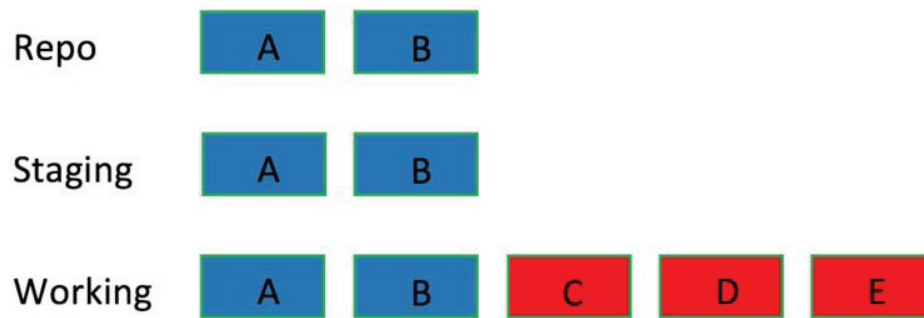


图 5: After git reset --mixed B

```
$git reset [--hard] <B commit>
```



图 6: After git reset --hard B

## 3.2 版本回退

### 3.2.1 查看提交历史

```
$git log --pretty=oneline --graph --all --decorate
```

### 3.2.2 版本回退

```
$git reset --hard commit_id
```

### 3.2.3 版本前置

```
$git reflog  
$git reset --hard commit_id
```

## 3.3 撤销修改

### 3.3.1 丢弃工作区的修改

```
$git checkout <files>
```

### 3.3.2 丢弃已添加到暂存区的修改

```
$git reset HEAD <files>  
$git checkout <files>
```

## 3.4 删除文件

### 3.4.1 删除工作区文件

```
$git rm <files>
```

### 3.4.2 确认删除则提交版本库

```
$git add .  
$git commit -m <message>
```

### 3.4.3 误删则暂存区替换工作区版本

```
$git checkout <files>
```

## 4 远程仓库

### 4.1 添加远程库

```
$git remote add origin <URL>
```

### 4.2 推送最新修改

```
$git push -u origin master (at first)  
$git push
```

### 4.3 从远程库克隆

```
$git clone <URL>
```

## 5 分支管理

### 5.1 分支的管理与合并

#### 5.1.1 查看分支

```
$git branch
```



### 5.1.2 创建分支

```
$git branch <name>
```

### 5.1.3 切换分支

```
$git checkout <name>
```

### 5.1.4 创建+切换分支

```
$git checkout -b <name>
```

### 5.1.5 合并某分支到当前分支

```
$git merge <name>
```

### 5.1.6 删除分支

```
$git branch -d <name>
```

## 5.2 修复bug 临时存储

```
$git stash  
$git checkout <bug>  
$git stash pop
```

## 5.3 多人协作

### 5.3.1 查看远程库信息

```
$git remote -v
```

### 5.3.2 本地推送分支

```
$git push origin branch-name (if fail,git pull)
```

### 5.3.3 本地创建和远程分支对应的分支

```
$git checkout -b branch-name origin/branch-name
```

### 5.3.4 建立本地分支和远程分支的关联

```
$git branch --set-upstream branch-name origin/branch-name
```

### 5.3.5 远程抓取分支

```
$git pull
```

## 5.4 REBASE

### git rebase

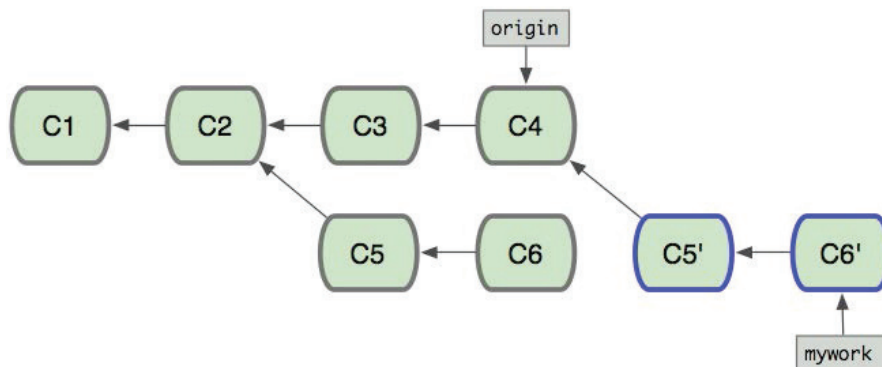


图 7: rebase

```
$git rebase <branch name>  
$fix conflict  
$git add <files>  
$git rebase --continue or git rebase --abort or git rebase --skip
```

## 6 标签管理

### 6.1 创建标签

```
$git tag <tagname> or  
$git tag -a <tagname> -m <message>
```

### 6.2 查看所有标签

```
$git tag
```

### 6.3 操作标签

#### 6.3.1 推送本地标签到远程

```
$git push origin <tagname> (one tag)  
$git push origin --tags (all tags)
```

#### 6.3.2 删除本地标签

```
$git tag -d <tagname>
```

#### 6.3.3 删除远程标签

```
$git push origin :refs/tags/<tagname>
```