# SVN简介

## 1.1 SVN介绍

SVN的全称是Subversion，即版本控制系统，它是很流行的一个开放源代码的版本控制系统。作为一个开源的版本控制系统，Subversion管理着随时间改变的数据，这些数据放置在一个中央资料档案库（Repository）中，它会记住每一次文件的变动，这样就可以把档案恢复到旧的版本，或是浏览文件的变动历史。Subversion是一个通用的系统，可用来管理任何类型的文件，其中包括程序源码。

SVN采用客户端/服务器体系，项目的各种版本都存储在服务器上，程序开发人员首先将从服务器上获得一份项目的最新版本，并将其复制到本机，然后在此基础上，每个开发人员可以在自己的客户端进行独立的开发工作，并且可以随时将新代码提交给服务器。当然也可以通过更新操作获取服务器上的最新代码，从而保持与其他开发者所使用版本的一致性。

SVN的客户端有两类，一类是基于Web的WebSVN等，另一类是以Tortoise SVN为代表的客户端软件。前者需要Web服务器的支持，后者需要用户在本地安装客户端。两种都有免费的开源软件供使用。SVN存储版本数据也两种方式：BDB（一种事务安全型存储系统）和FSFS（一种不需要数据库的存储系统）。

## 1.2 SVN特性

1. 集中式管理，客户端只需要同步提交即可，使用方便，操作简单，很容易就可以上手
2. 在服务端统一控制好访问权限，利用代码的安全管理
3. 所有的代码以服务端为准，代码一致性高
4. 支持二进制文件
5. 跨平台使用

基于以上特性，SVN在普通公司的开发团队中应用十分普遍。

## 1.3 SVN基本概念

1. 配置库/版本库（Repository）

* SVN的核心是配置库，储存了所有的数据，配置库按照文件树形式储存数据（包括文件和目录），任意数量的客户端可以连接到配置库，读写这些文件。通过写数据，别人可以看到这些数据；通过读数据，可以看到别人的修改。
* SVN会记录配置库中的每一次更改，不仅针对文件也包括目录本身，包括增加、删除和重新组织文件和目录。

1. 检出（Checkout）：将服务器端文件检出到本地系统作为工作副本。
2. 更新（update）：获取服务器端改动内容并将其更新到本地系统的过程。更新可以针对一个文件、几个选中的文件或者整个文件目录。

注：在本地修改文件时，要先更新该文件以保证是在该文件最新版本基础上所做修改

1. 添加（add）：申请向服务器端添加文件（操作完成后服务器端尚未添加）
2. 删除（delete）：申请删除服务器具体文件（操作完成后服务器端尚未删除）
3. 提交（commit）：将申请执行的操作更新到服务器
4. 检查更新（Check for modifications）：显示本地系统对文件所做的修改有哪些还没有提交。（他人对文件的改动、增加文件或目录、删除文件或目录、移动文件或目录等都可以通过此项功能来查询）
5. 撤销更改（Revert）：在修改了某些文件后（文件未上传到配置库），将本地系统上的文件恢复到修改前的状态。
6. 锁定和解锁（Get lock and Release lock）：当项目需要时可以在本地系统中将检出的内容进行锁定，锁定后他人将无法修改，解锁后他人才能继续对文件进行修改。
7. 获取历史文件（Show log）：显示该文件或者该目录执行的操作记录，包括操作执行人、执行修改的时间和日期、执行的操作等。

## 1.4 SVN下载

1. 下载地址：

<http://subversion.apache.org/packages.html>

1. 客户端：建议使用Tortoise SVN

<https://tortoisesvn.net/downloads.html>

1. 服务器端：建议使用VisualSVN server

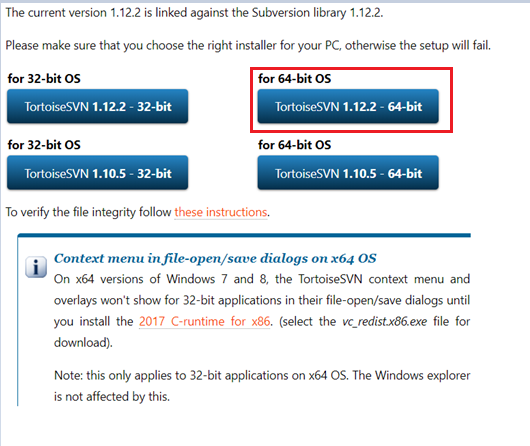
<https://www.visualsvn.com/downloads/>

# SVN安装

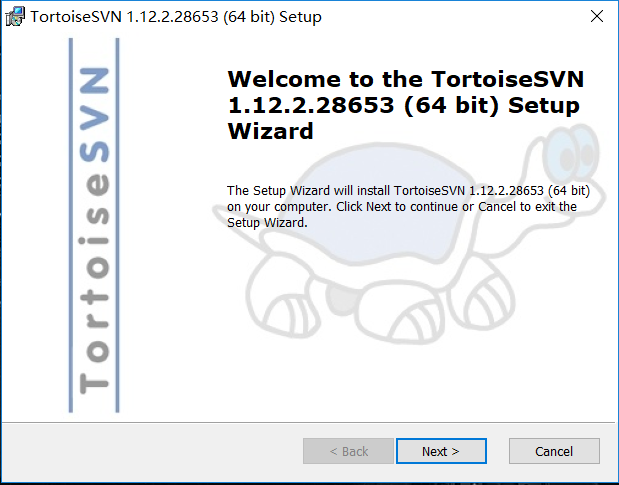
## 2.1 SVN客户端安装

以Windows系统安装Tortoise SVN为例。

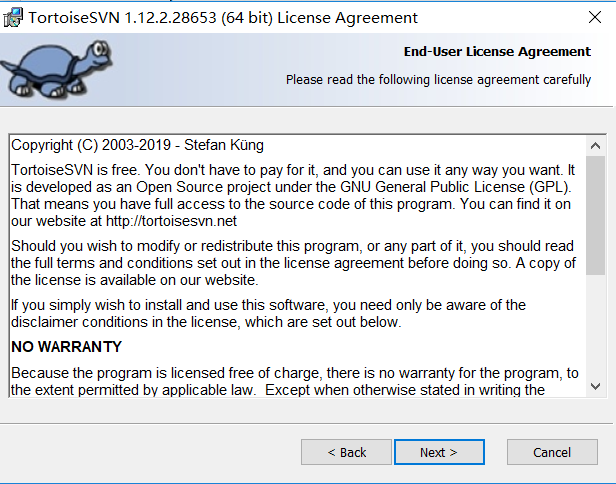
***Step 1:***根据自身操作系统选择合适版本客户端进行下载



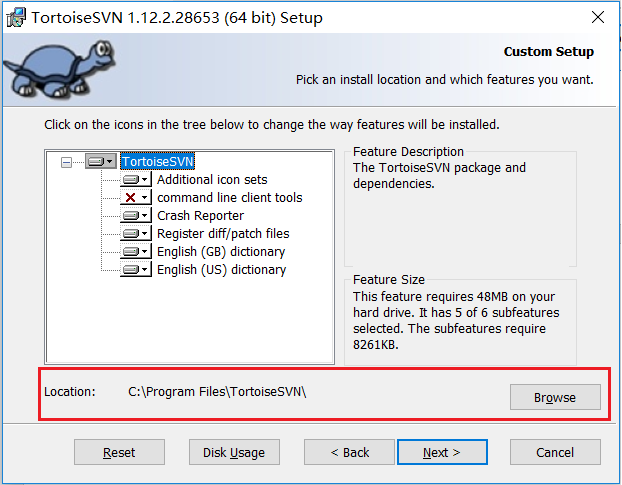
***Step 2:*直接进行安装**



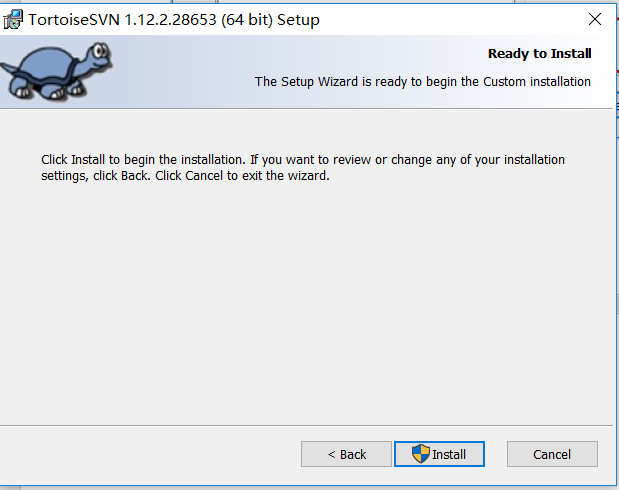
**点击下一步**



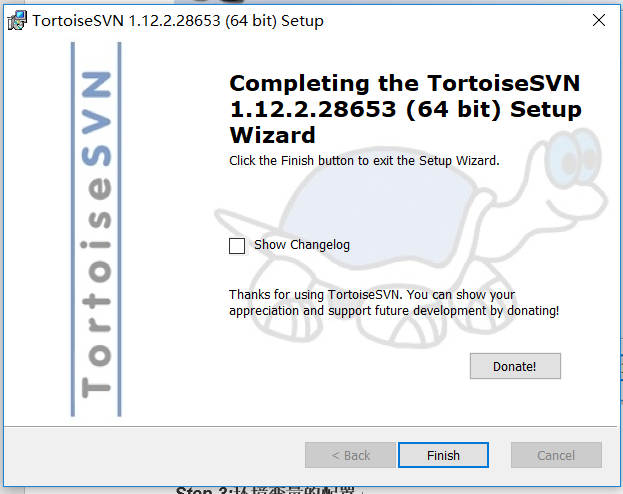
**选择安装路径，点击下一步**



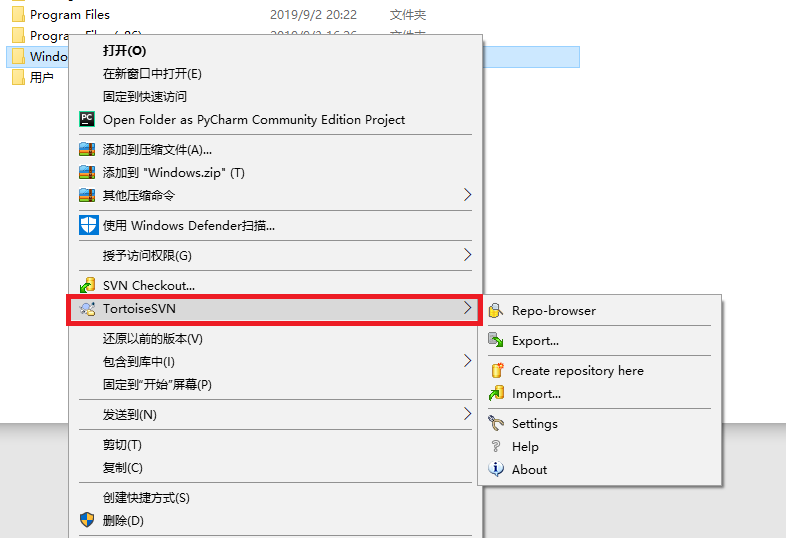
**开始安装**



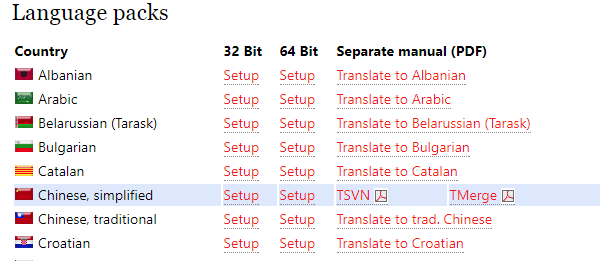
**完成安装**



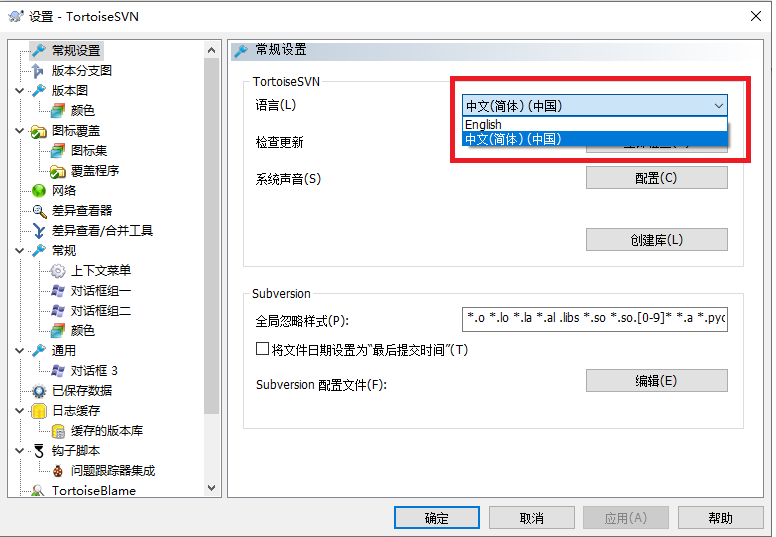
***Step 3:*验证右键菜单中出现TortoiseSVN选项即表示安装成功（如果安装完成后右键菜单中没有该选项，需重启电脑重新尝试）**



***Step 4:*可选功能，在客户端下载界面下拉找到语言包模块，下载并安装中文汉化包，先安装客户端，再安装汉化包**



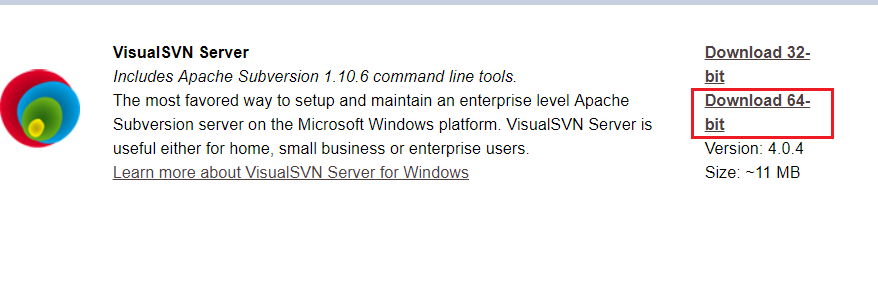
在开始菜单中选择Settings，，在常规设置中将语言选择为中文（简体），应用并确定后关闭Settings，重新打开即为中文模式



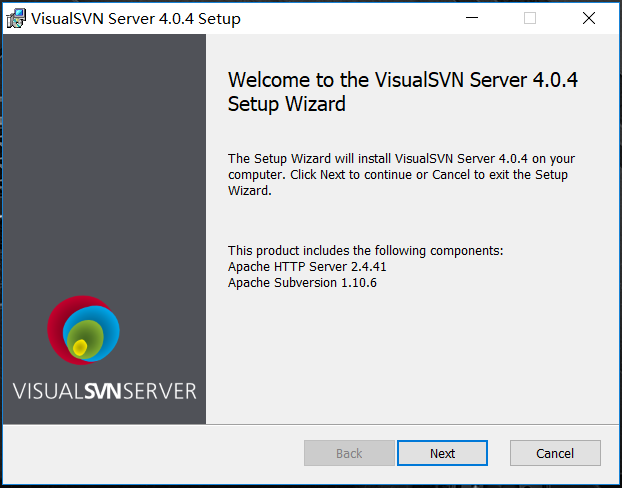
## 2.1 SVN服务器端安装

以Windows系统安装VisualSVN server为例。

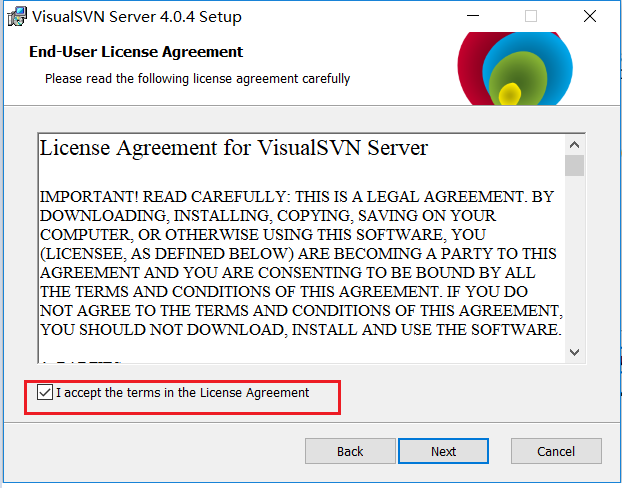
***Step 1:***根据自身操作系统选择合适版本服务器端进行下载



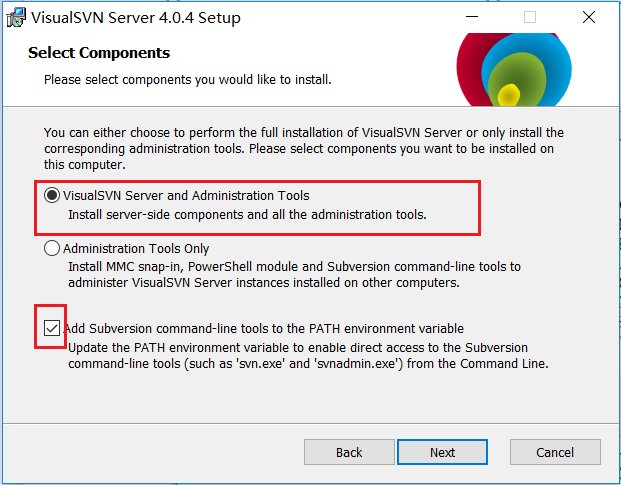
***Step 2:***直接进行安装



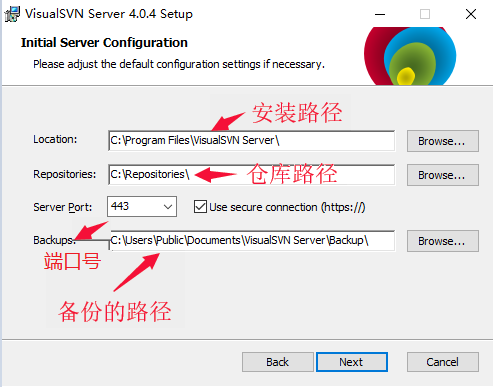
接受许可协议，点击下一步



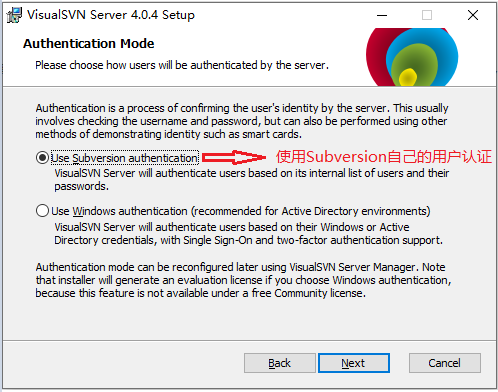
选择安装服务器和管理工具，点击下一步



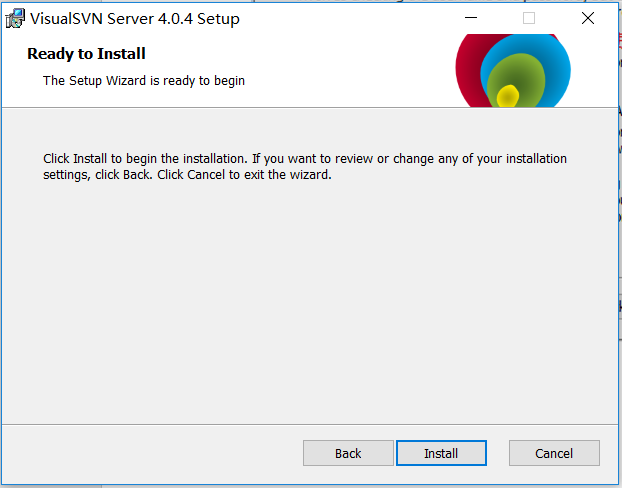
自定义安装路径、仓库路径和备份路径，点击下一步



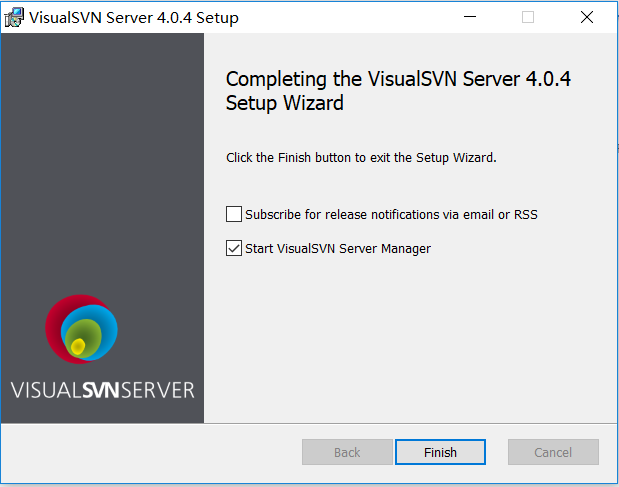
选择SVN自己认证工具，点击下一步



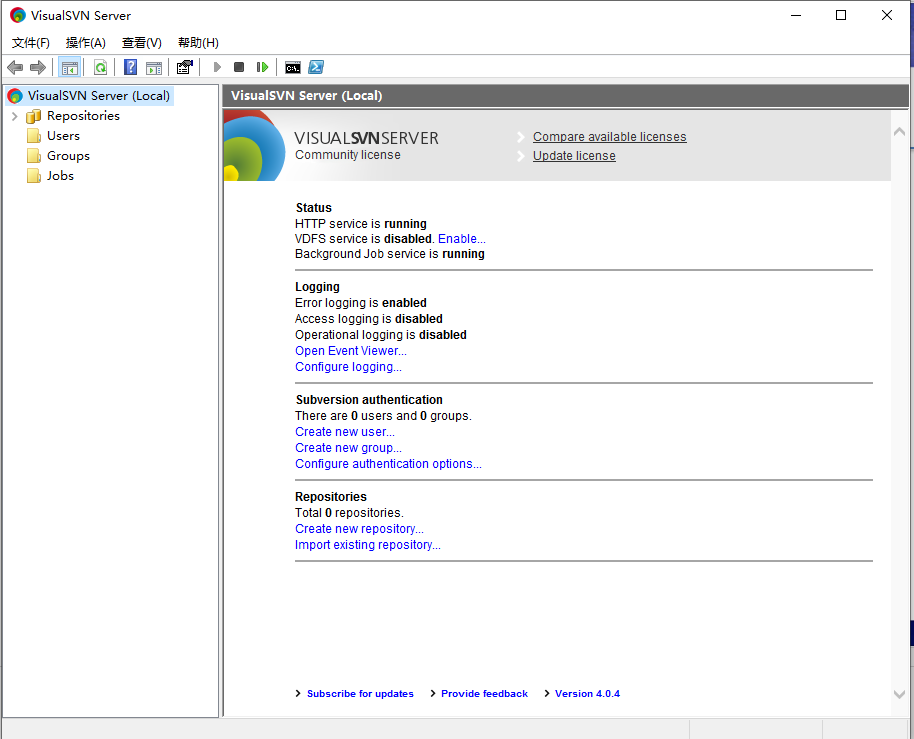
点击开始安装



完成安装



***Step 3:***完成安装后打开VisualSVN Server Manager，出现服务器管理菜单即表示安装成功

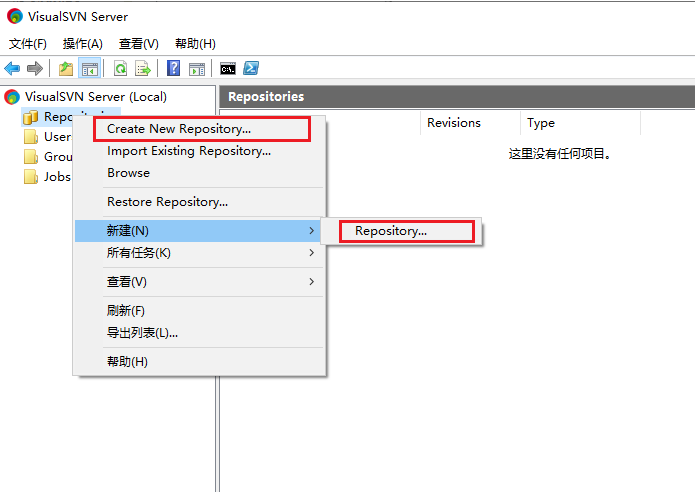


# 服务器配置

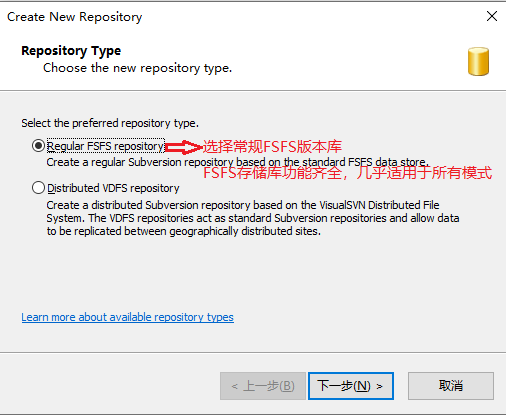
## 3.1 版本库配置

***Step 1:*创建版本库。**

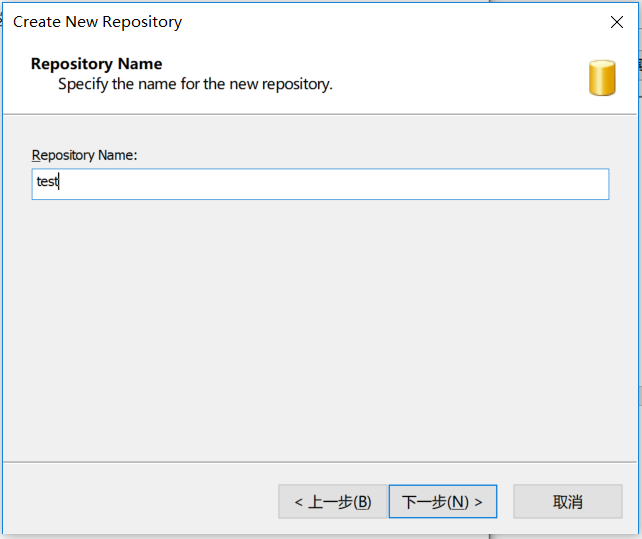
启动VisualSVN Server Manager，在Repositories上右键-选择Create New Repository…或者选择新建-Repository…



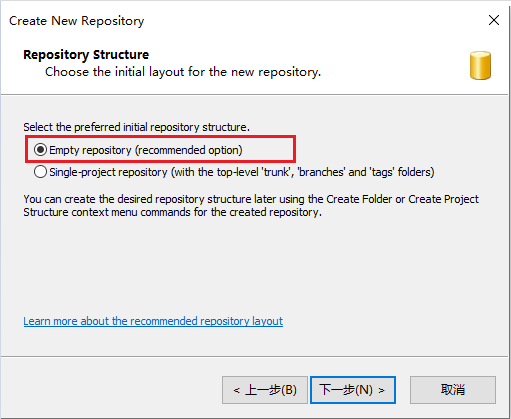
选择FSFS版本库，点击下一步



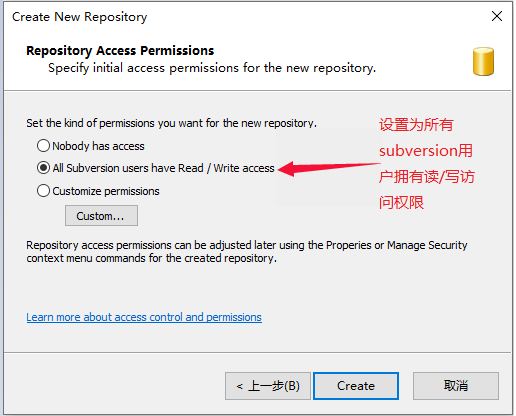
输入版本库名称，点击下一步



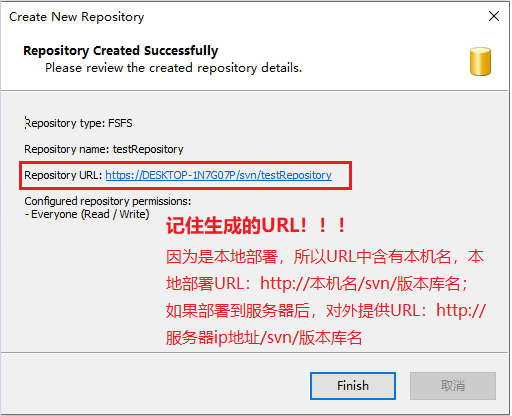
选择空版本库，点击下一步



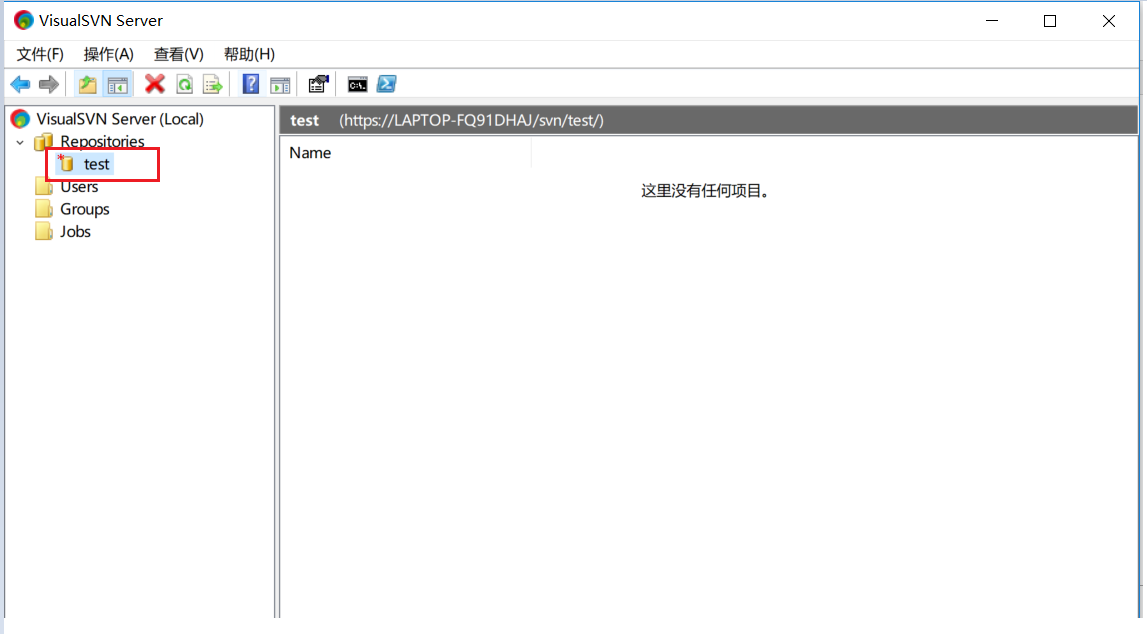
选择所有用户均具有读/写权限，点击创建



创建成功，点击完成

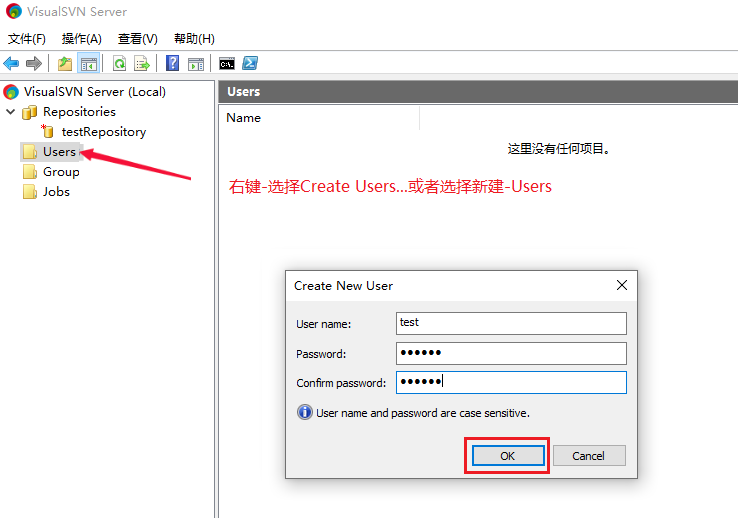


服务器已有创建好的版本库

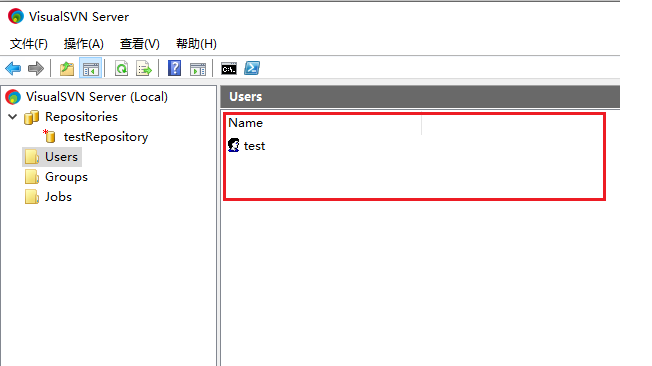


***Step 2:*创建用户。**

启动VisualSVN Server Manager，在Users上右键-选择Create New User…或者选择新建-User…，输入用户名密码，点击OK。

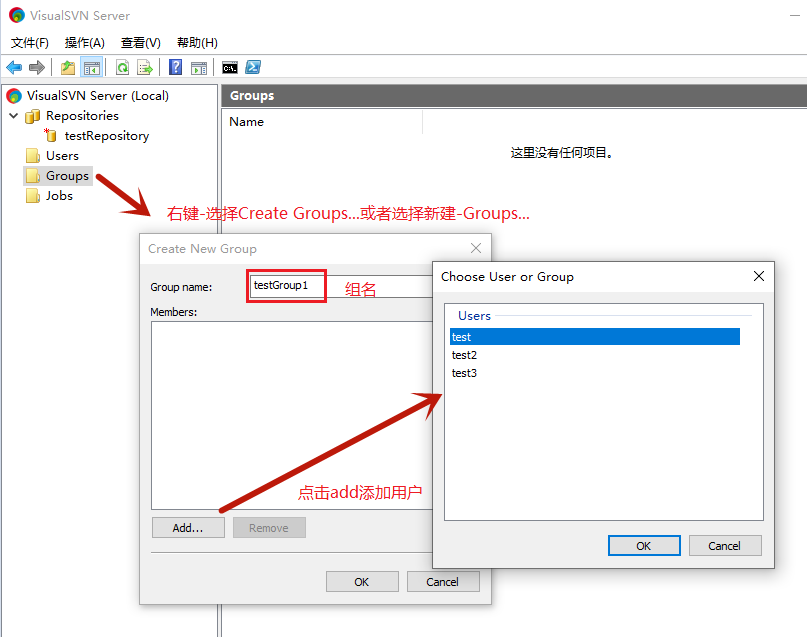


用户创建完成后，可在Users菜单栏显示用户名

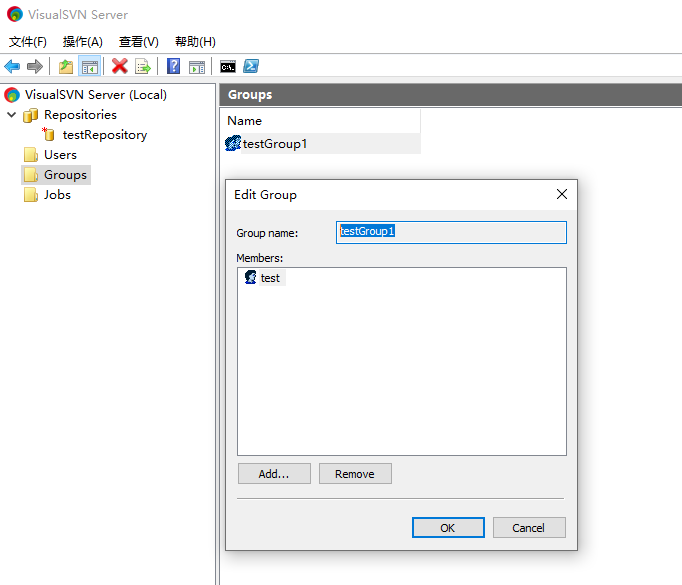


***Step 3:*创建组。**

启动VisualSVN Server Manager，在Groups上右键-选择Create New Group…或者选择新建-Group…，输入组名，为组添加成员，点击OK。



组创建完成后，可在Groups菜单栏显示组名，双击可编辑



## 3.2 权限配置

### 3.2.1权限配置规则

权限设置原则应遵守：

•从根目录到子目录；

•从最宽泛权限到最精细权限；

•从只读权限到读写权限。

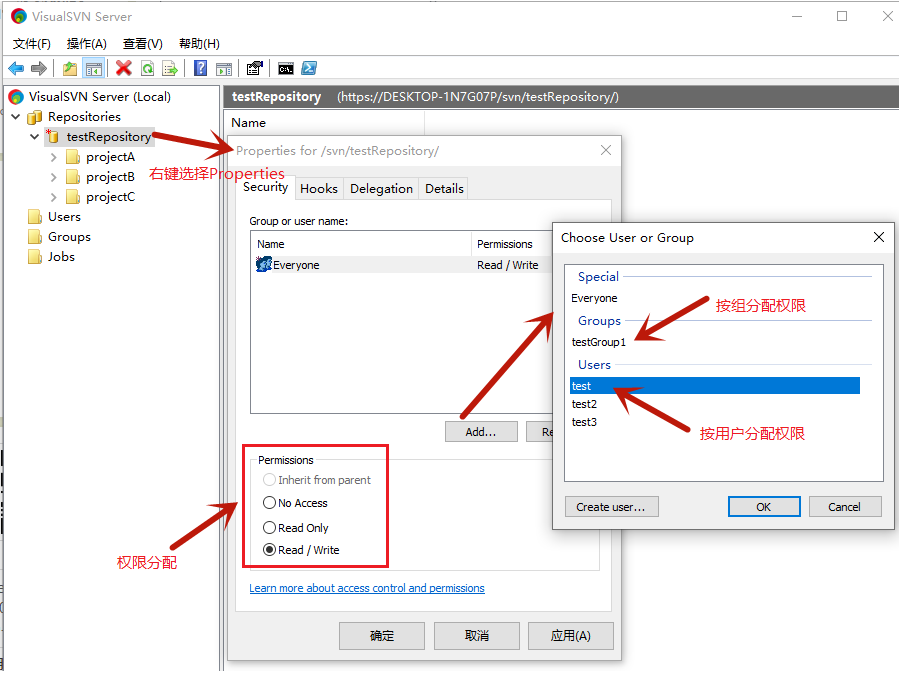
如未按照该原则设置权限，用户权限将按如下规则获取：

1. 在路径上配置过权限的用户，无论在其父目录上有无权限，都对该路径有权限；
2. 同一条路径多次配置，会合并所有配置内容，如果有对相同用户的权限配置，则最后的配置生效；
3. 如果某路径的配置中包含\*=r，那么所有用户的权限最低是r（即使有配置过用户权限为空，如：user=），如果有配置\*=rw，则所有用户的权限为rw；
4. 如果一条路径的配置内容中没有\*=（或者\*=r,\*=rw），则该路径会继承父目录的权限；
5. 如果一条路径和其父目录对一个用户有不同的权限配置，则访问路径自身的配置生效。

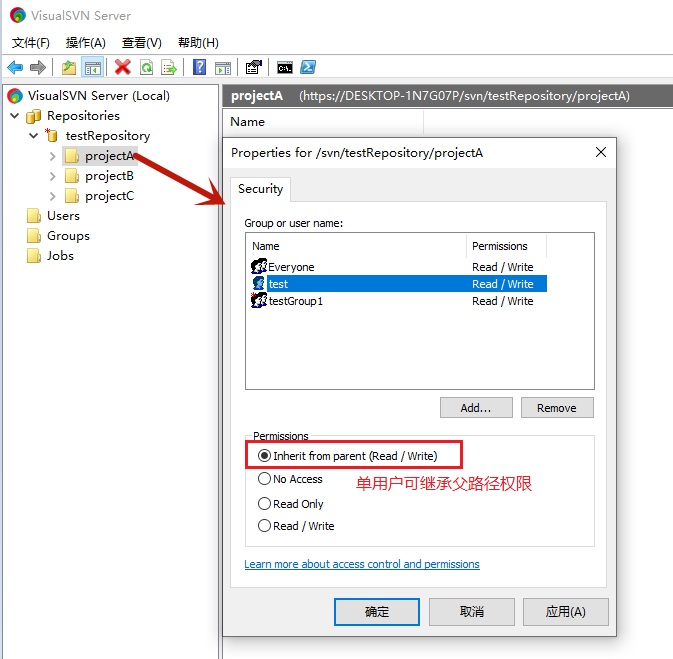
### 3.2.2权限配置方法

**方法一：界面配置。**

启动VisualSVN Server Manager，在创建好的版本库上右键-选择Properties，添加用户/组，并为其分配权限。



权限的分配还可以细化到版本库中某一项目甚至某一文件



**方法一：配置文件配置。**

版本库创建完成后，会生成基于版本库名为根目录的目录结构和基本的配置文件，权限配置信息保存在authz文件中，通过修改该配置文件为版本库配置用户权限。

1)配置文件相对路径：

.../版本库名/conf/authz

注：如果对含有中文名称的文件夹进行权限管理，需要把authz保存为UTF-8格式，建议使用UltraEdit或EditPlus等文本编辑工具。

2)权限配置基本语法：

[/path]

user1=r #路径/path上user1有读权限

user2=rw #路径/path上user2有读写权限

\*= #路径/path上其他所有人没有权限

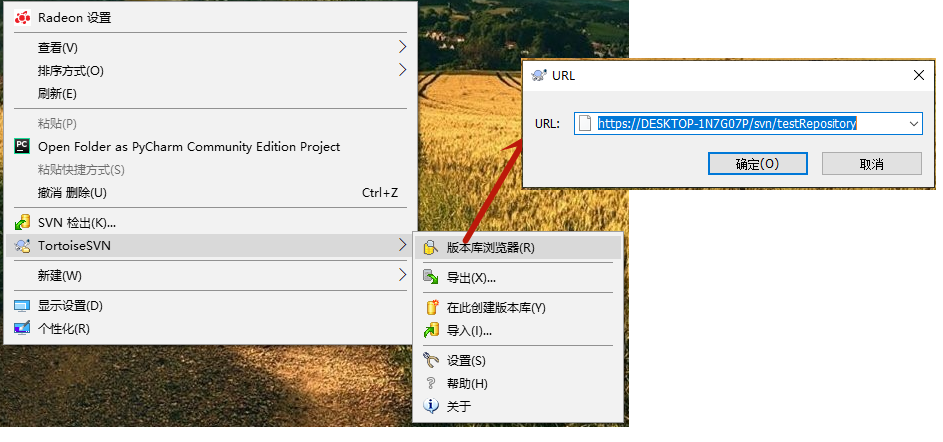
[/path/path2]

user1=rw #路径/path/path2上user1有读写权限

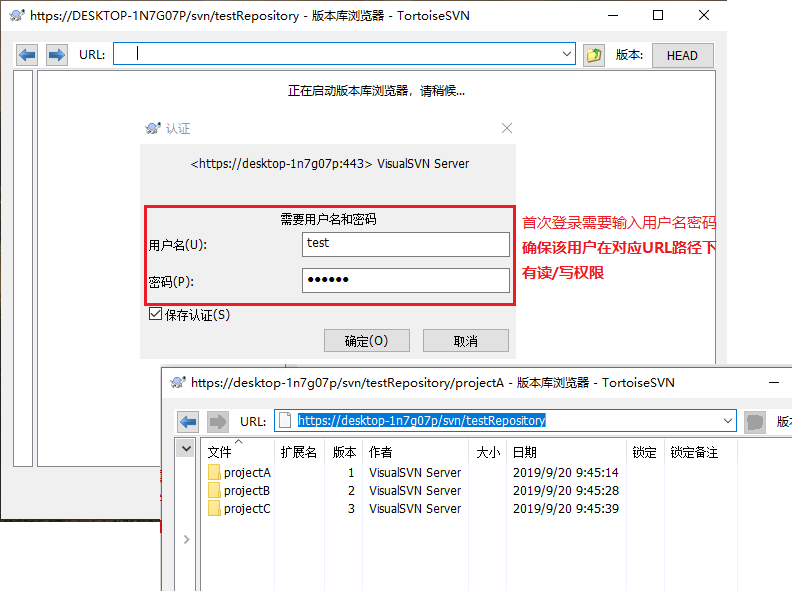
# 客户端操作

**操作一：连接版本库**

右键-TortoiseSVN-版本库浏览器，输入创建版本库时生成的URL，点击确定

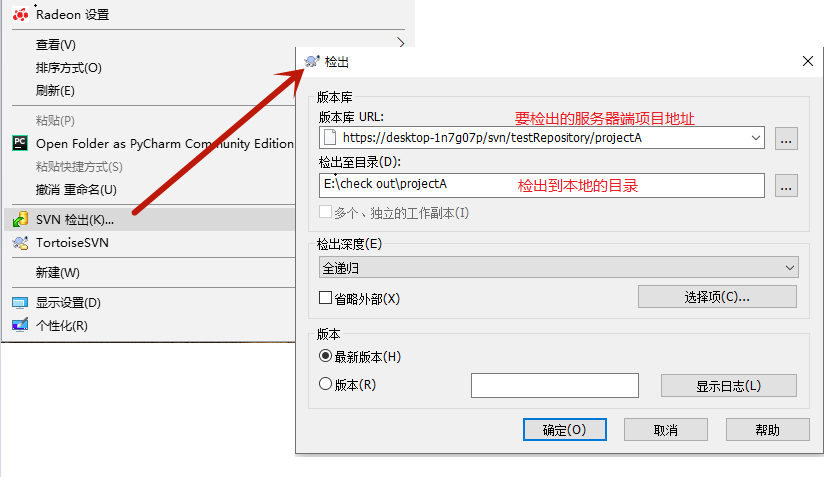


首次登陆需要输入用户名密码进行认证，可勾选保存认证，下次登陆不需要再次认证

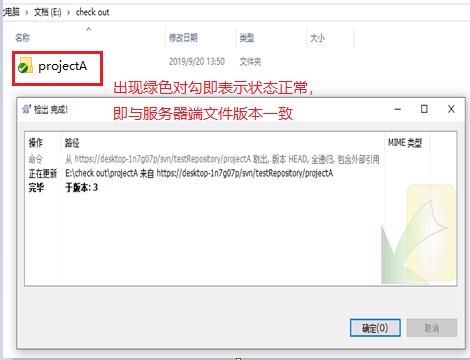


**操作二：检出项目**

右键-SVN检出-输入URL和检出目录，点击确定

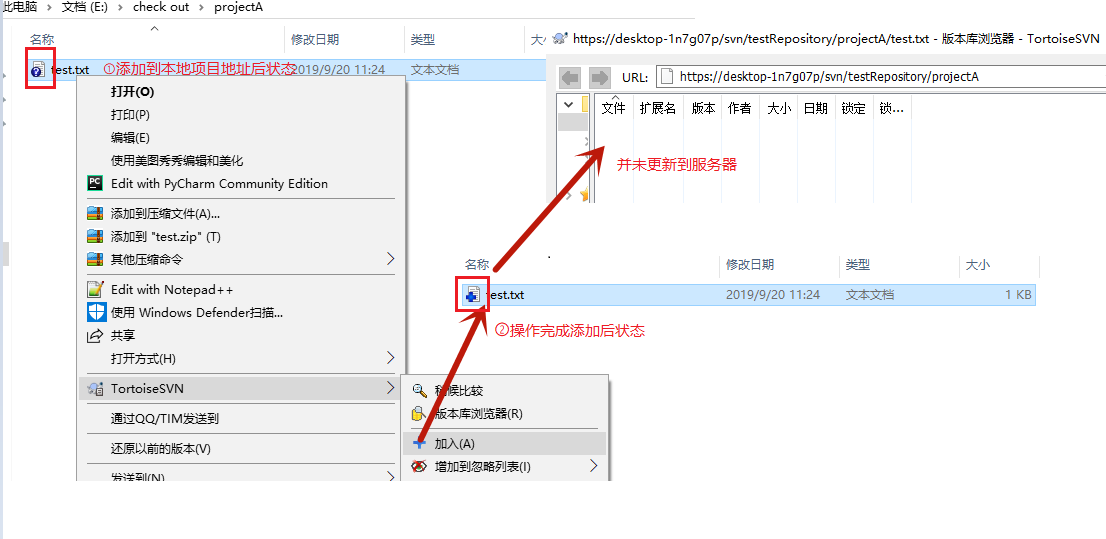


检出过程会被详细显示

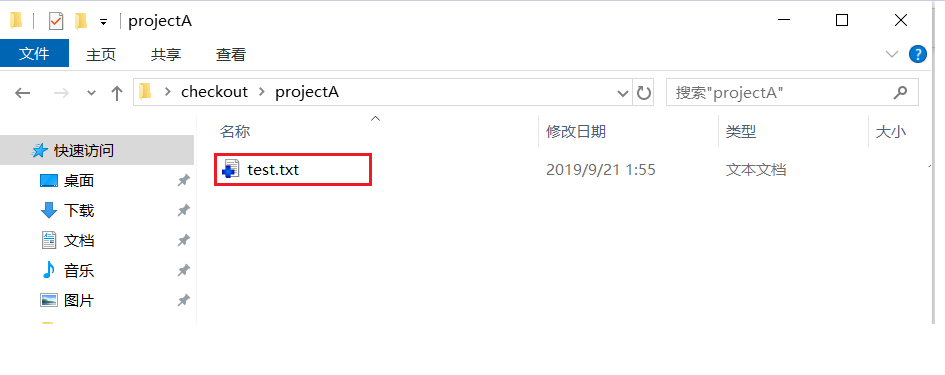


**操作三：添加文件（本地添加）**

复制文件到检出目录，文件右键**TortoiseSVN-加入，点击确定**

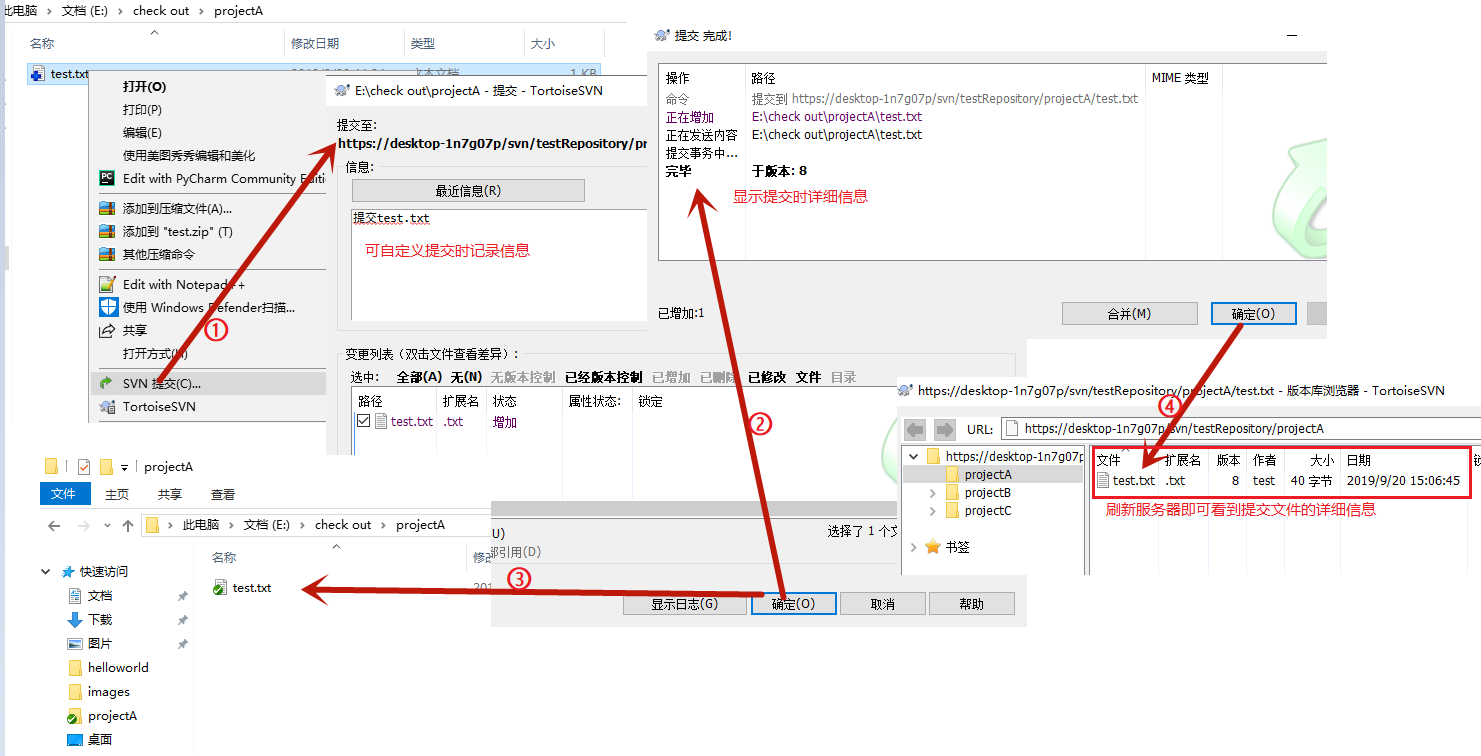


文件状态显示为蓝色加号即表示添加成功，此时并未实际将文件添加到服务器

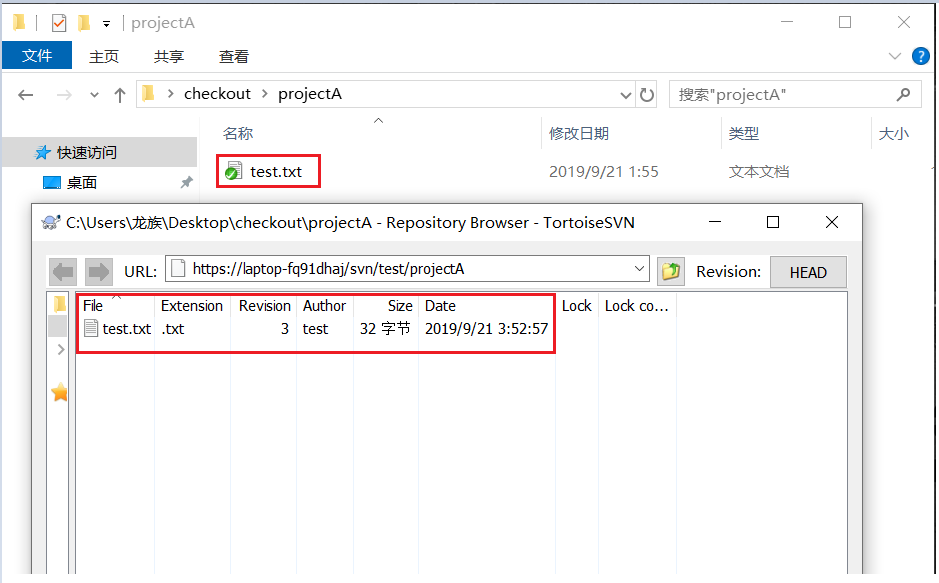


**操作四：提交文件**

文件右键TortoiseSVN提交，点击OK进行提交。

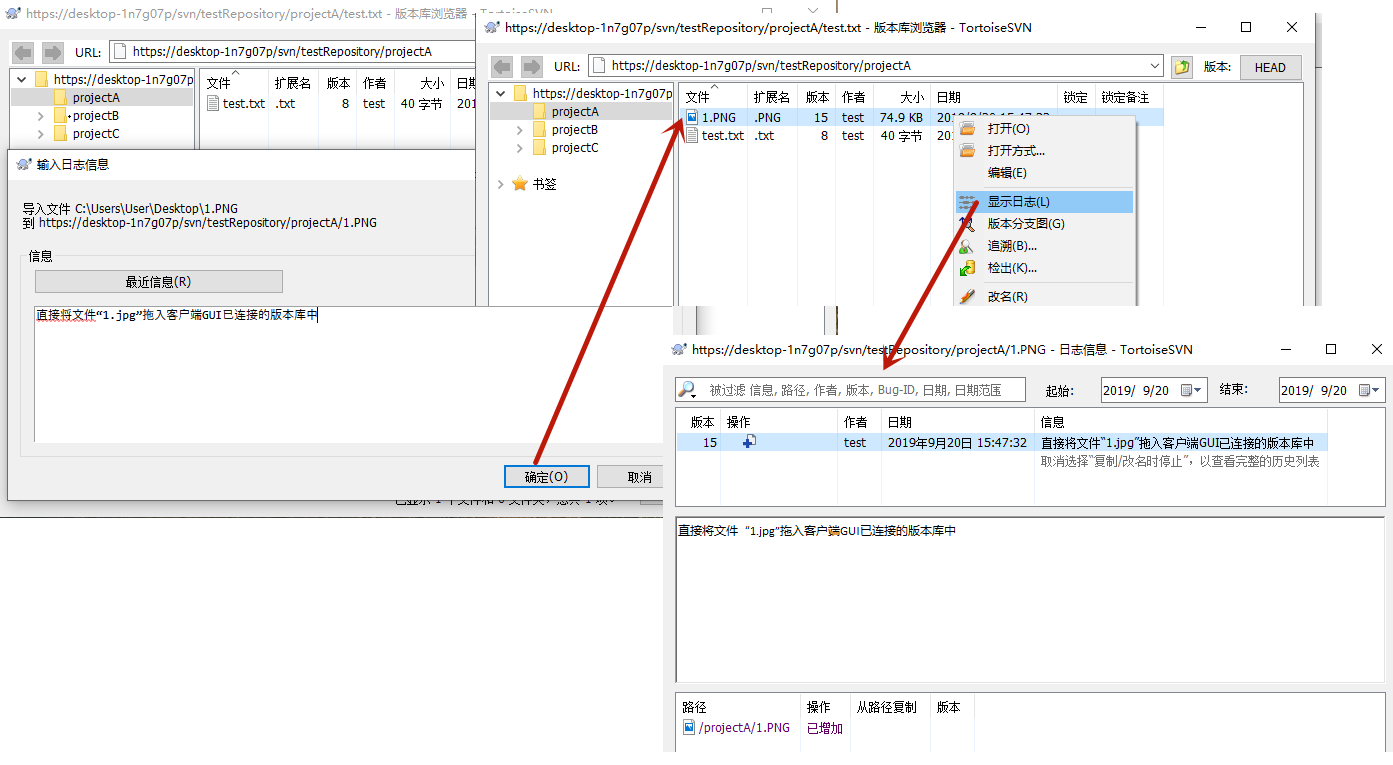


提交完成后文件状态变为绿色对勾，服务器也收到提交的文件

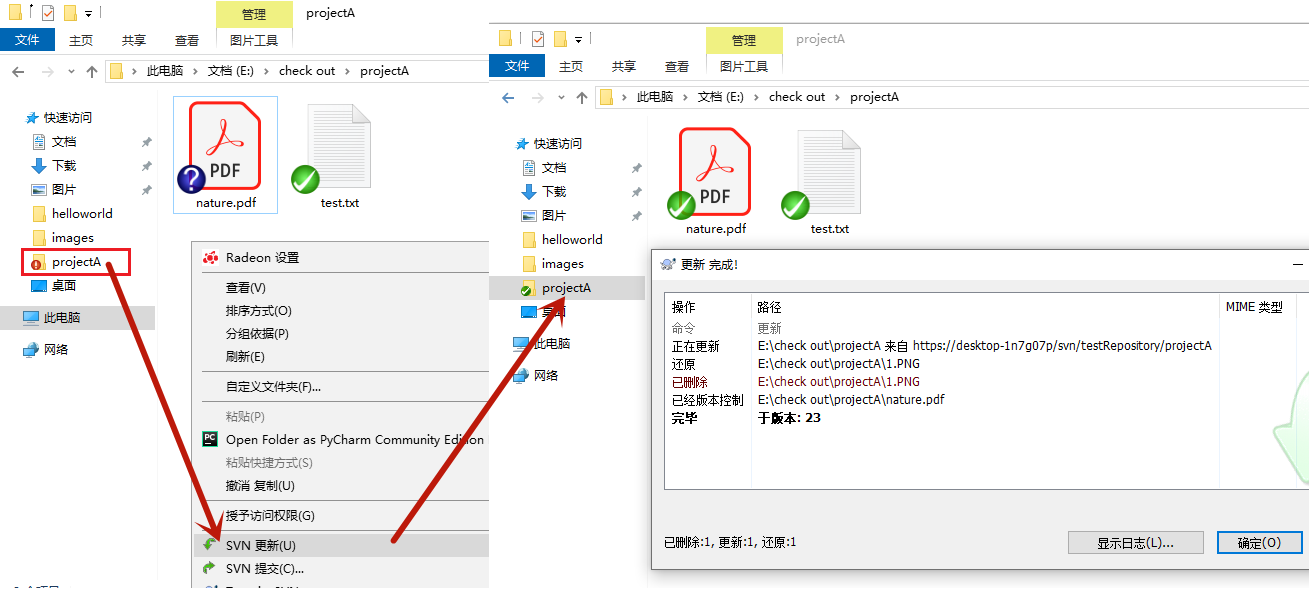


**操作五：添加文件（客户端添加）**

直接在客户端将文件拖入服务器对应路径下，点击确定，文件便添加到了服务器。

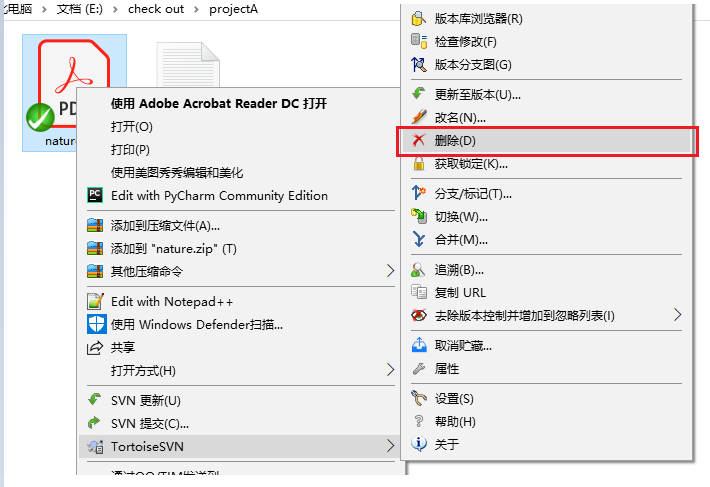


**操作六：更新**

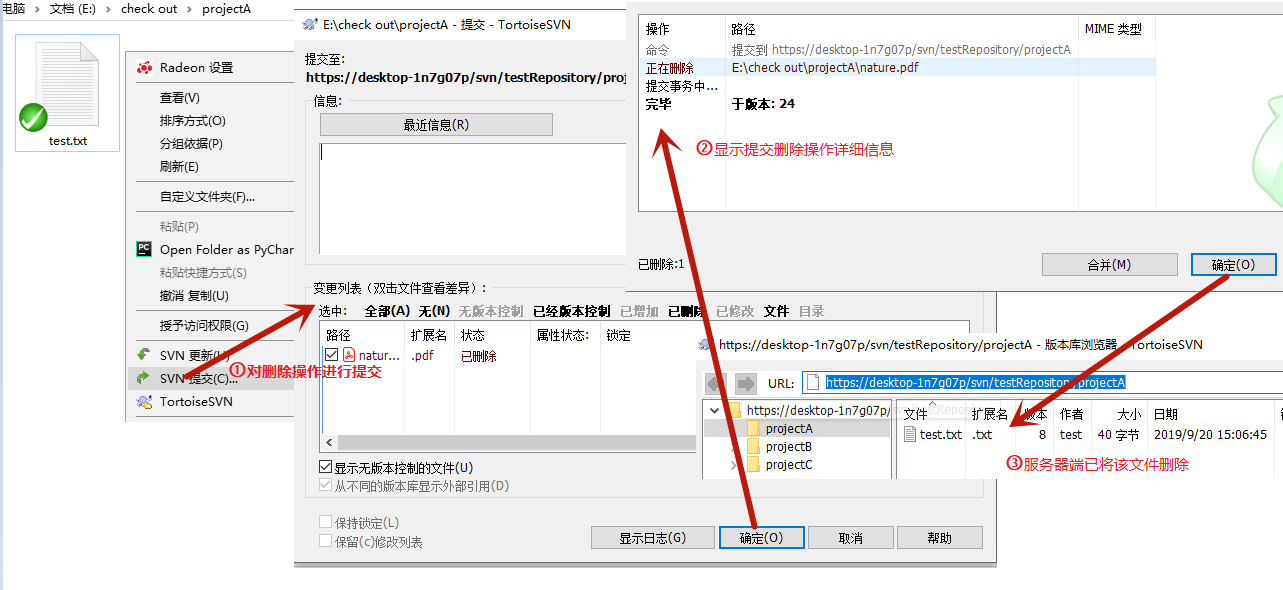
文件右键TortoiseSVN更新，点击OK可将本地系统文件版本更新成与服务器一致版本。该操作不仅可以针对文件使用，还可以针对文件夹甚至整个版本库使用

**操作七：删除文件（本地删除）**

需要使用文件右键TortoiseSVN-删除操作，此时服务器端并未删除该文件

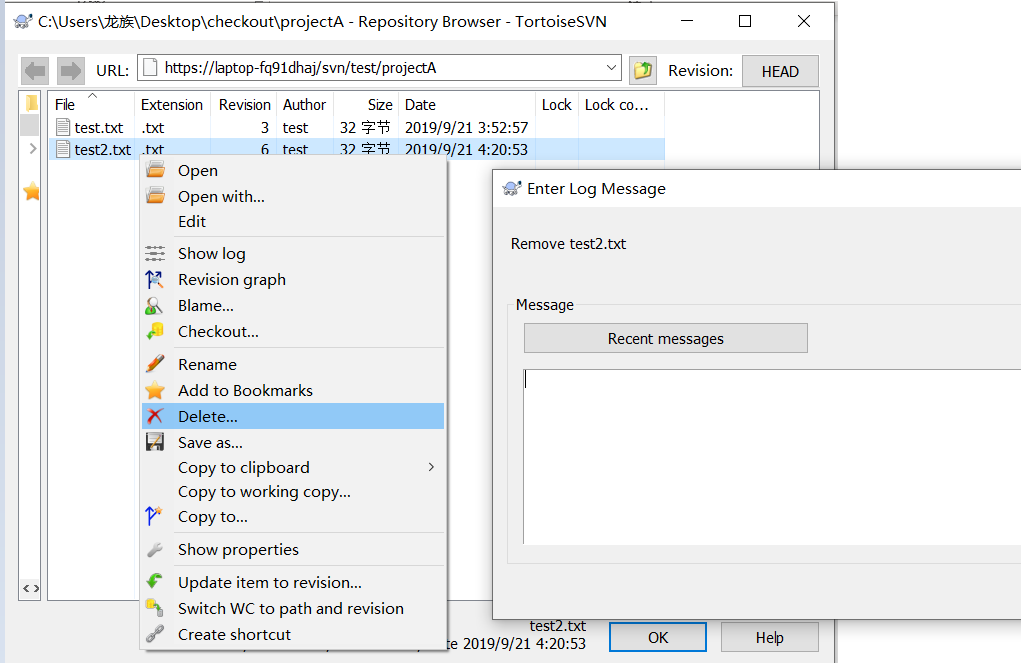


右键TortoiseSVN-提交，服务器端才会真正执行删除操作

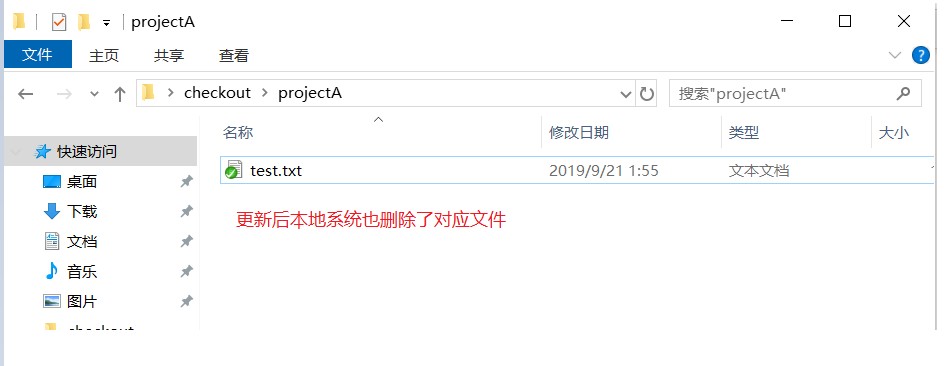


**操作八：删除文件（客户端删除）**

在客户端中选中文件后右键删除，点击OK文件便从服务器被删除

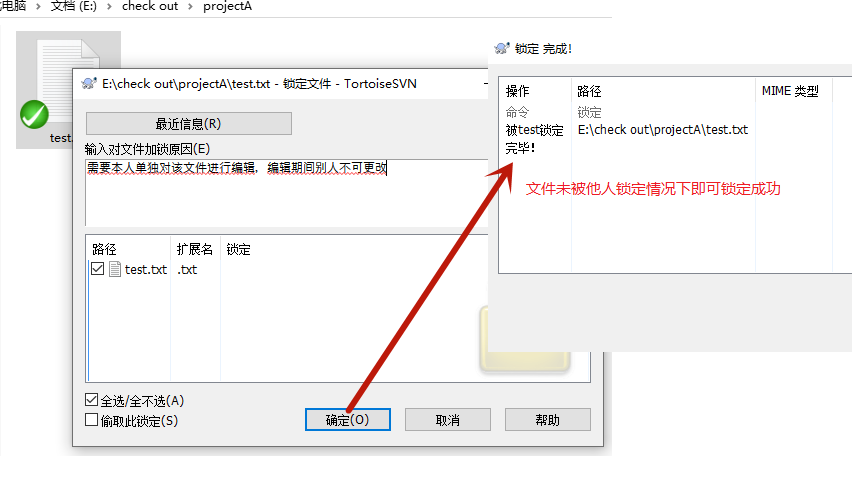


服务器删除文件后会使本地系统与服务器端存储内容不一致，在本地右键执行SVN更新操作后，本地系统也会删除对应文件，保持与服务器的一致性

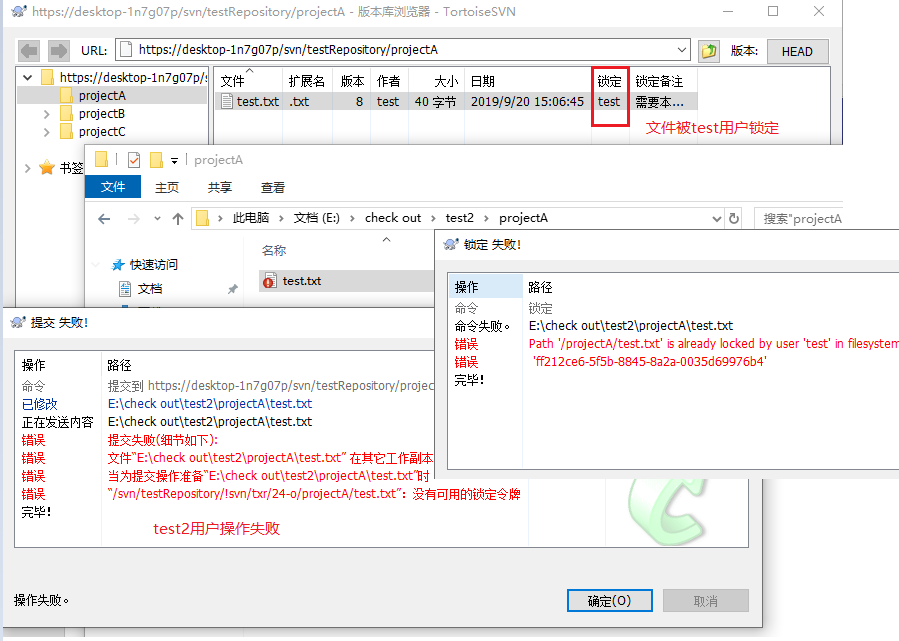


**操作九：锁定**

文件右键TortoiseSVN-锁定，点击确定锁定文件。锁定操作也可以针对文件夹甚至整个版本库

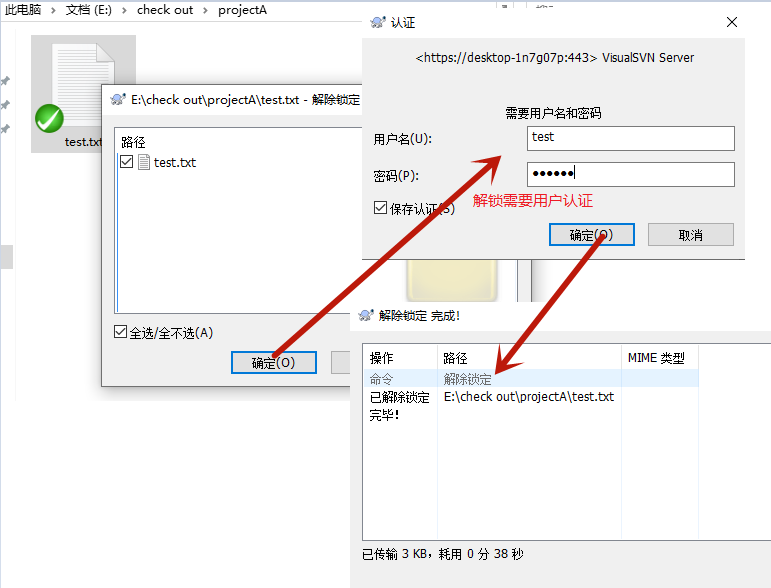


文件锁定期间只允许上锁人对文件进行更新，他人必须等待解锁后才能更新该文件，否则会提示更新失败

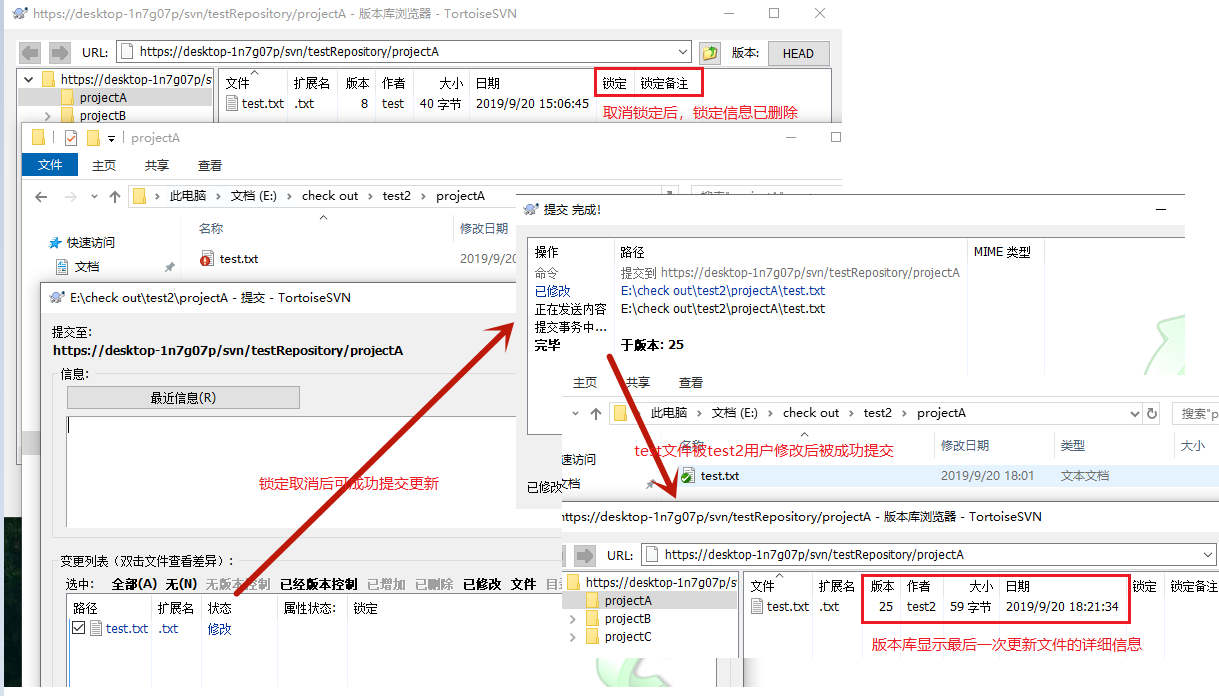


**操作十：解锁**

文件右键TortoiseSVN-解锁，输入认证信息点击确定，文件即解锁成功

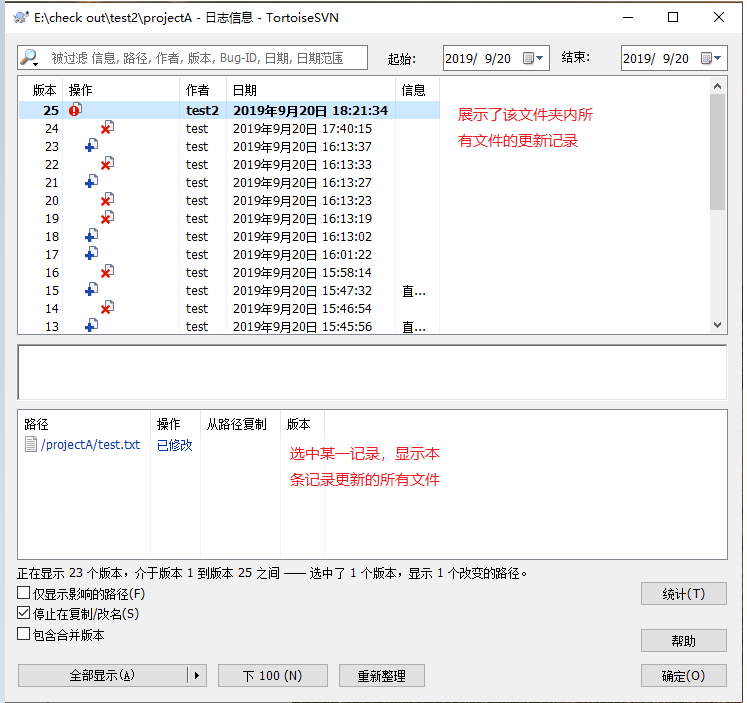


解锁后他人可继续更新该文件，服务器显示最后一次更新的详细信息

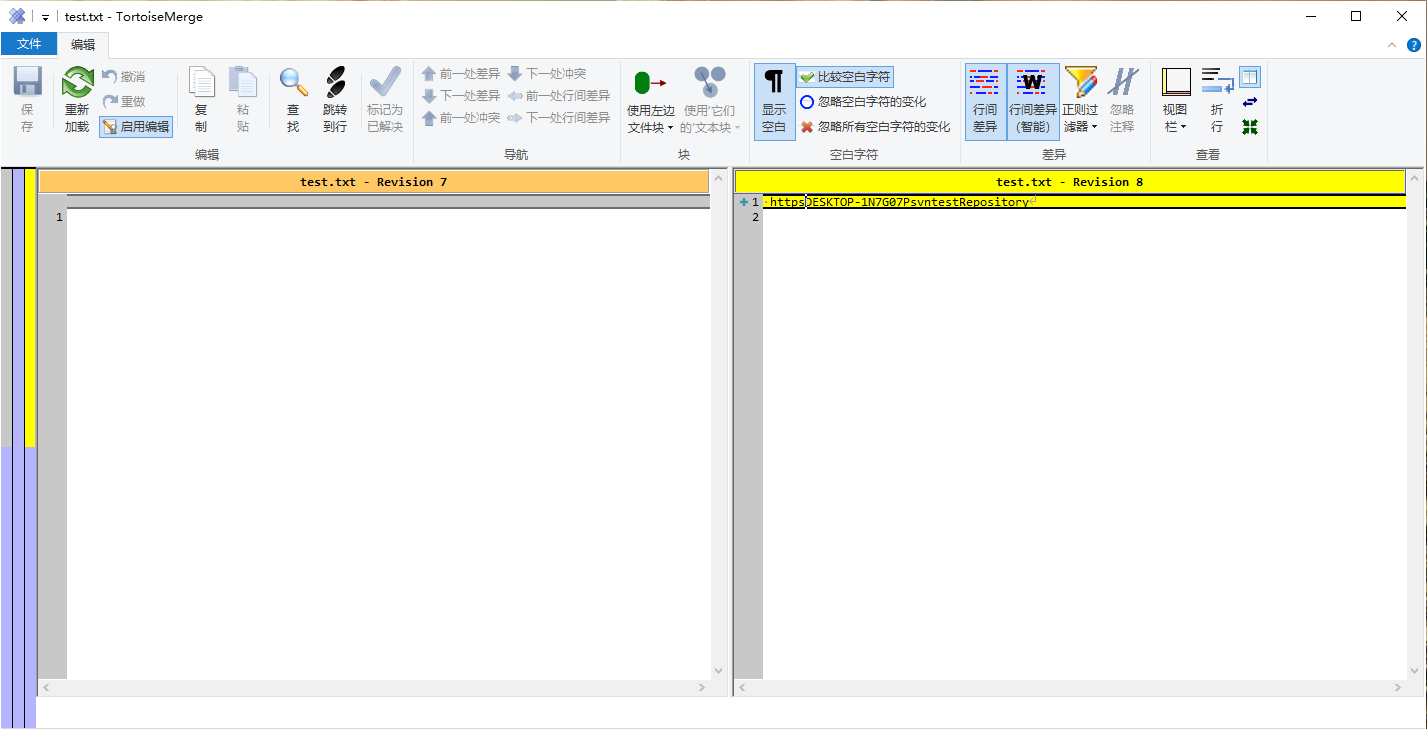


**操作十一：获取历史版本**

文件夹右键TortoiseSVN-显示日志，记录了文件夹内所有文件的更改记录

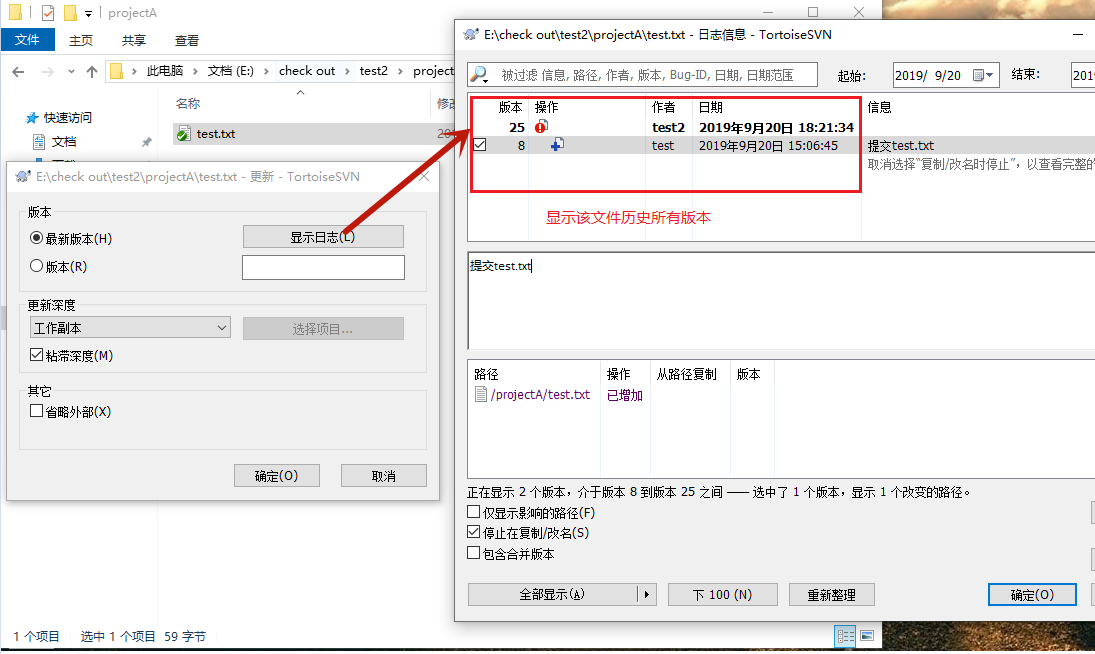


双击某一文件可以比较文件版本间的内容差异



**操作十二：版本回退**

文件右键TortoiseSVN-更新至版本，记录了文件的所有版本记录，选择自己想要回退的版本，点击确定即可回退到目标版本



# 参考文献

<https://baike.baidu.com/item/SVN>

<https://www.runoob.com/svn/svn-intro.html>