# 初识git

## 什么是git

Git 是一个开源的分布式版本控制系统，可以有效、高速的处理从很小到非常大的项目版本管理。 Git 是 Linus Torvalds 为了帮助管理 Linux 内核开发而开发的一个开放源码的版本控制软件。

## 分布式和集中式区别

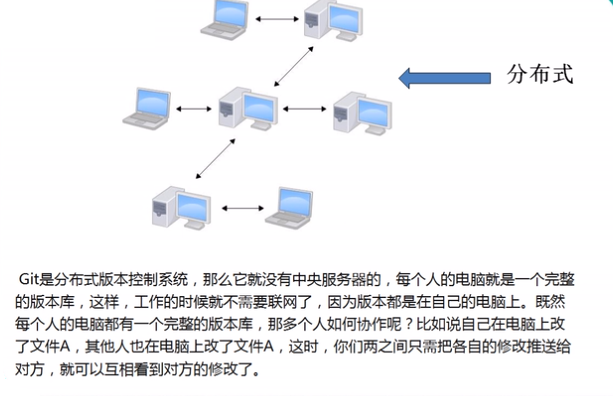
分布式相比于集中式的最大区别在于开发者可以提交到本地，每个开发者通过克隆（git clone），在本地机器上拷贝一个完整的 Git 仓库。

从一般开发者的角度来看，git 有以下功能：

从服务器上克隆完整的 Git 仓库（包括代码和版本信息）到单机上、 在自己的机器上根据不同的开发目的，创建分支，修改代码、 在单机上自己创建的分支上提交代码、 在单机上合并分支、把服务器上最新版的代码 fetch 下来，然后跟自己的主分支合并等。

优点：

适合分布式开发，强调个体。公共服务器压力和数据量都不会太大。速度快、灵活。任意两个开发者之间可以很容易的解决冲突。离线工作。



## 好处

### 快速，开源分布式版本控制系统，无中心仓库

### 源码备份

### 版本管理

### 协同工作

### 冲突解决

### 分支管理

### 里程碑管理

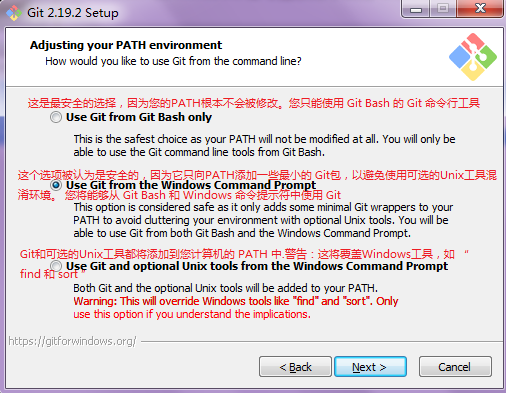
### 和github关联协同

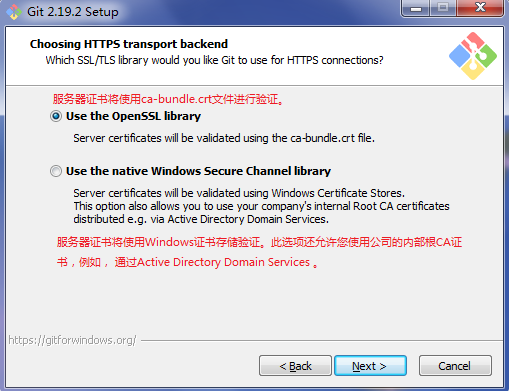
# git下载和安装

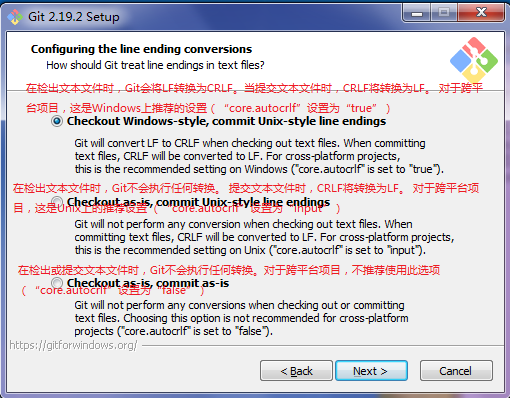
## git下载

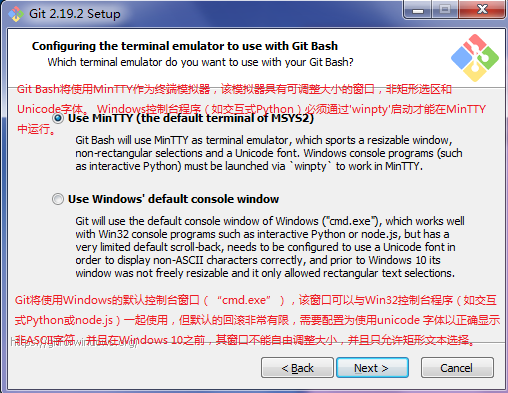
https://www.git-scm.com/download/win

## git安装(部分图解说明)











# git命令

## 初始化配置

git config --global user.name "grady"

git config --global user.email [grady\_lee@163.com](mailto:grady_lee@163.com)

## 新建仓库，查看状态

git init:新建本地仓库

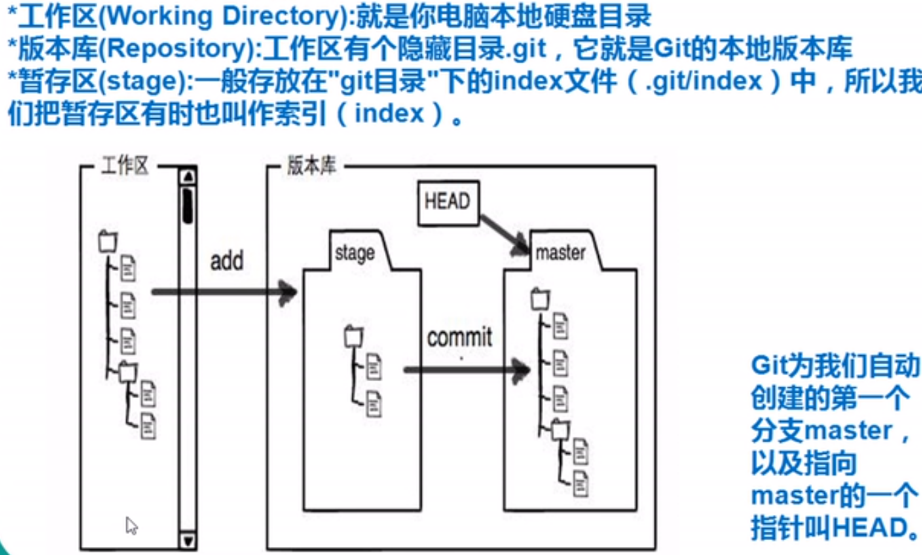
git status:检测状态

## 添加暂存区和本地仓库

git add file:添加文件到仓库缓存区

git commit -m "create":提交到仓库并添加消息

git diff a.txt 查看工作空间和仓库区别



## 查看日志记录

git log a.txt 查看文件日志

git log --pretty=oneline a.txt 显示简要日志

git reflog a.txt :查看版本日志

## 版本，回退和回撤

git reset --hard HEAD 查看当前的版本信息

git reset --hard HEAD^ 版本回退一个(回退多少步 ^^)

git reset --head HEAD~3:回退三步

git reset --hard 9c26b40:移动指针到指定操作

git checkout -- a.txt:版本回撤(未add和commit)

## 文件删除

git rm -f b.txt

git commit -m "delete"

## 分支

查看分支: git branch

创建分支: git branch dev

切换分支: git checkout dev

合并代码: git merge dev

主分支和从分支冲突: 人工干预

删除分支: git branch -d dev

新建并切换: git checkout -b test

# github

## 什么是github

gitHub 是一个面向开源及私有软件项目的托管平台，因为只支持 git 作为唯一的版本库格式进行托管，故名gitHub。

gitHub 于 2008 年 4 月 10 日正式上线，除了 git 代码仓库托管及基本的 Web 管理界面以外，还提供了订阅、讨论组、文本渲染、在线文件编辑器、协作图谱（报表）、代码片段分享（Gist）等功能。目前，其注册用户已经超过 350 万，托管版本数量也是非常之多，其中不乏知名开源项目 Rubyon Rails、 jQuery、 python 等。

作为开源代码库以及版本控制系统， Github 拥有超过 900 万开发者用户。随着越来越多的应用程序转移到了云上， Github 已经成为了管理软件开发以及发现已有代码的首选方法。

如前所述，作为一个分布式的版本控制系统，在 Git 中并不存在主库这样的概念，每一

份复制出的库都可以独立使用，任何两个库之间的不一致之处都可以进行合并。

在 GitHub，用户可以十分轻易地找到海量的开源代码

## 注册github，创建ssh key,添加并测试连通

ssh-keygen -t rsa -C liguotao\_1@163.com

~/.ssh生成目录

github中配置

测试:ssh -T [git@github.com](mailto:git@github.com)

## 本地新建，推送远程

git remote add origin https://github.com/liguotao1/test.git

git push -u origin master:推送

git pull origin master:更新

## 远程新建,本地克隆

git clone <https://github.com/liguotao1/test2.git>

## fork(自行测试)

# tortoiseGit安装与使用

## 下载

<https://tortoisegit.org/download/>

## 安装

## 操作

添加提交

查看版本

回退版本

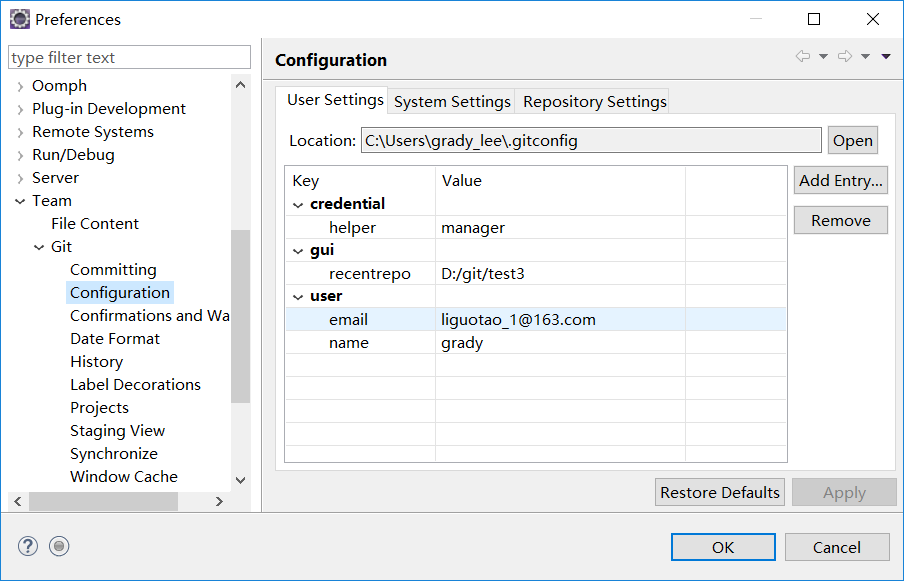
编辑冲突

提交github

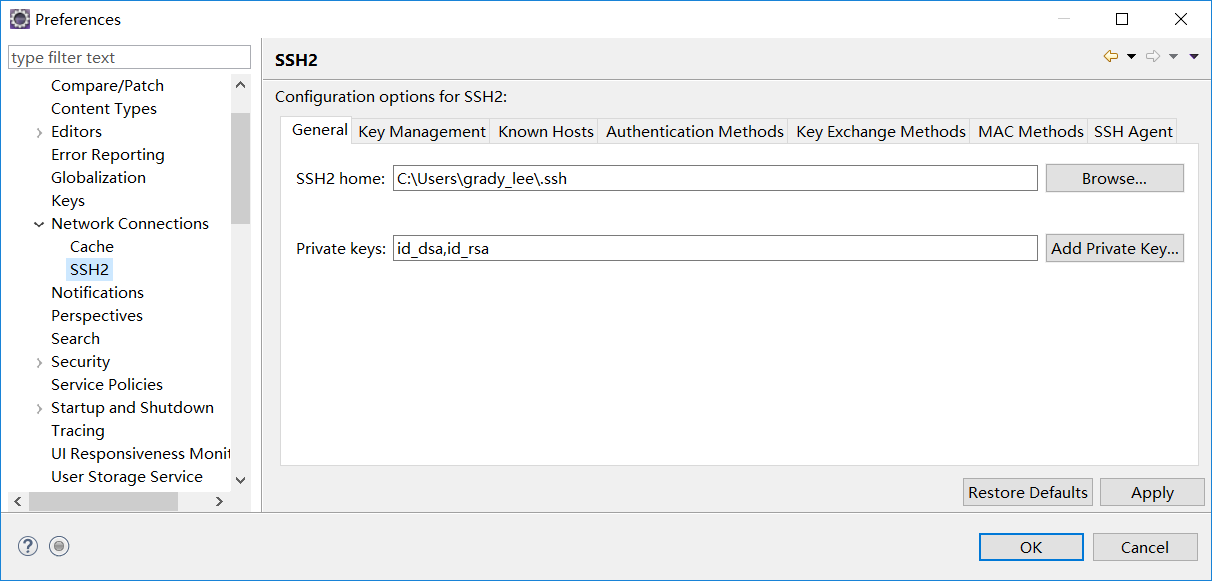
Clone github

# Egit使用

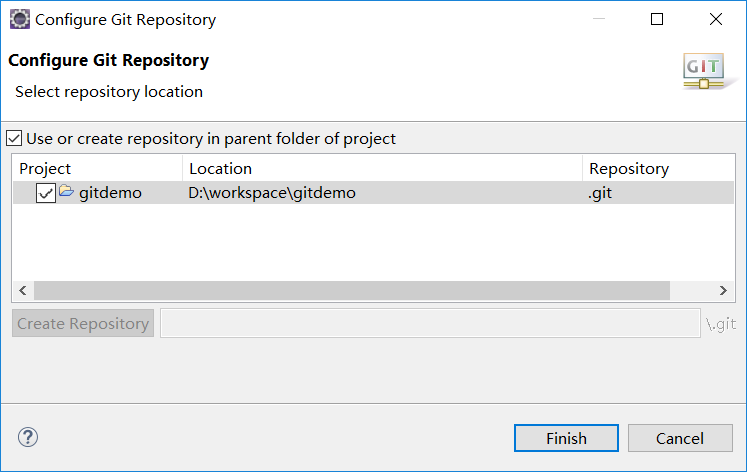
1. 配置



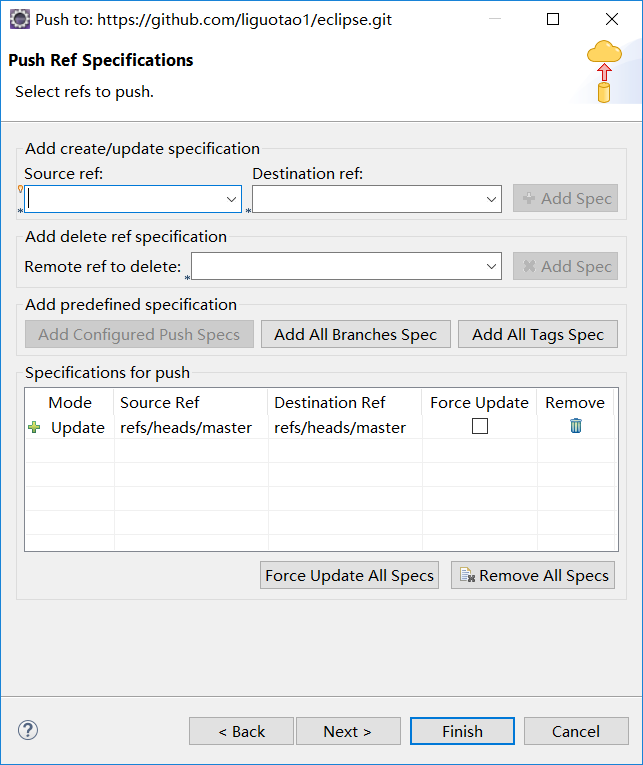
1. 查看ssh2



1. 创建仓库



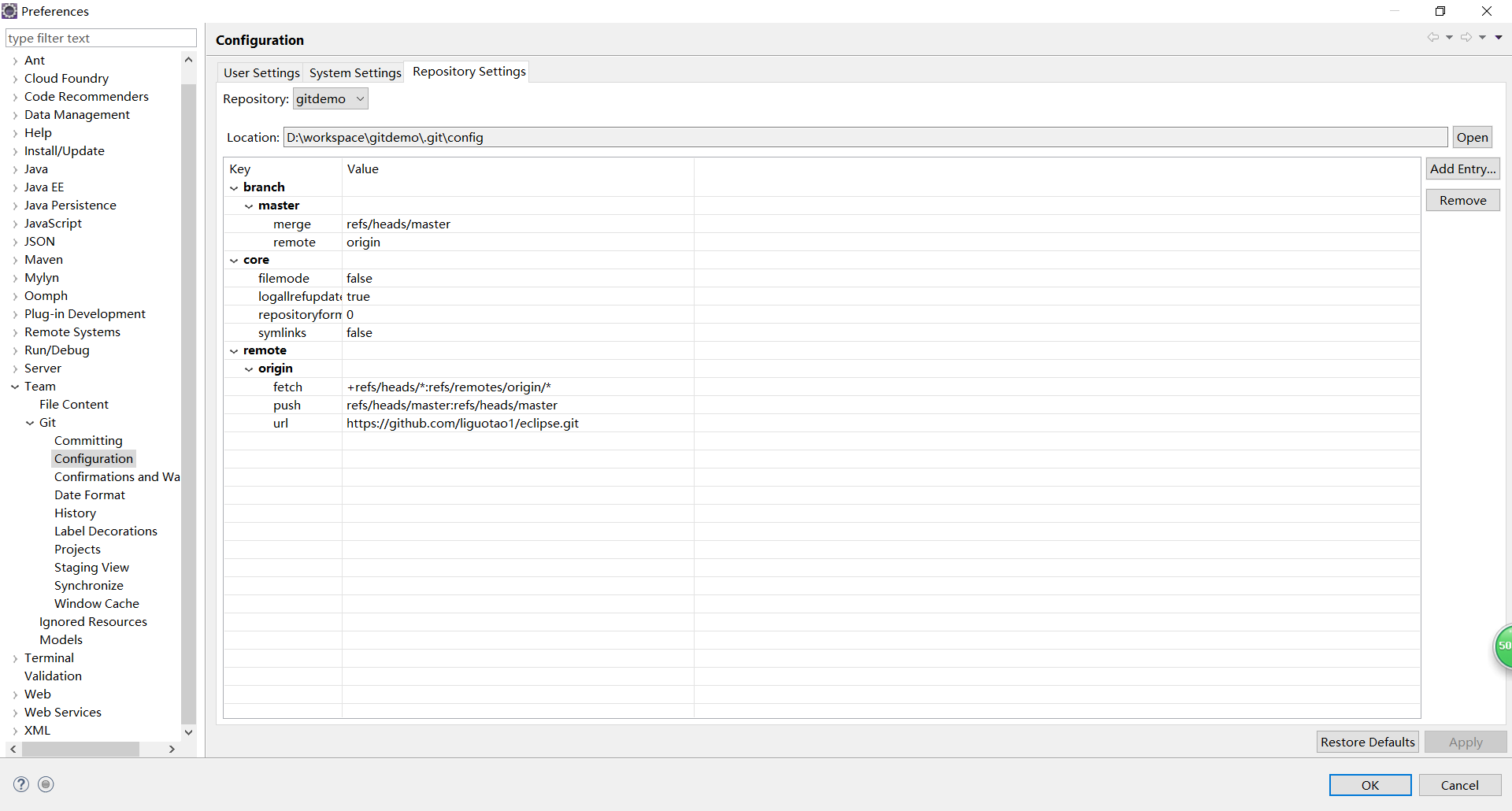
1. 添加到暂存区
2. 提交仓库
3. 与远程库交互



1. Pull从远程仓库

出错

添加git config



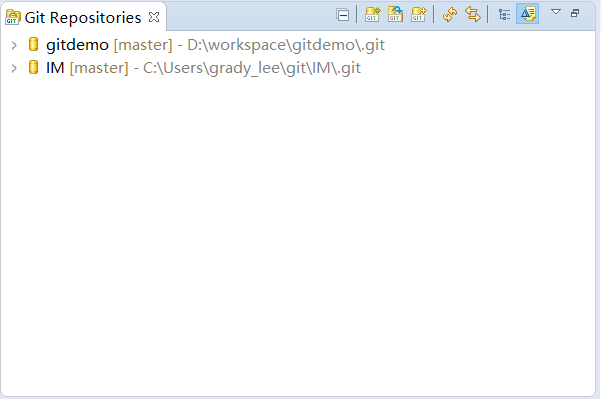
|  |
| --- |
| [remote "origin"]  fetch=+refs/heads/\*:refs/remotes/origin/\*  url=https://github.com/liguotao1/eclipse.git  push=refs/heads/master:refs/heads/master  [branch "master"]  remote=origin  merge=refs/heads/master |

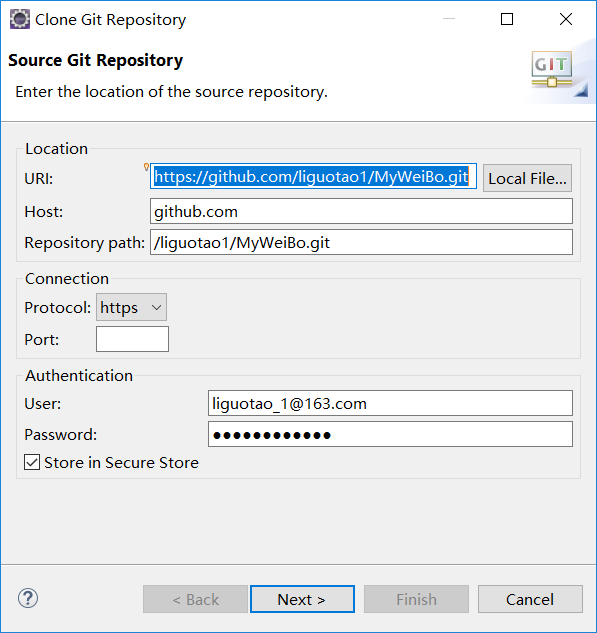
1. 同步

Sync workspace

Push/pull

1. Clone





Import🡪git

1. 冲突

服务器更新

本地:Add 🡪 commit 🡪push

解决:

Sync workspace

* Pull 🡪手动处理
* Add –>commit🡪push