集中式版本控制 代表： svn

需要去联网去服务器看代码

分布式版本控制 ：git

每个人都拥有全部的代码，

不会因为服务器损坏或网络问题，造成不能工作的情况

git和svn的区别

svn是集中式版本控制系统，版本库是集中放在中央处理器的，而工作的时候，用的都是自己的电脑，所以首先要从中央处理器得到最新版本，然后工作，完成工作后，需要把自己做完的活推送到中央服务器。集中式版本控制系统必须联网才能工作，对网络带宽要求较高

git是分布式版本控制系统，没有中央服务器，每个人的电脑就是一个完整的版本库，工作的时候不需要联网，因为版本都在自己的电脑上。协同的方法是这样的：比如说自己在电脑上改了文件A，其他人也在电脑上改了文件A，这是，你们两之间只需要把各自的修改推送给对方，就可以互相看到对方的修改了。git可以直接看到更新了哪些代码和文件

**git是目前世界上最先进的分布式版本控制系统**

**常用liunx命令**

cd .. 上一级目录

cd 切换目录

pwd 显示当前所在目录

clear 清屏

ls 列出当前文件夹下的所以文件及目录

touch index.js 新建一个文件

rm index.js 移除文件

mkdir test 新建目录

rm -r 删除要给文件夹

mv 目标对象 目标文件夹 移动文件

reset 重新加载终端

history 历史命令

exit 退出

help 帮助

git config -l 查看所以配置

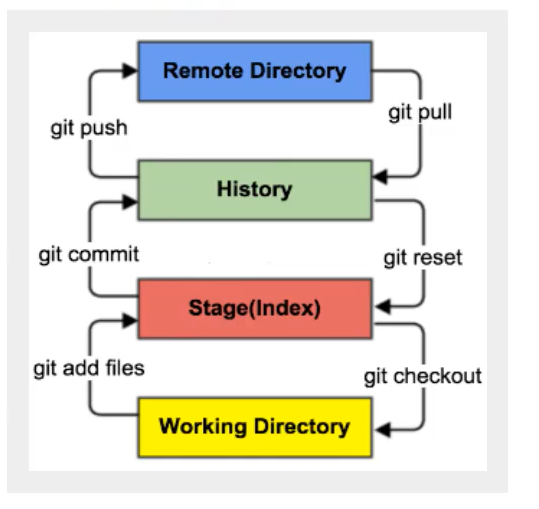
git config --system --list查看系统配置

git config --global --list 查看本地配置

配置用户（必须配置）

git config --golbal user.name "wangshengjian"

git config --global user.email 2628570396@qq.com



两边命令是相反的

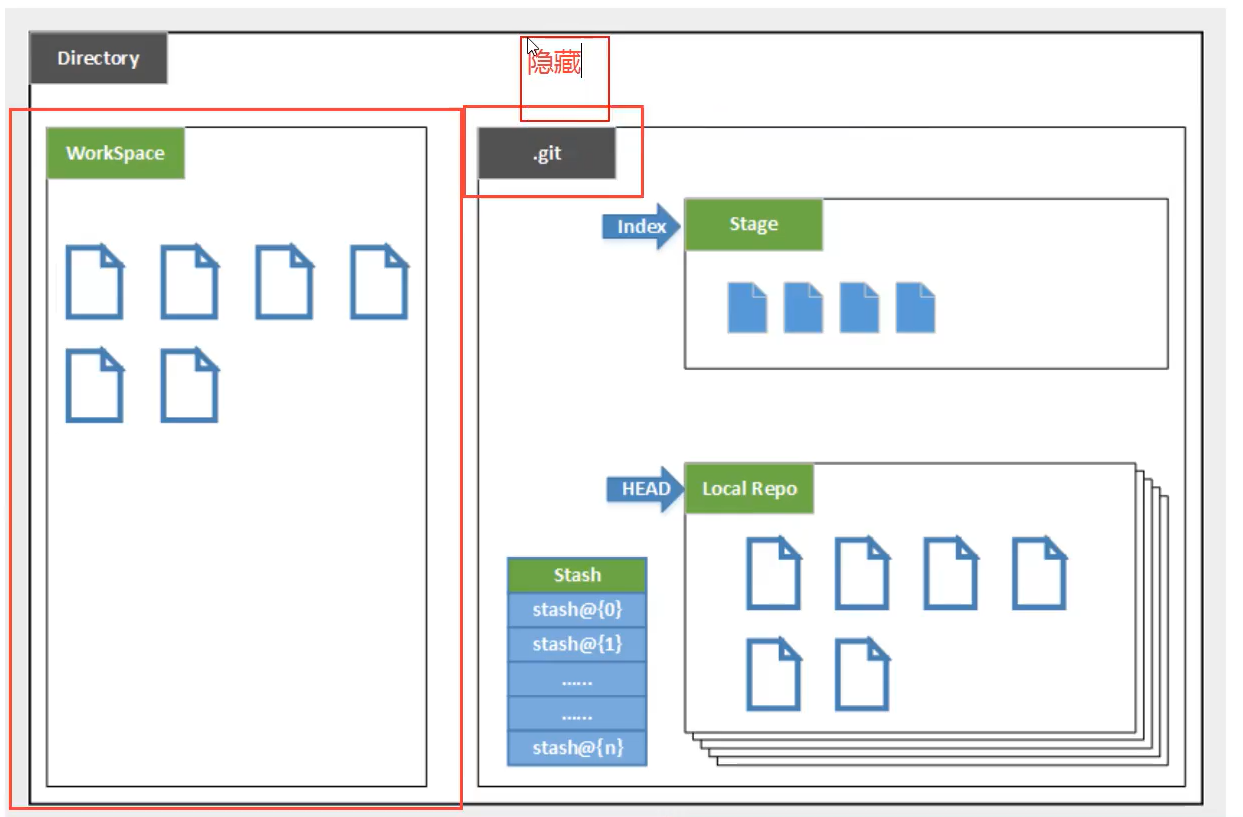
项目目录就是工作区

Workspace：工作区，就是你平时存放项目代码的地方

Index/Stage：暂存区，用于临时存放你的改动，事实上它只是一个文件，保存即将提交到文件列表的信息

Repository：仓库区（或本地仓库），就是安全存放数据的位置，这里面有你提交到所有版本的数据。其中HEAD指向最先放入仓库的版本

Remote：远程仓库，托管代码的服务器，可以简单的认为是你项目组中的一台电脑用于远程数据交换



git的工作流程一般是这样的：

1.在工作目录中添加、修改文件；

2.将需要进行版本管理的文件放入暂存区域；（git add）

3.将暂存区域的文件提交的git仓库。（git commit）

因此，git管理的文件有三种状态：已修改（modified），以暂存（staged），已提交（committed）

本地搭建仓库：git init

网上克隆仓库： git clone [url]

添加所有文件到暂存区：git add .

git status

提交暂存区的内容到本地仓库 -m是消息内容

git commit -m "消息内容"

在主目录下创建.gitignore文件

\*.txt 忽略所有 .txt结尾的文件，这样的话上传就不会选中

!lib.txt 表示lib.txt将不被忽略

如果文件的左前门是一个路径分隔符(/)，表示要忽略的文件在此目录下，而子目录中的文件不忽略

如果名称的后面是一个路径分隔符(/)，表示要忽略的是此目录下盖名称的子目录，而非文件（默认文件或目录都忽略）

.ssh文件夹设置公钥

ssh-keygen -t rsa