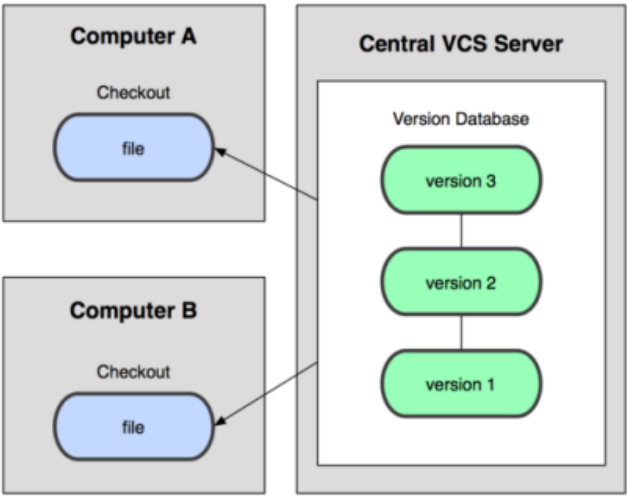


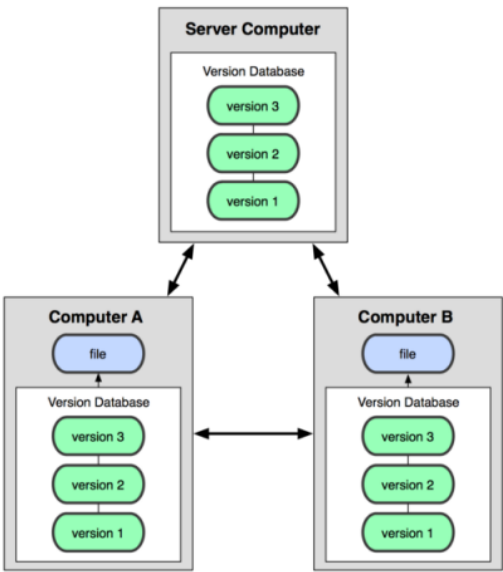
**集中式的版本控制系统**（Centralized Version Control Systems CVCS），有一个单一的中央管理服务器。



致命缺点：

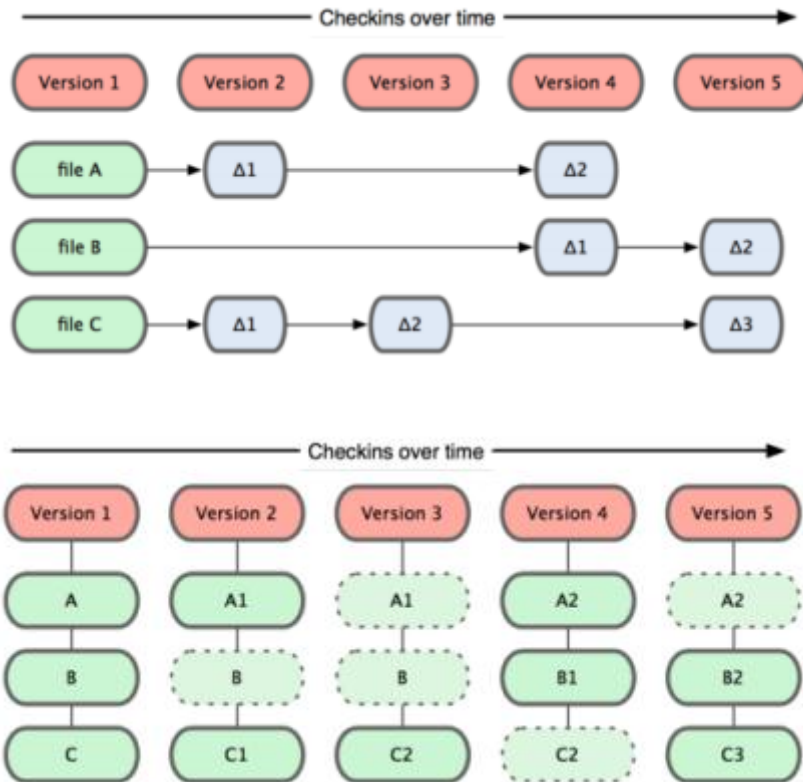
中央服务器的单点故障，导致版本数据丢失。

**分布式版本控制系统**（Distributed Version Control System）



Git 是所谓的分布式版本管理系统，并没有集中在某一台服务器上。所以，每个人都可以在本机进行版本控制。然而做项目，总要把大家的成果综合起来，放在一个安全的地方，交给某个人管理。

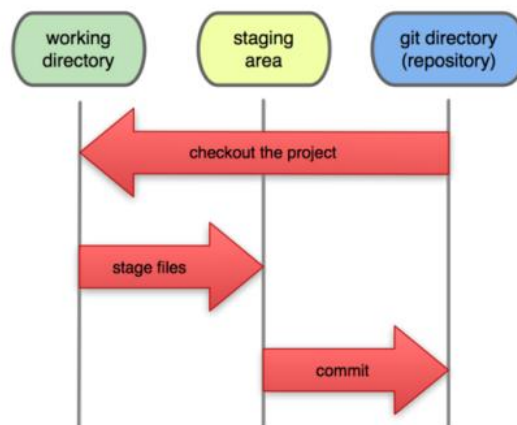
二者对比：



## Focus on Git

三种状态：已提交（committed），已修改（modified）和已暂存（staged）

### Local Operations



## Git 安装

```
$ apt-get install git-core
```

## Git 配置

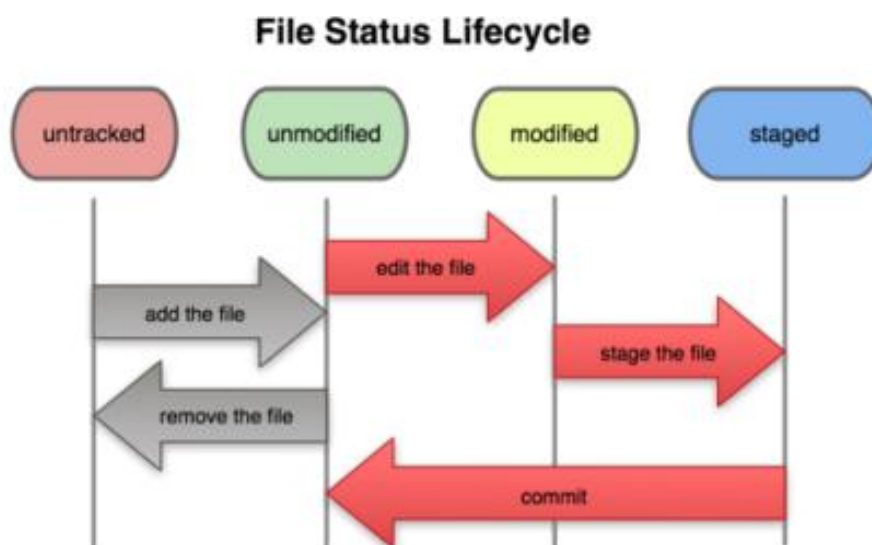
|        |   |
|--------|---|
| 用户信息   | <code>git config --global user.name/user.email</code> |
| 文本编译器  | <code>git config --global core.editor emacs</code>    |
| 差异分析工具 | <code>git config --global merge.tool vimdiff</code>   |
| 查看配置信息 | <code>git config --list</code>                        |
| 获取帮助   | <code>git &lt;verb&gt;--help</code>                   |

## Git 仓库

`git init` 初始化一本地仓库

`git clone git://git.com/username/test.git` 从远程仓库 clone。

Git 文件状态



状态转换

```
git add .
```

```
git commit -m '****'
```

.ignore 文件

能够实现在提交的时候，自动过滤掉满足条件的文件。

查看已经暂存和未暂存文件的差异

```
git diff
```

删除文件

```
git rm → git commit
```

查看提交历史

```
git log
```

撤销操作

修改上一次的提交(有些文件没有提交或提交信息错误)

```
git commit --amend
```

取消已暂存的文件

当 add 之后，就会将其纳入暂存。撤销已暂存，`git reset HEAD file_name`

取消对文件的修改

`git checkout -- fine_name`

## 远程仓库

查看远程仓库

`git remote [-v]`

`git remote show remote-name`

添加远程仓库

`git remote add myremote git://github.com/user/test.git`

拉取远程仓库

`git fetch/pull myremote`

推送到远程仓库

`git push [remote-name] [branch-name]`

远程仓库重命名

`git remote rename my-remote your-remote\`

删除远程仓库

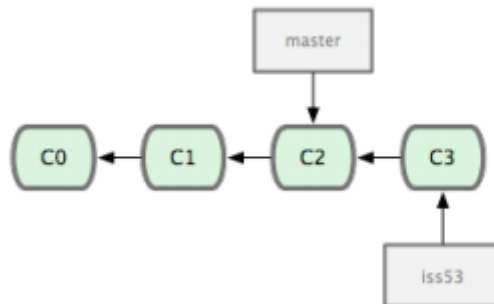
`git remote rm your-remote`

## 分支

创建分支 `git branch branch-name`

切换分支 `git checkout branch-name`

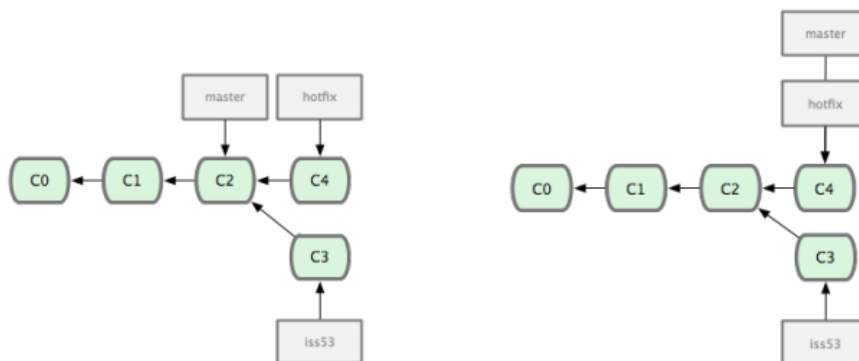
创建并切换 `git checkout -b branch-name`



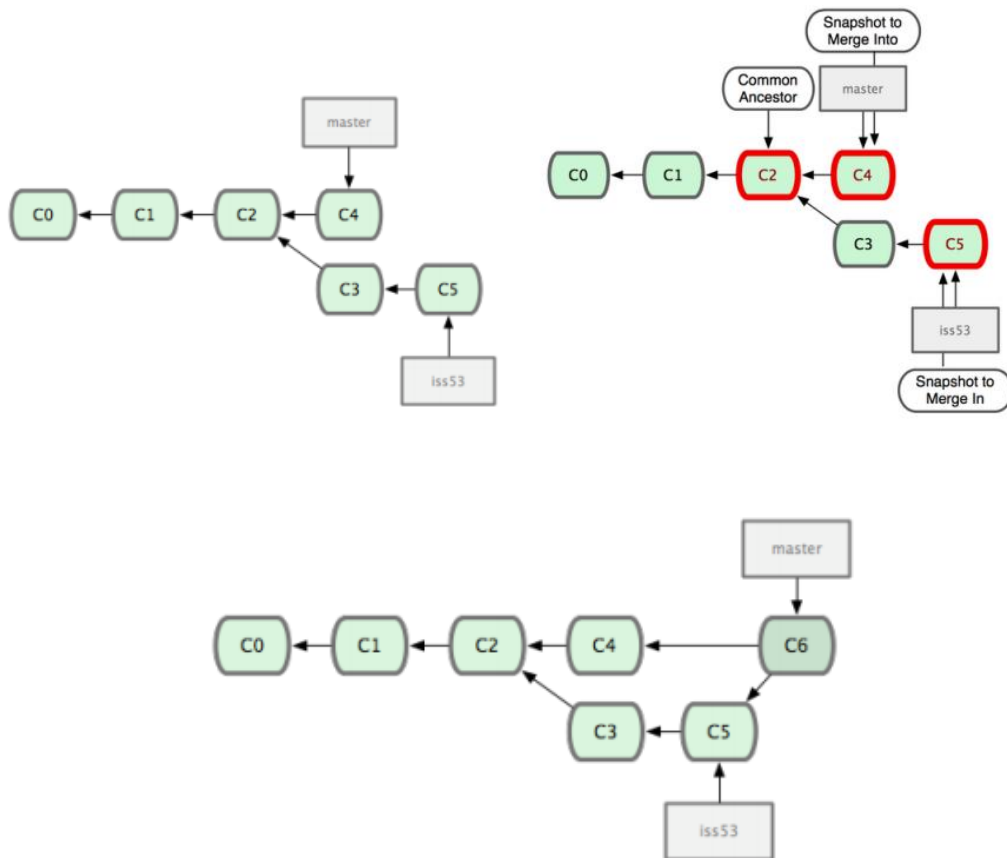
合并分支 将 feature 分支合并到 dev 上

`git checkout dev`

`git merge feature`



删除分支: `git branch -d branch-name`



分支冲突:

```

<<<<<<< HEAD:index.html
<div id="footer">contact : email.support@github.com</div>
=====
<div id="footer">
  please contact us at support@github.com
</div>
>>>>>> iss53:index.html
  
```

查看当前可用分支 `git branch [-v]`

详细使用参见【Pro Git】

[www.Linuxide.com](http://www.Linuxide.com)

THE EXPERT'S VOICE® IN SOFTWARE DEVELOPMENT

# Pro Git

*Everything you need to know about  
the Git distributed source control tool*



Linux公社 ( [www.LinuxDC.com](http://www.LinuxDC.com) ) 是专业的Linux系统门户网站, 实时发布最新Linux资讯。