

# SVN 使用手册

撰写分组： 三四组

全体组员：

三组 :刘久菊 翟君敏 陈刚 穆堃龙 汪欣

四组：刘莎瑞 杨慧莉 杨申睿

王江 张姍 龙思同 张皓鑫

## 成员分工

姓名	分工
刘莎瑞	SVN 安装、检入项目
刘久菊	检入项目、导出项目
杨申睿 张姍	导出项目、提交
王江	提交、更新
杨慧莉	更新、查看日志
龙思同	查看日志、冲突及解决
翟君敏	冲突及解决、锁定解锁
陈刚	锁定解锁、版本控制及回滚
穆堃龙	版本控制及回滚、SVN 安装
张皓鑫 汪欣	任务安排、文件整合

## 目录

1 SVN 安装.....	4
1.1. SVN 功能简介 .....	4
1.2. 安装流程.....	4
2 导入项目.....	7
2.1. 场景.....	7
2.2. 操作流程.....	7
3 检出项目 .....	10
3.1. 场景.....	10
3.2. 操作流程.....	10
注释.....	11
4 提交 .....	13
4.1. 场景.....	13
4.2. 操作流程.....	13
5 更新 .....	16
5.1. 场景.....	16
5.2. 操作流程.....	16
6 查看日志 .....	16
6.1. 场景.....	16
6.2. 操作流程.....	17
7 冲突及解决 .....	18
7.1. 产生冲突的原因.....	18
7.2. 解决方法.....	19
7.3. 选择比较工具.....	22
8 锁定/解锁 .....	24
8.1. 场景.....	24
8.2. 操作流程.....	24
9 版本控制及回滚 .....	26
9.1. 场景.....	26
9.2. 操作流程: .....	26
10 svn 使用注意事项.....	35
10.1. 安装.....	35
10.2. 检出.....	35
10.3. 导入.....	36
10.4. 提交.....	37
10.5. 更新.....	37
10.6. 查看日志.....	37
10.7. 冲突与解决方案.....	38
10.8. 锁定与解锁.....	39
10.9. 版本的控制及回滚.....	39
10.10. 经验总结.....	39

# 1 SVN 安装

## 1.1. SVN 功能简介

SVN 是 Subversion 的简称,是一个开放源代码的版本控制系统,它采用了分支管理系统,所有开发者在开始新一天的工作之前必须从服务器获取代码,然后开发,最后解决冲突,提交。所有的版本信息都放在服务器上,如果脱离了服务器,开发者基本上可以说是无法工作的。简而言之,SVN 就是用于多个人共同开发同一个项目,达到共用资源的目的工具。

## 1.2. 安装流程

下载网址：<https://tortoisesvn.net/downloads.html>

Step1.在官网上根据自己的操作系统位数下载 SVN。

for 32-bit OS

TortoiseSVN 1.9.7 - 32-bit

for 64-bit OS

TortoiseSVN 1.9.7 - 64-bit

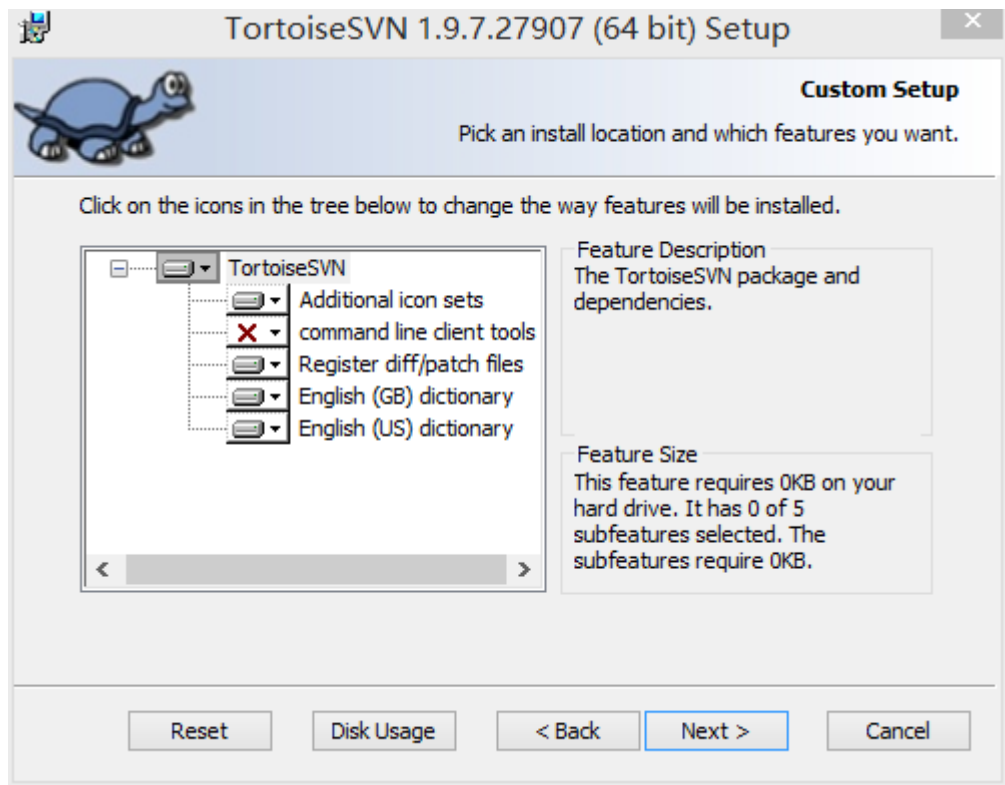
Step2.下载好安装包后进行安装。

 TortoiseSVN-1.9.7.27907-x64-svn-1.9.... 2018-01-19 10:33 Windows Install... 17,172 KB

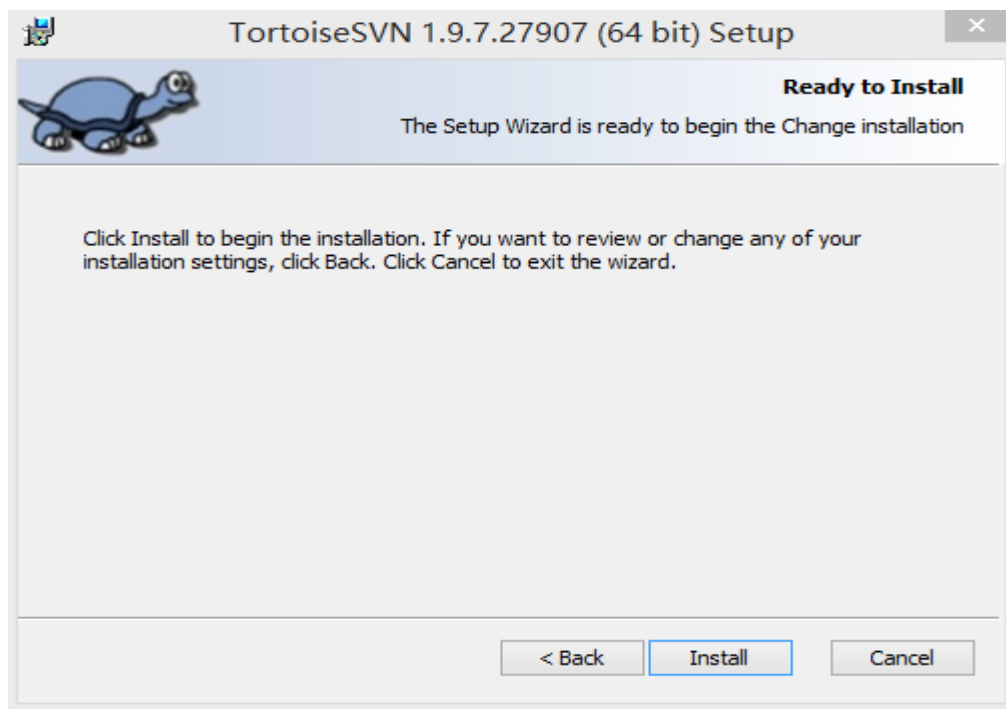
点击 Next 进行下一步



Step3.进入自定义安装界面，可以修改路径，也可以直接点击 Next。



Step4.点击 Install 完成安装。



Step5.安装完成后，点击 finish 后，桌面点击右键出现 SVN Checkout 选项，说明安装成功。



## 2 导入项目

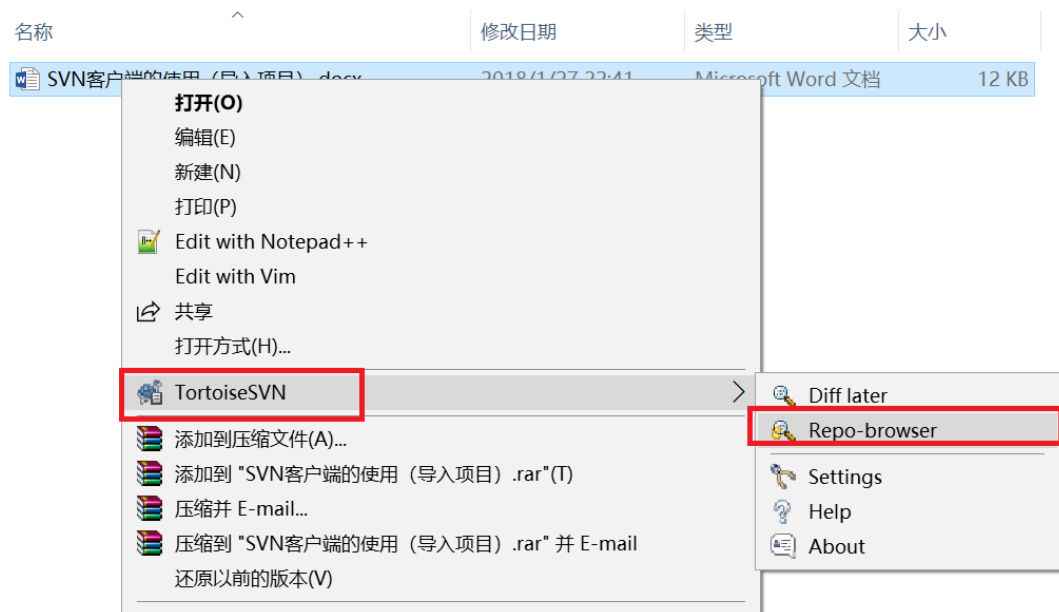
### 2.1. 场景

如果有做好的项目需要导入进已有的仓库,可以使用SVN导入项目的功能。

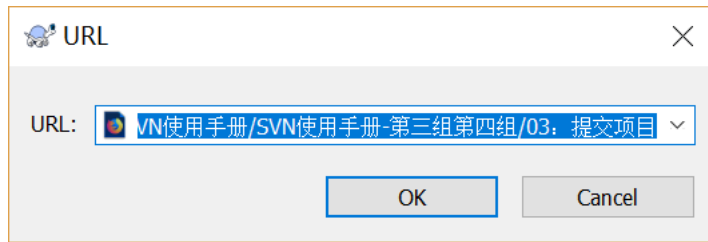
**注意：导入的项目会直接进行提交，所以要导入的项目一定是确定要交付的版本**

### 2.2. 操作流程

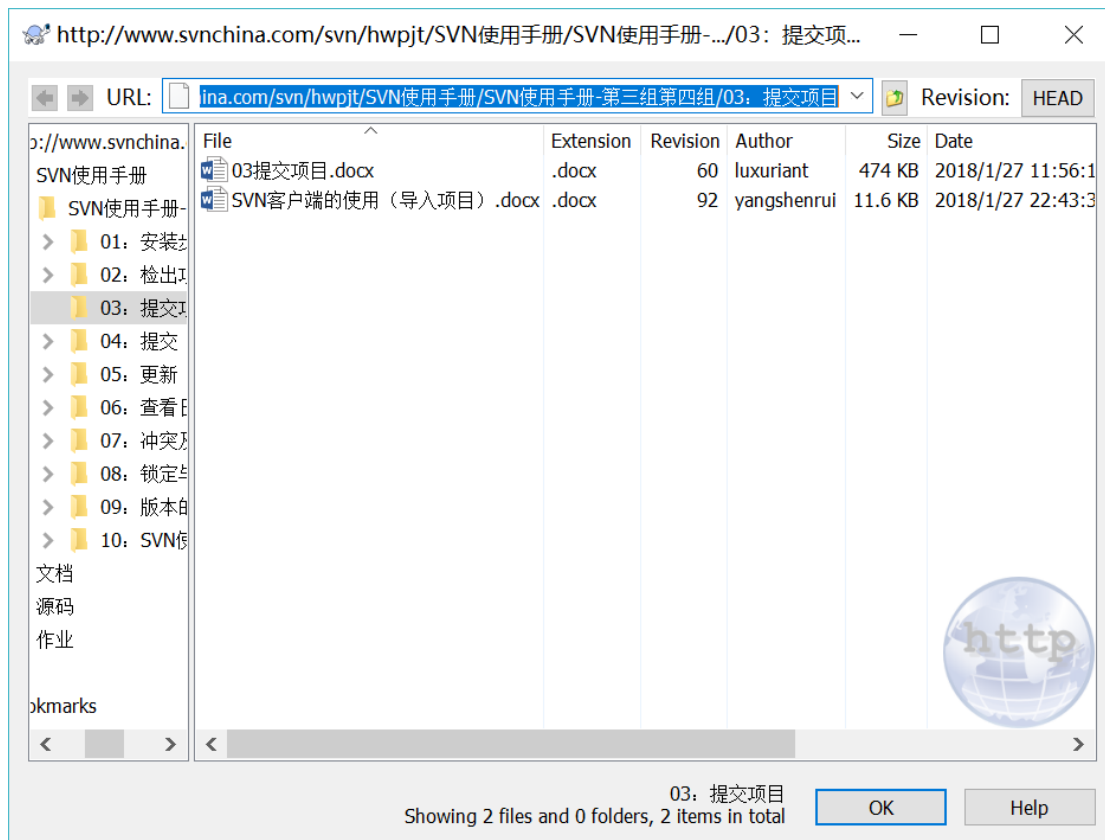
Step1.打开 Tortoise SVN 选择 Repo-browser。



Step2.如果是在本地仓库打开的 Repo-browser 会自动获取仓库 URL,如果是在本地仓库外,需要填入要导入的仓库 URL。

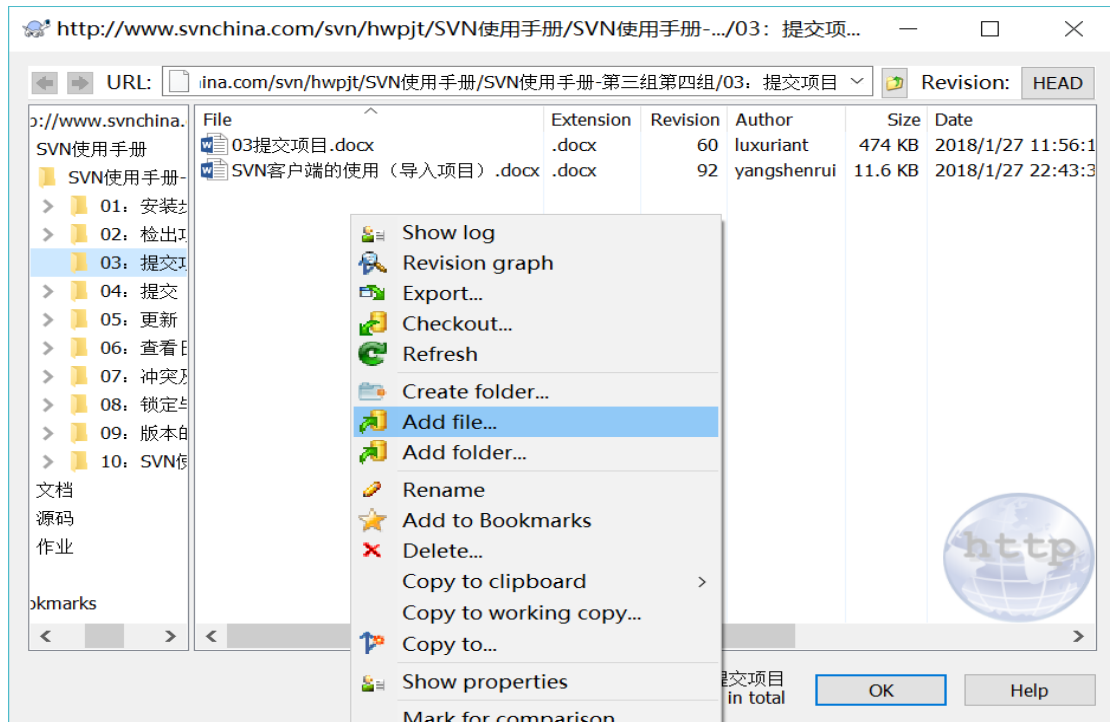


选择 OK，出现如下窗口：

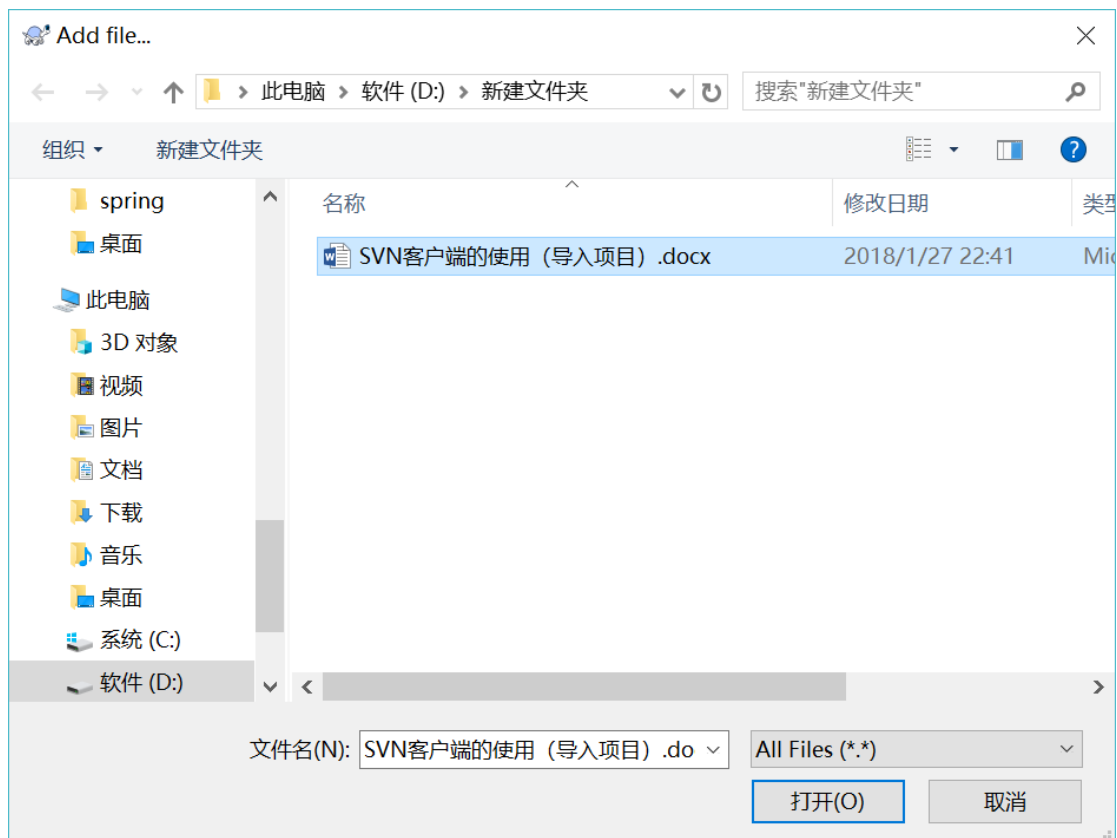


Step3.右击，选择 Add file...导入文件，或选择 Add folder...导入文件夹。

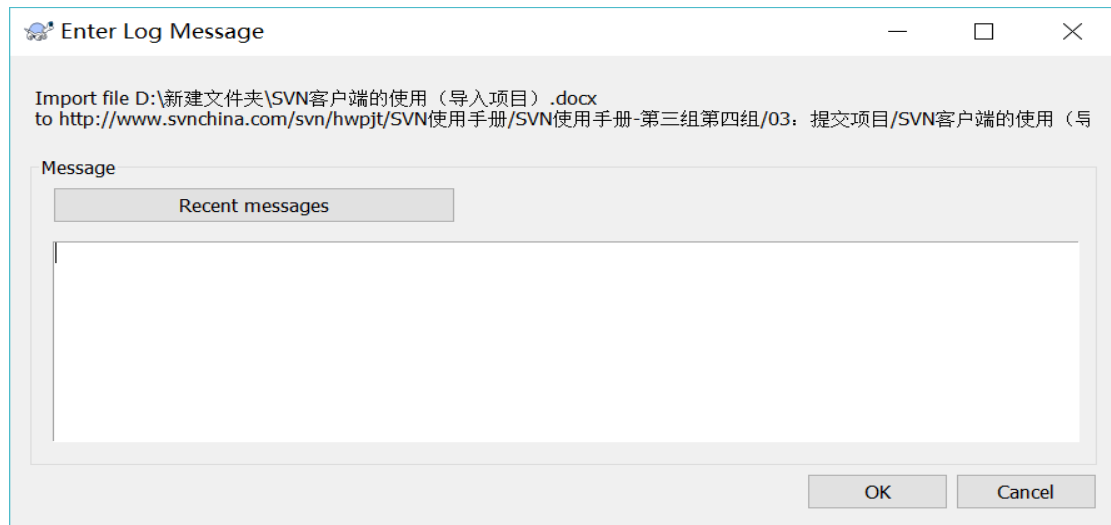




Step4.在弹出的 Add file...窗口填入要导入的本地文件( 文件夹 )的 URL 或选择要导入的本地文件 ( 文件夹 )。



Step5.填入提交日志。



点击 OK，完成导入。

## 3 检出项目

### 3.1. 场景

从版本库检出文件，含有.SVN 隐藏文件夹，受 SVN 客户端控制，对其进行  
的增删改操作可以被 SVN 客户端识别。

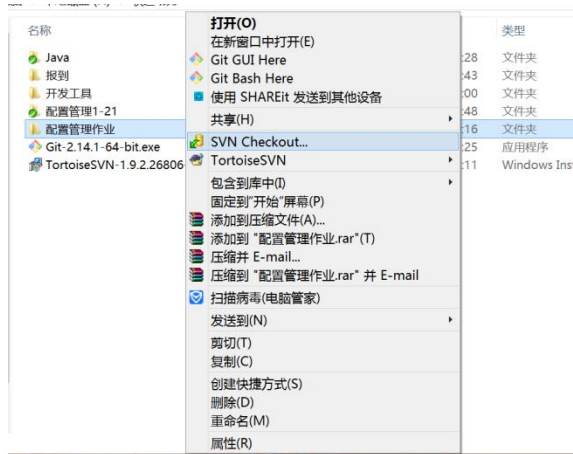
### 3.2. 操作流程

Step1.在本地创建一个保存检出项的目录（文件夹）。

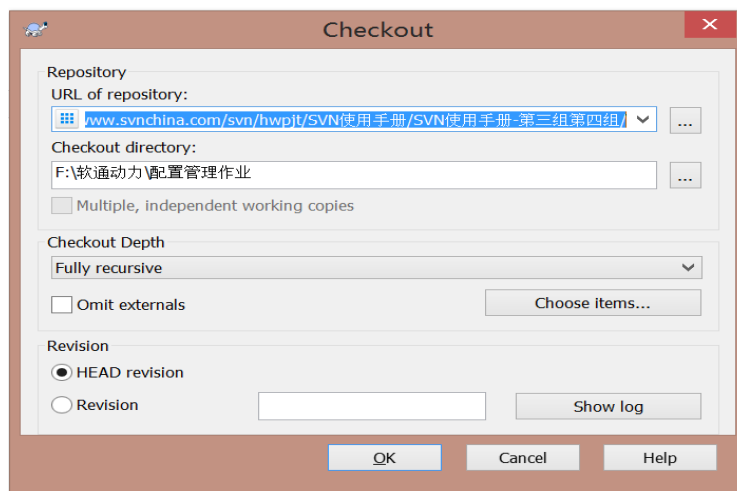
Step2.输入版本库地址和检出目录。

方式一：

(1) 选中文件夹单击右键>>SVN Checkout...>>输入 SVN 用户名和密码。



(2) 输入版本库地址。



## 注释

- **URL of repository** : 填写仓库路径即可。
- **Checkout directory** : 填写迁出目录 ( 即保存检出项的目录 ) 。
- **Checkout Depth 可选项** :
  - Fully recursive——全递归 ( 检出完整的目录树 , 包含所有的文件或子目录 ) 。
  - Immediate children,including folders——直接子节点 , 包含文件夹 ( 检出目录 , 包含其中的文件或子目录 , 但是不递归展开子目录 ) 。
  - Only file children——仅文件子节点 ( 检出指定目录 , 包含所有文件 , 但是不检出任何子目录 ) 。
  - Only this item——仅此项 ( 只检出目录。不包含其中的文件或子目录 )
- **Choose items** : 选择要检出的目录。
- **HEAD Revision** : 最新版

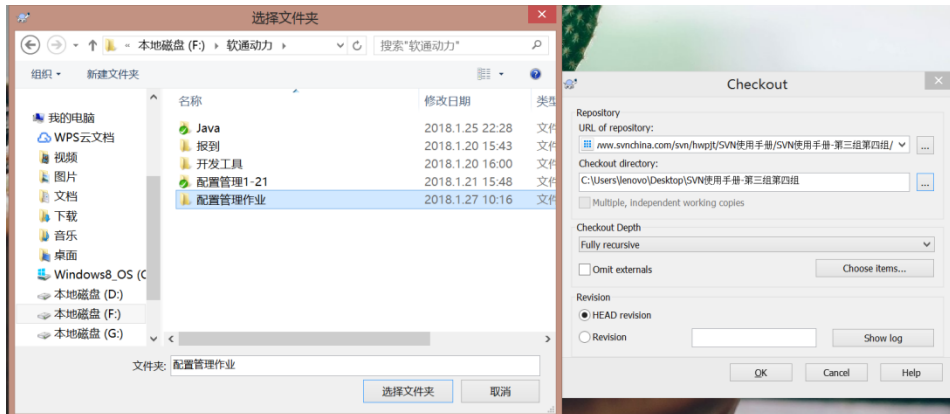
- **Revision** : 点击 Show log 可以选择任意一个版本。

方式二：

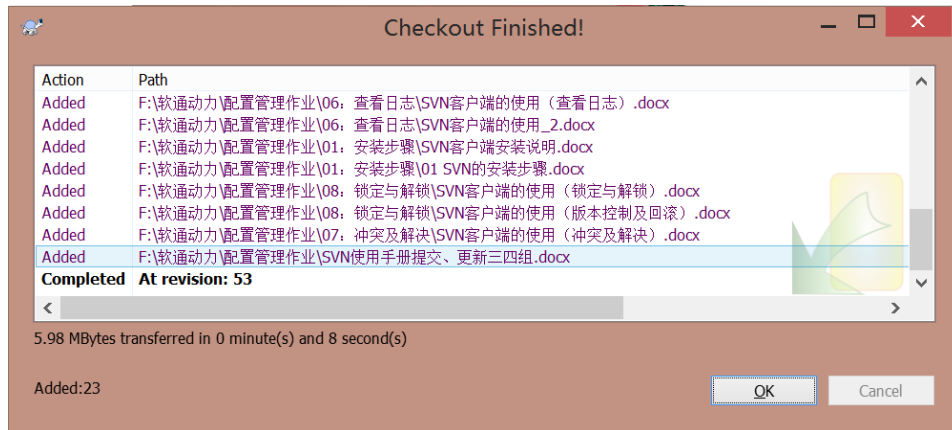
- (1) 在桌面单击右键>>SVN Checkout...>>输入 SVN 用户名和密码登录。



- (2) 输入版本库地址，点击浏览按钮，选择保存检出项的目录。



Step3.点击 OK，开始检出，检出完成。



检出完成后就可以在保存项目的目录下找到已经检出的项目 ,可对其进行其他的操作。

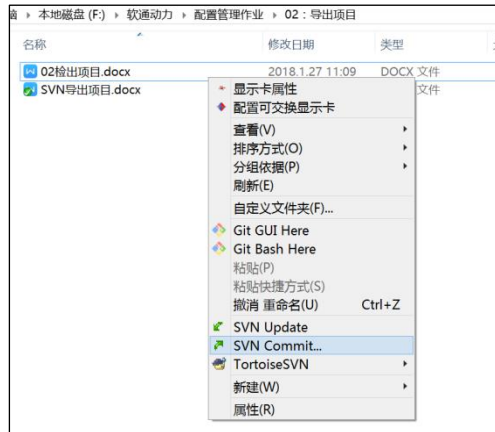
## 4 提交

### 4.1. 场景

对检出的项目做了修改后，提交到 SVN 服务器上。

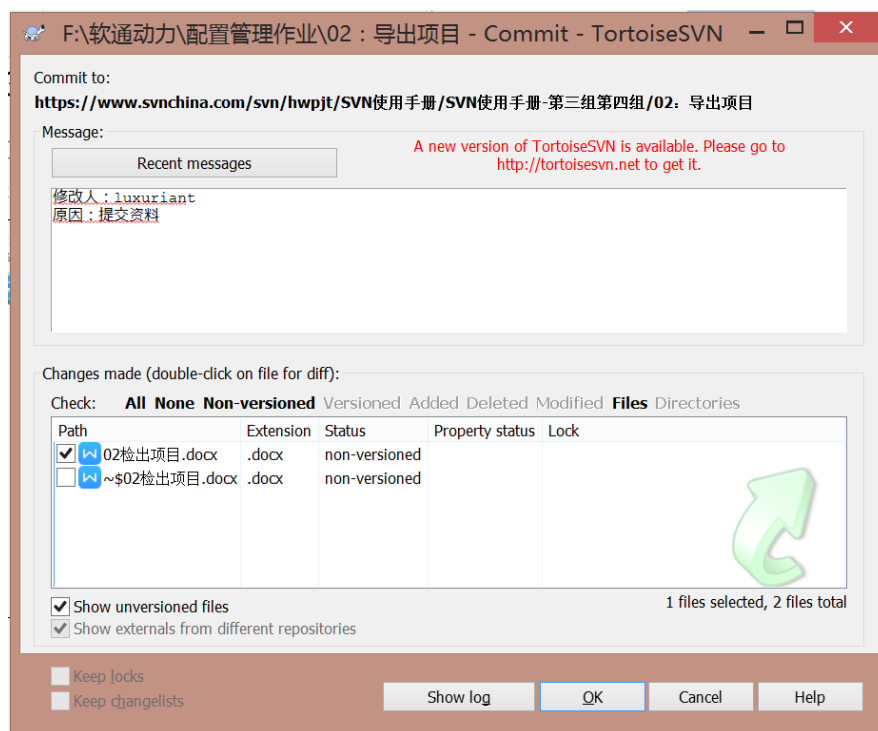
### 4.2. 操作流程

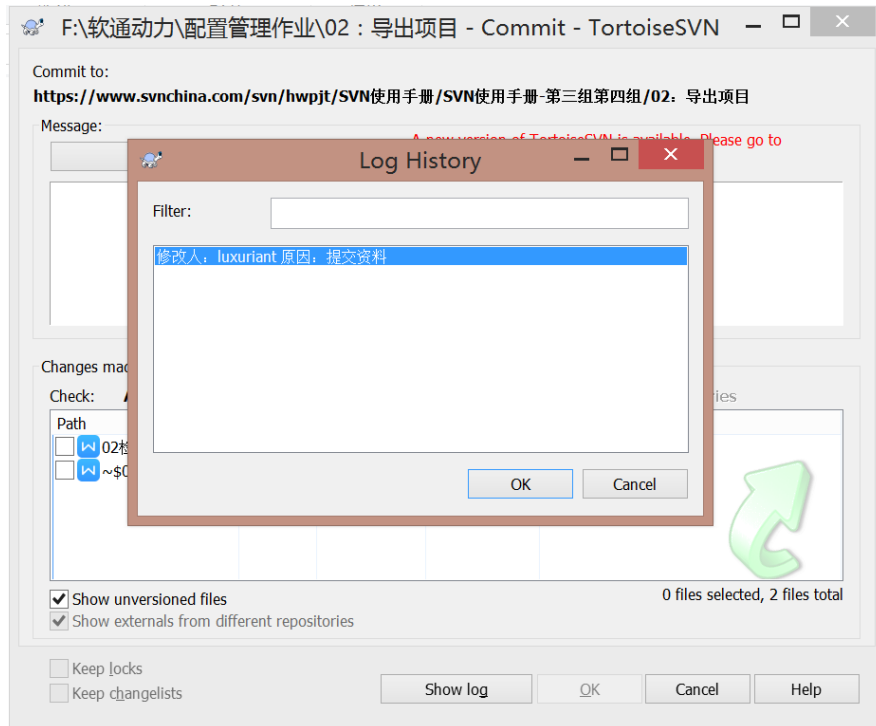
Step1.在目录空白处单击右键>>SVN Commit...，默认选择所有有更改的项目进行提交；如果只选择某一个文件提交，可以在文件单击右键进行提交。



Step2.Message 必填，填写修改人和原因等备注信息，可以点击“Recent messages”，选择以前的模板进行填写。

Step3.勾选要提交的项目。





Step4.填完所有信息之后，点击 OK，开始提交，直到提交完成。



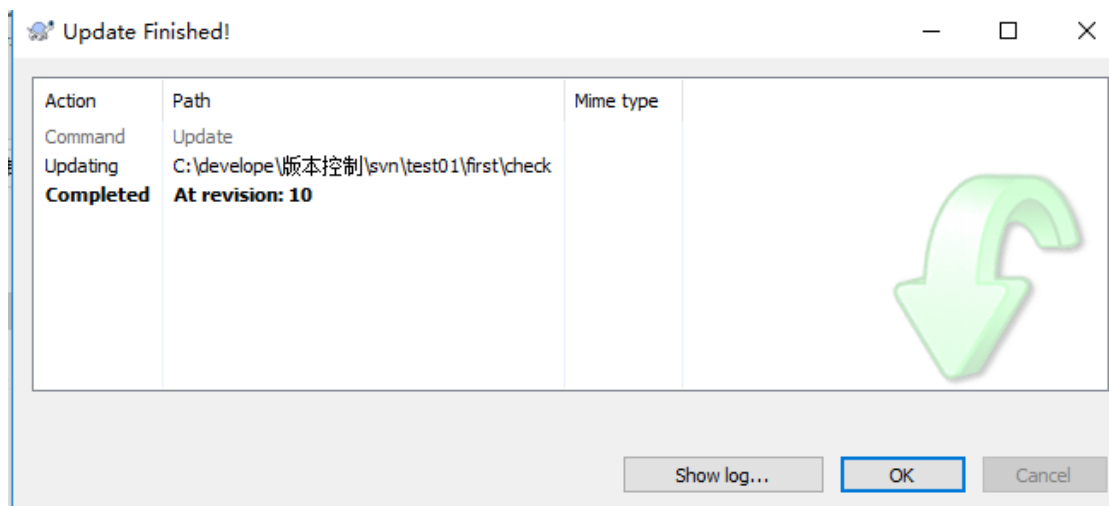
## 5 更新

### 5.1. 场景

当修改了一段代码需要提交到 SVN 上时需要先进行更新。

### 5.2. 操作流程

选中你需要更新的文件或文件夹，右键 update 进行更新。



## 6 查看日志

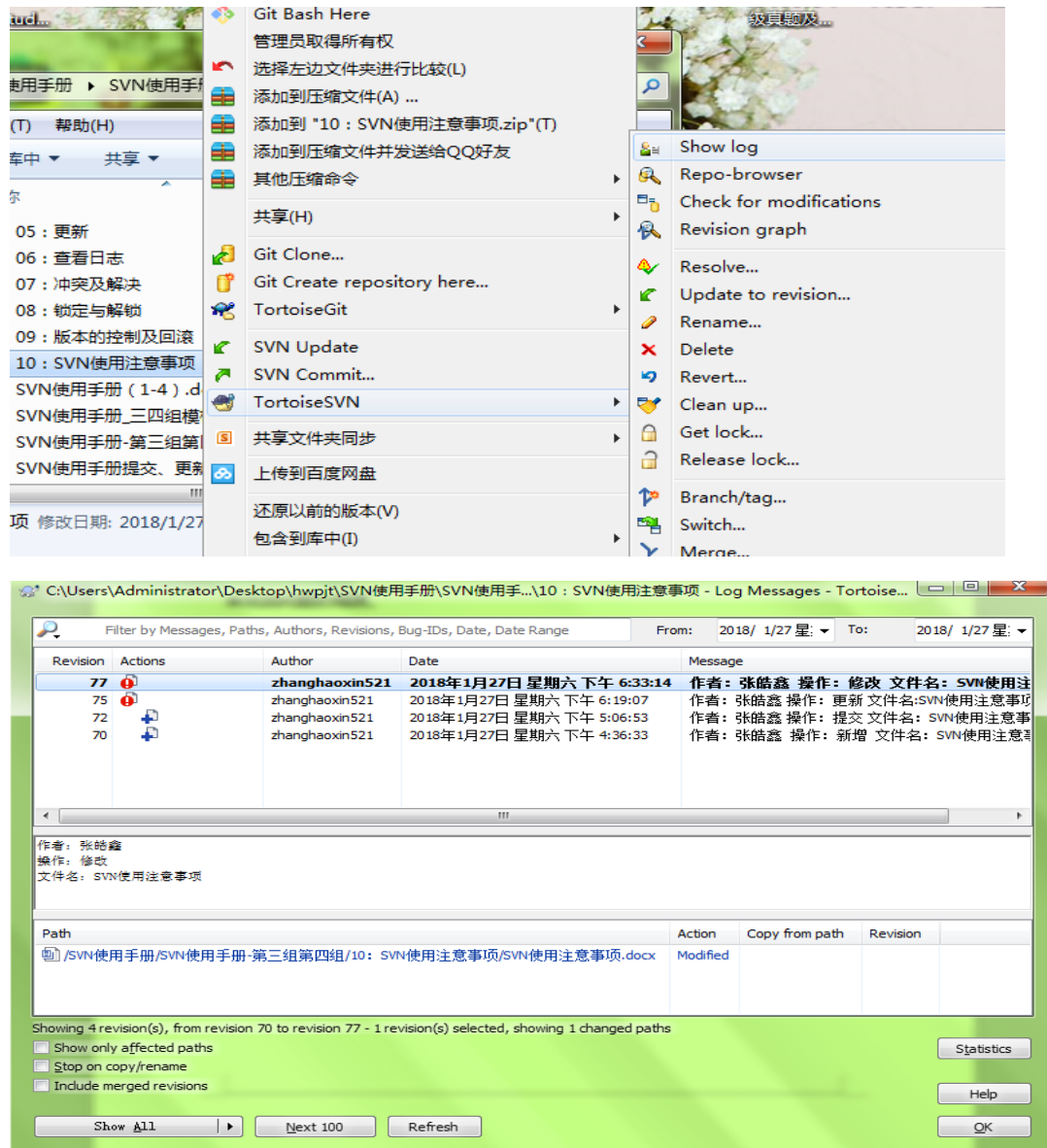
### 6.1. 场景

当有人提交的代码造成系统运行异常或者冲突时，可以通过查看日志来找到问题的根源。



## 6.2. 操作流程

选中要查看的文件，右键 showLog，此时我们就可以看到项目在整个时期的变化。

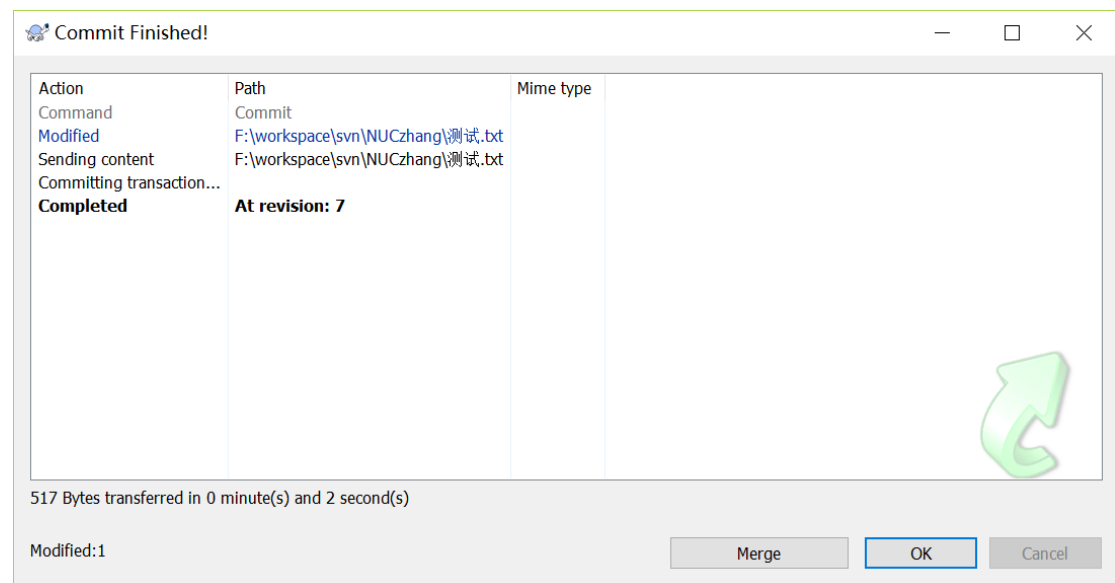
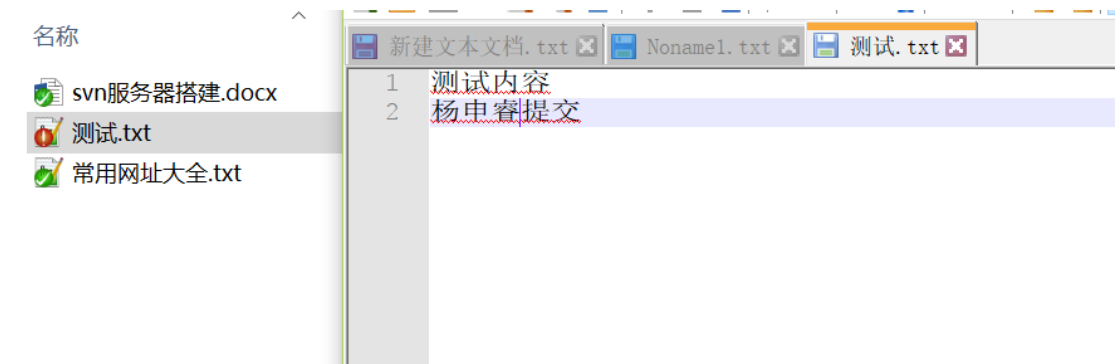


## 7 冲突及解决

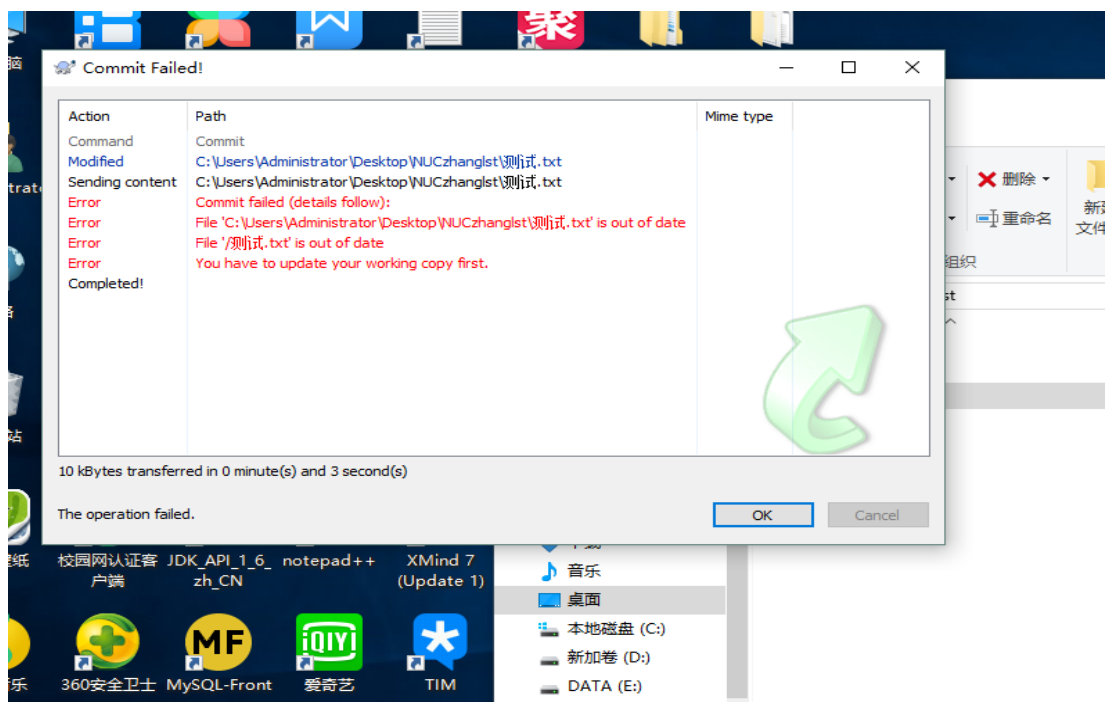
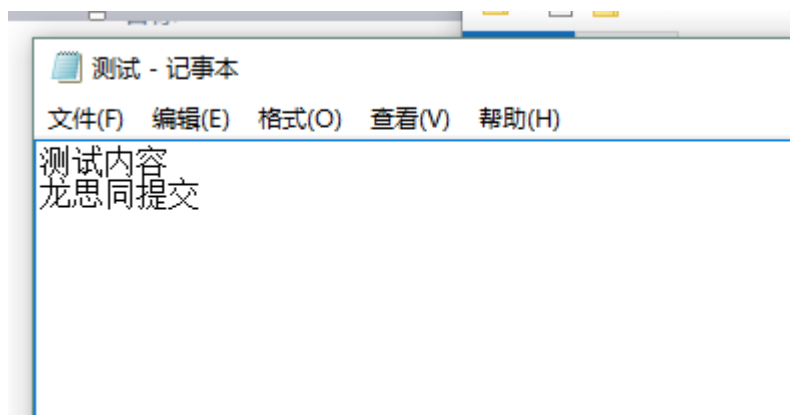
### 7.1. 产生冲突的原因

(1) 提交之前没有进行更新，操作的不是最新版本。

新建一个服务器，进行测试。



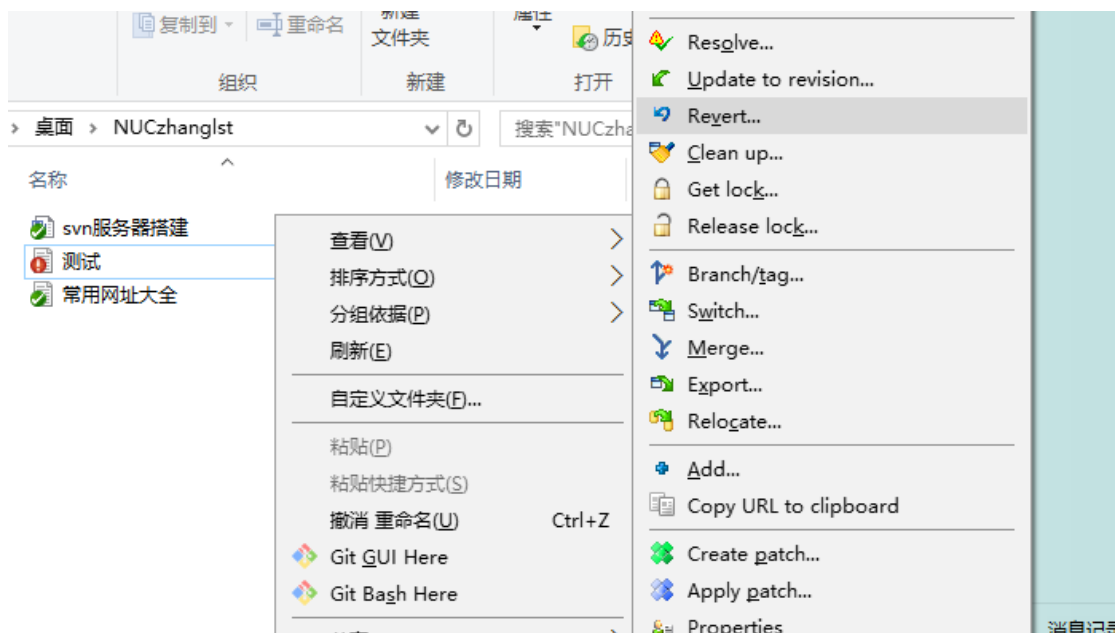
第一个人提交成功后，第二个人不更新直接提交这时就会产生文件冲突，如下



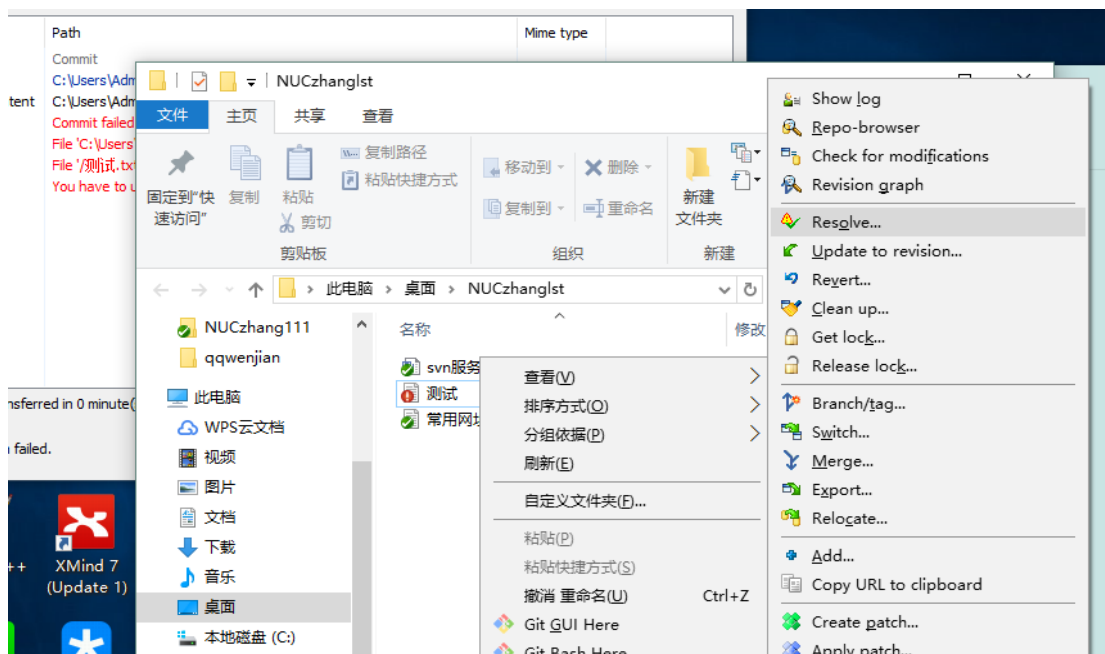
(2) 两个或者多个人对同一文件同时提交。

## 7.2. 解决方法

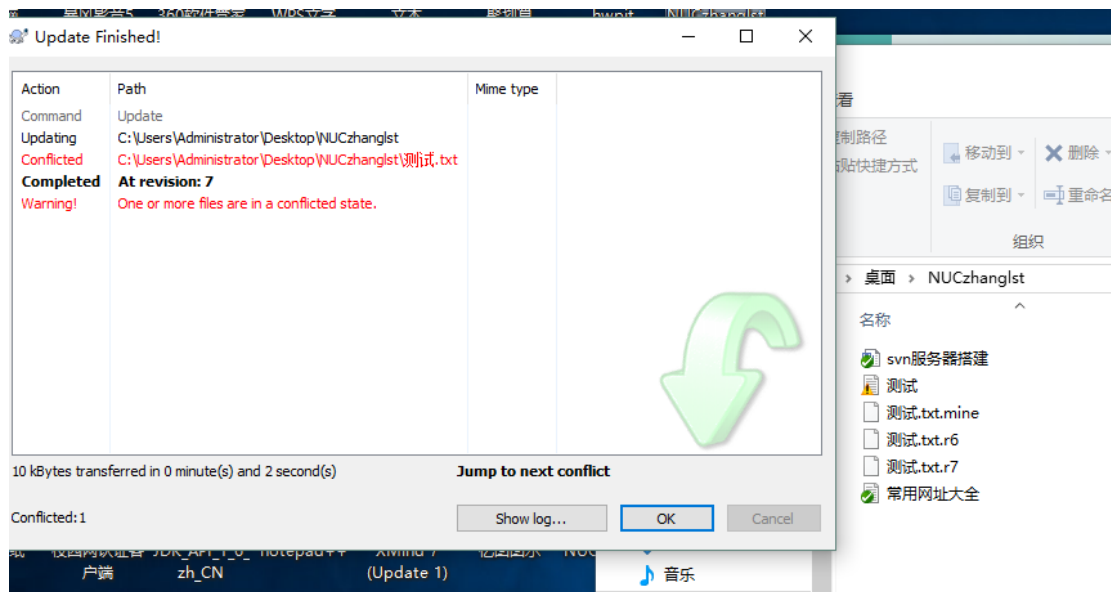
(1) 放弃提交内容，进行 revert 操作，更新到最新版本后再提交。



(2) 使用别人的更新。使用最新获取的版本覆盖目标文件，执行 resolved filename 并提交(选择文件—右键—解决)。



(3) 手动解决冲突文件。



冲突后在当前目录进行 updata 操作，会出现 3 个新的文件夹，

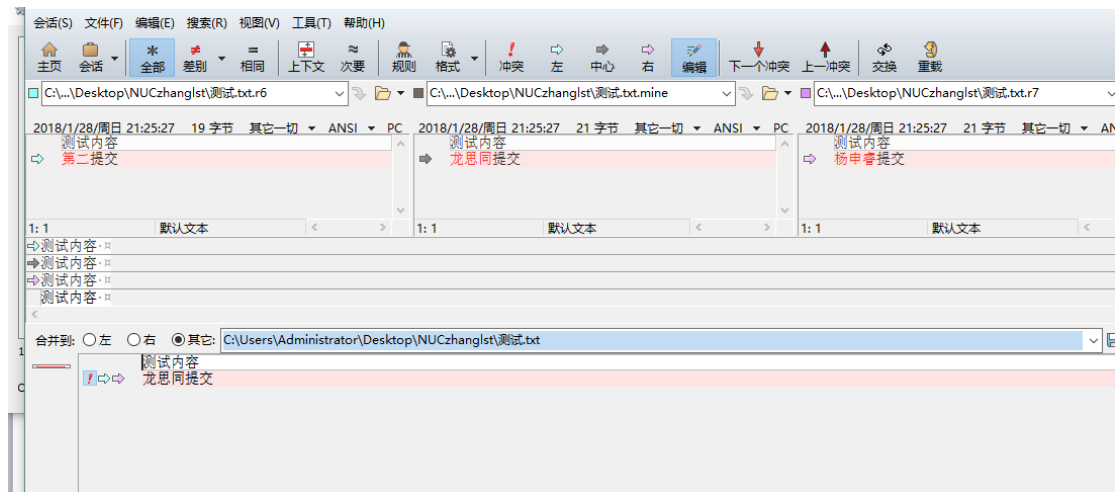
.mine 是冲突前自己的文件

.r6 是冲突前本地的版本文件

.r7 是其他用户赶在你之前提交的版本

Command	Update	
Updating	C:\Users\Administrator\Desktop\NUCzhanglst	
Conflicted	C:\Users\Administrator\Desktop\NUCzhanglst\测试.txt	
Completed	At revision: 7	
Warning!	One or more files are in a conflicted state.	

点击错误信息，就会进入 compare 比较页面进行比较，红色表示冲突的部分。



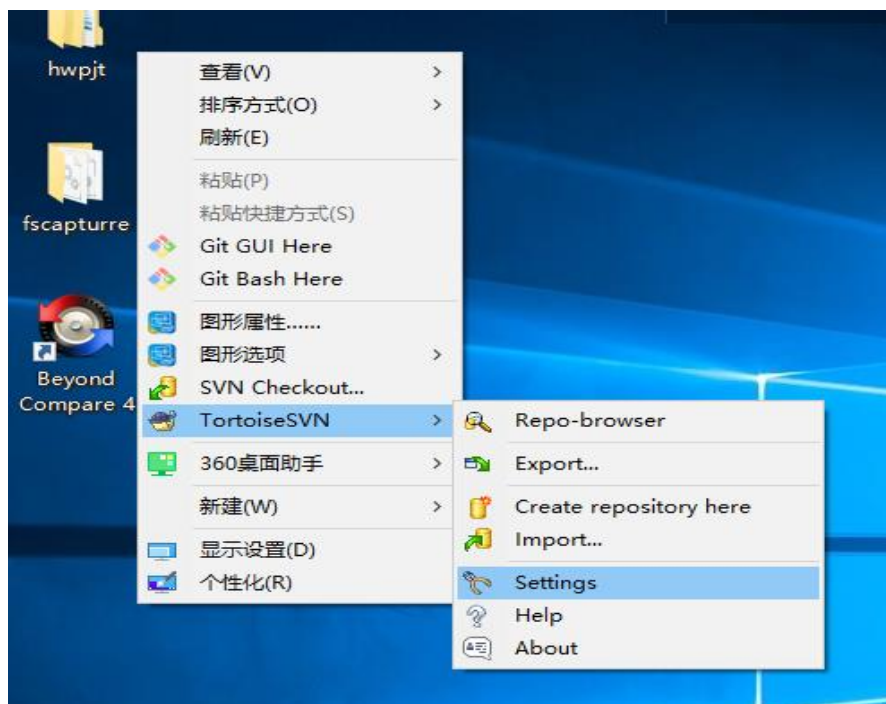
将红色的冲突改为一样的即可解决冲突。保存文件进行提交即可。

### 7.3. 选择比较工具

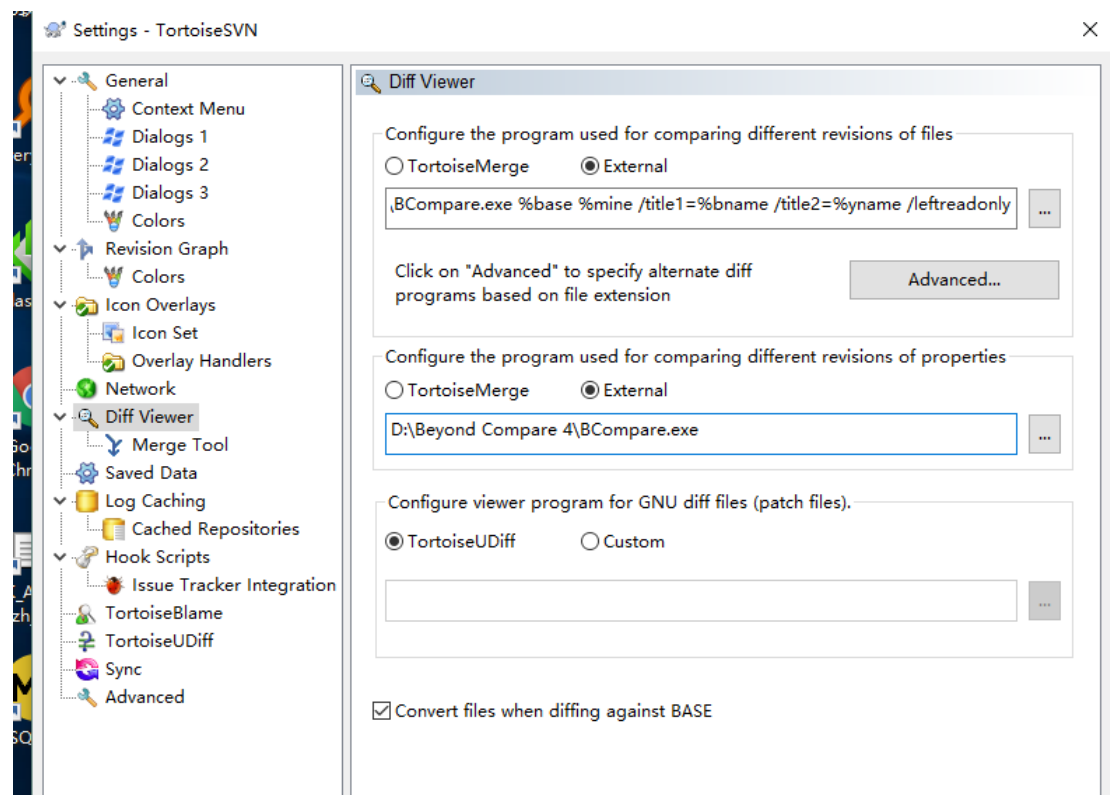
进行比较时，可以用 SVN 自带的比较工具，也可以更改为其他比较工具，如 Beyond Compare，操作流程如下：

Step1. 下载 Beyond Compare 比较工具。

Step2. 桌面右键>> TortoiseSVN>> settings。

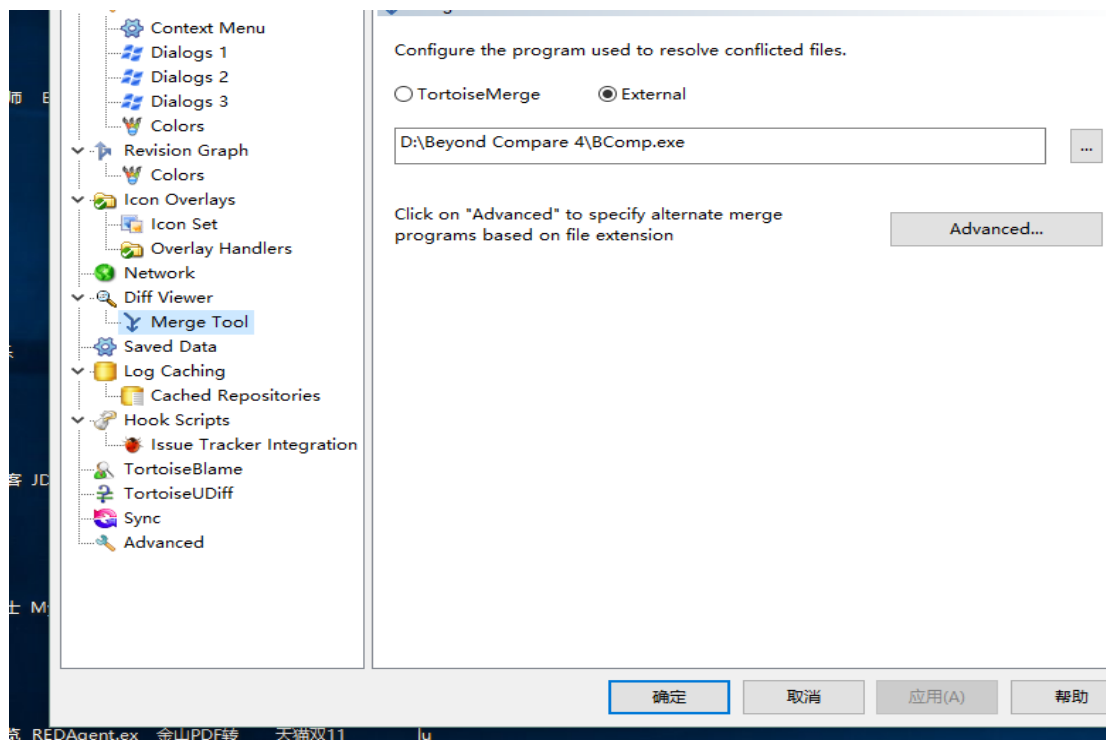


Step3.点击 diff viewer , 并将前两项选为 external。



第一个为 BCompare.exe 的安装路径 , 并且添加参数 %base %mine /title1=%bname /title2=%ynname /leftreadonly ;

第二个为 BCompare.exe 的安装路径。



点击 merge tool，选择 BCompare.exe 的安装路径，点击确定保存即可。

## 8 锁定/解锁

### 8.1. 场景

在 SVN 中，默认情况下任何文件都是没有锁定的，使用者可以随时对任何文件进行修改提交。如果你对某一文件进行锁定，那么只有你有权限对此文件进行提交，其他用户的提交都会被拒绝，一直到你对此文件释放锁。除此之外，其他用户也不能对文件进行更名甚至删除。

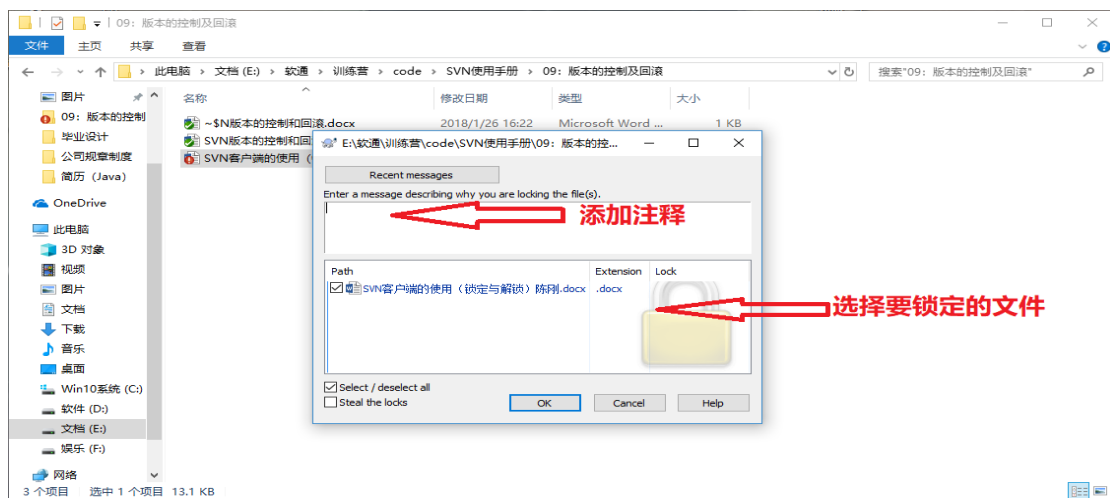
### 8.2. 操作流程

Step1.文件右键 TortoiseSVN >>Get lock ( 如果对一个文件夹执行此操作，则可锁定其中的所有项 )。





Step2.在出现的对话框中可以添加日志信息，用来告诉其他成员，你为什么要锁定这个文件。



Step4：单击 ok，此时这个文件已经被锁定。

## 9 版本控制及回滚

### 9.1. 场景

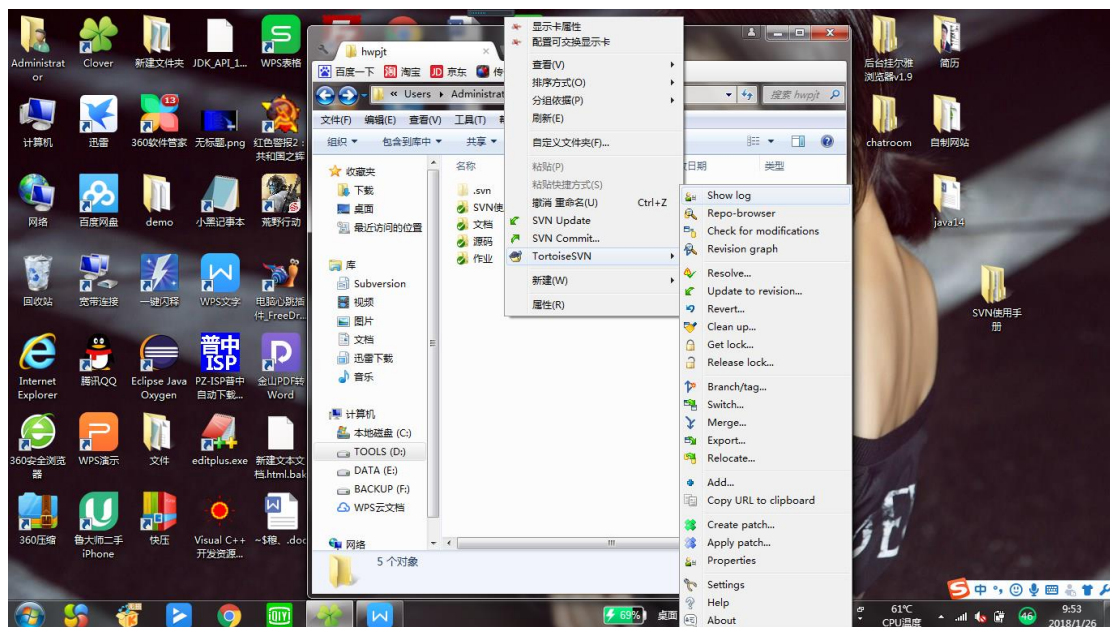
在代码编写的过程中，难免会出现一些错误，这样就涉及到版本的追踪，如果你之前提交的版本日志写的十分明确，则不太会出现什么问题。

(虽然在本文档中提到了版本回滚的方法，但在实际中使用版本回滚会导致很多问题，因此不建议大家经常使用这个功能)

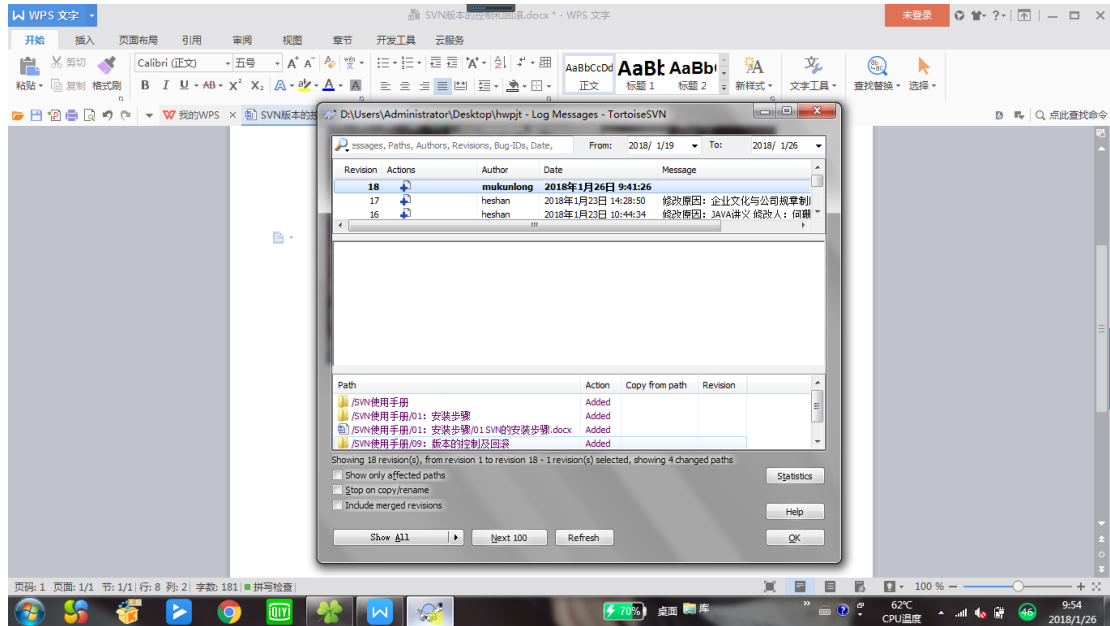
### 9.2. 操作流程：

#### 9.2.1. 方法一

Step1.TortoiseSVN>>Show log



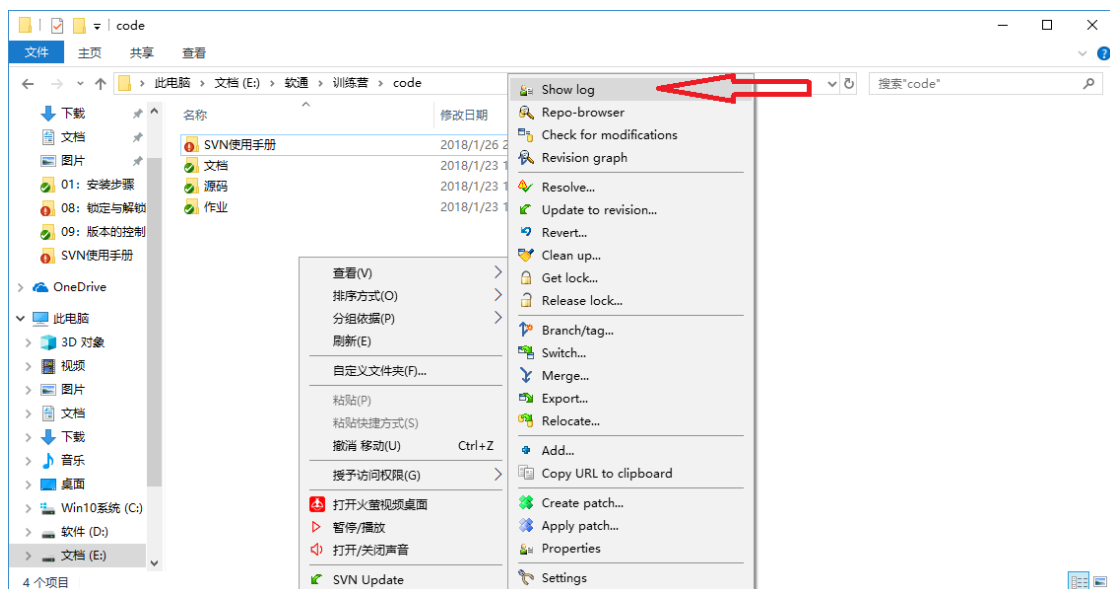
Step2：选中需要回滚的版本



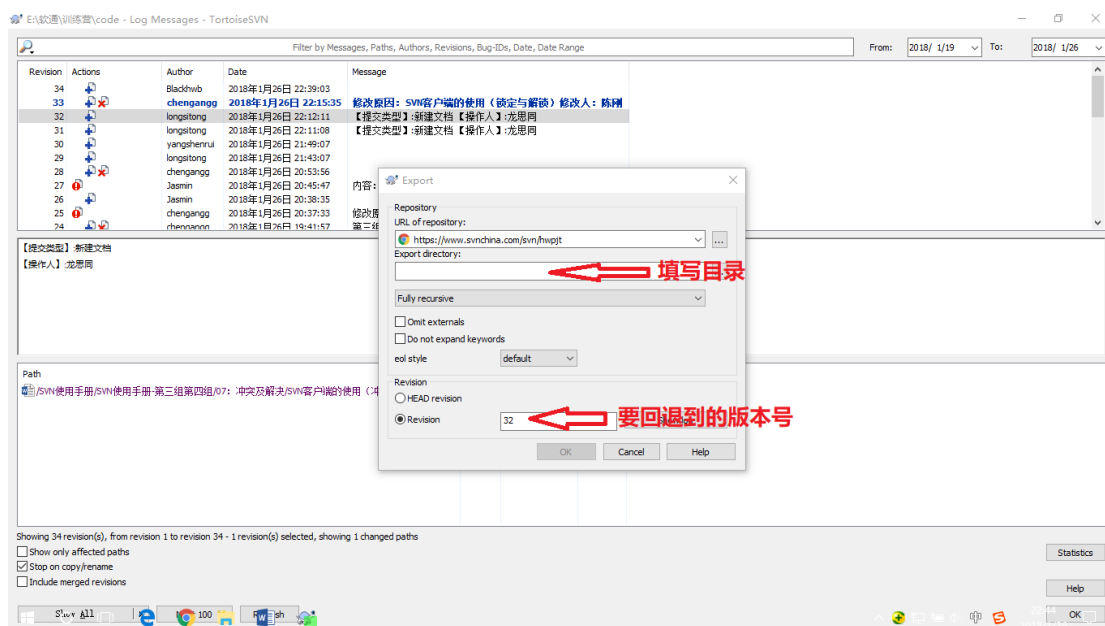
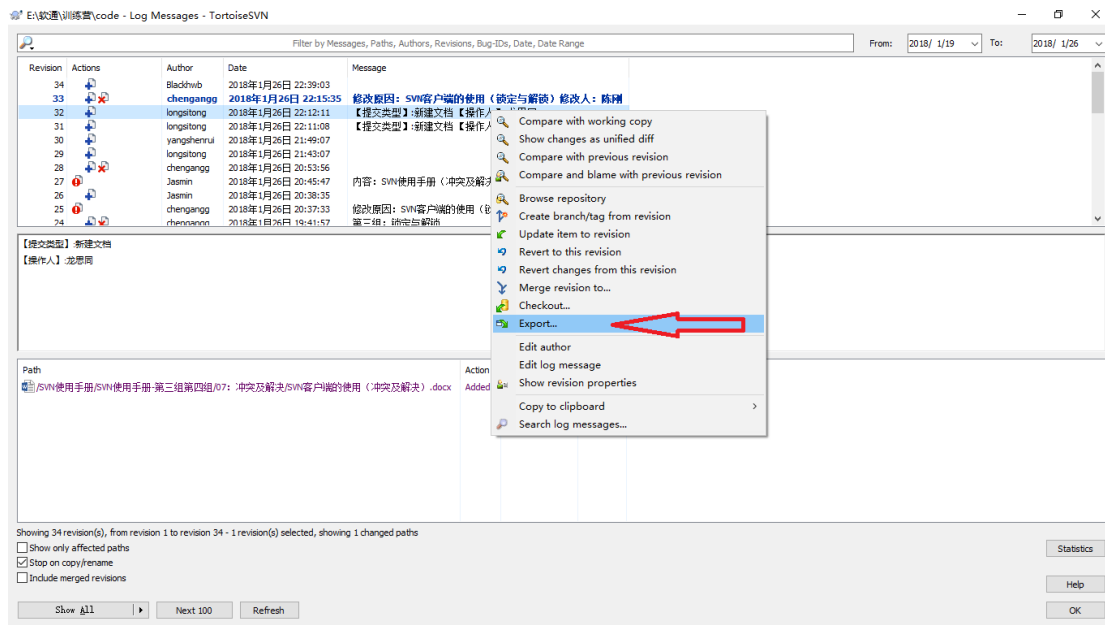
Step3.右键 Export , 然后 Commit 即可。

## 9.2.2. 方法二

Step1：在文件夹空白处 TortoiseSVN>>Show log。



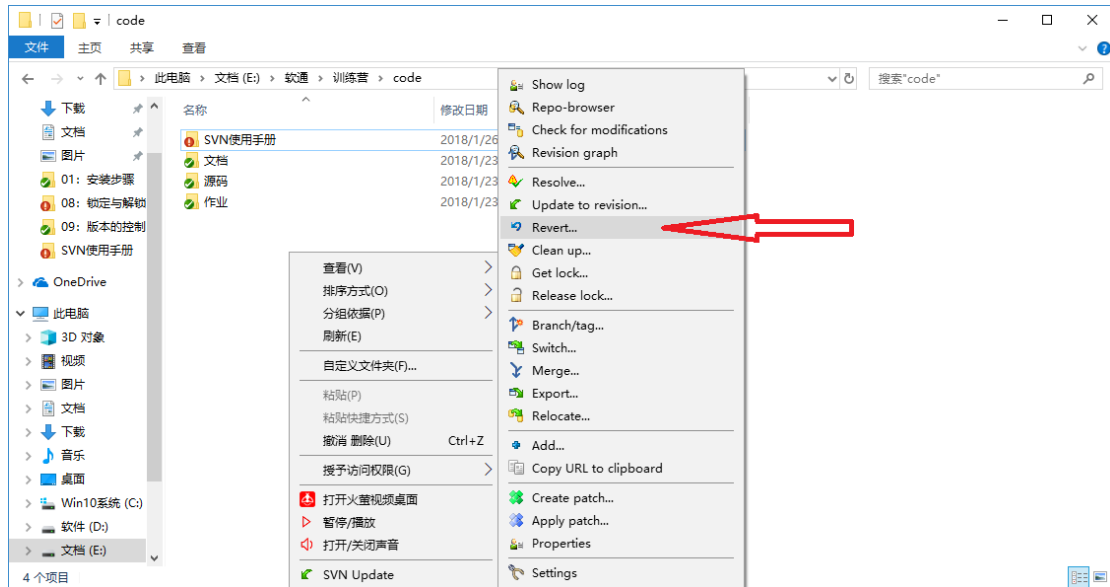
Step2：选择需要回滚的版本，右键—>Export,填写导出目录，点击 ok。



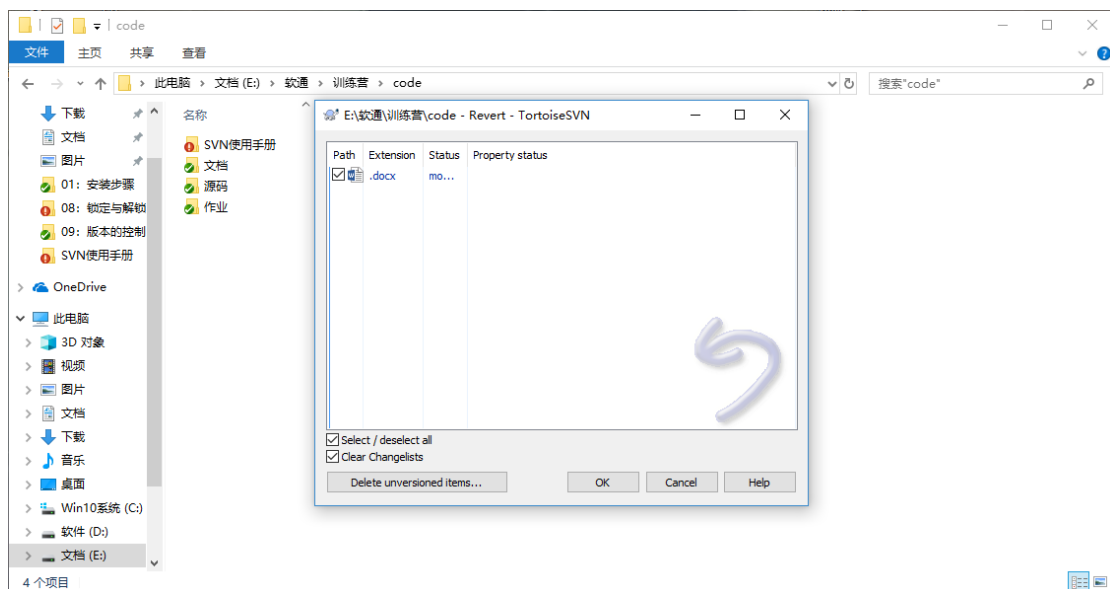
Step3.将你修改的文件覆盖到最新版本，然后右键 SVN Commit。此时版本已回退，并且你新建的文件也没有丢失。

### 9.2.3. 方法三

Step1.当你还没有提交时，想要放弃这些修改，在文件或目录空白处右键  
TortoiseSVN>>Revert。

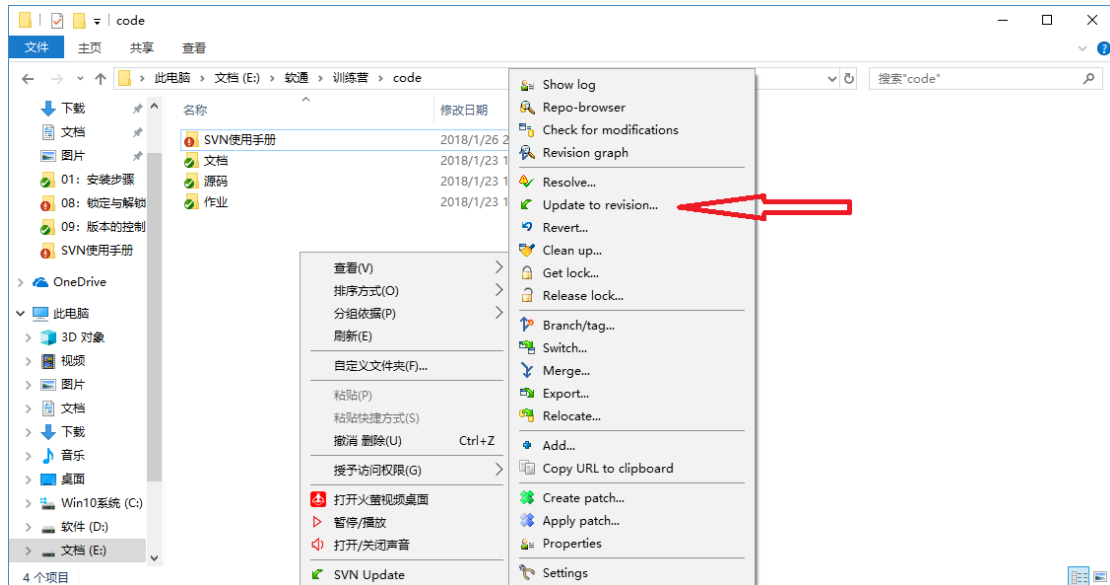


Step2.点击 ok，此时你的版本回退到了工程的最新版本。

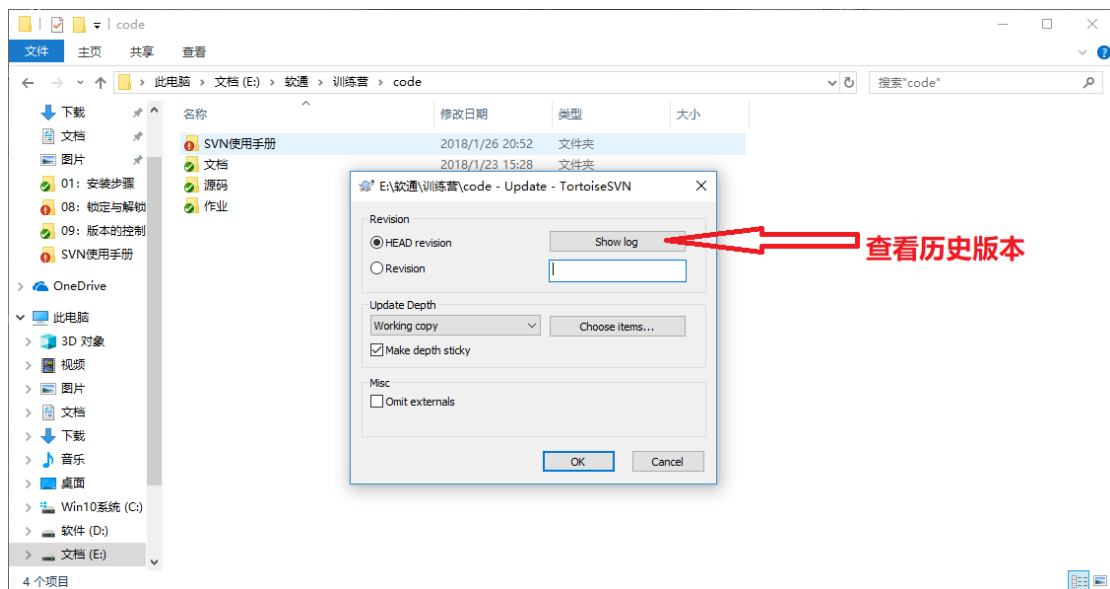


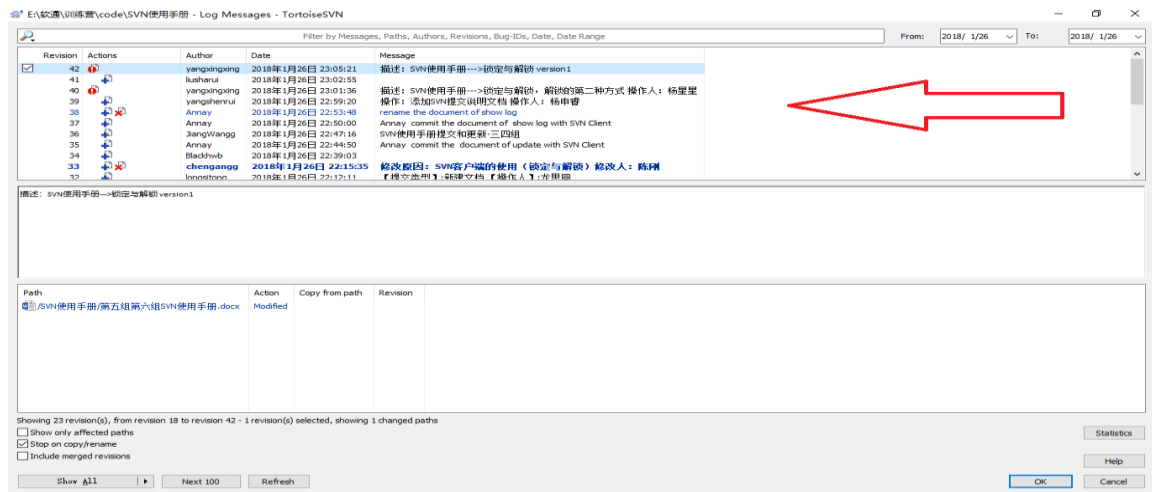
## 9.2.4. 方法四

Step1:右键 TortoiseSVN>>Update to reversion

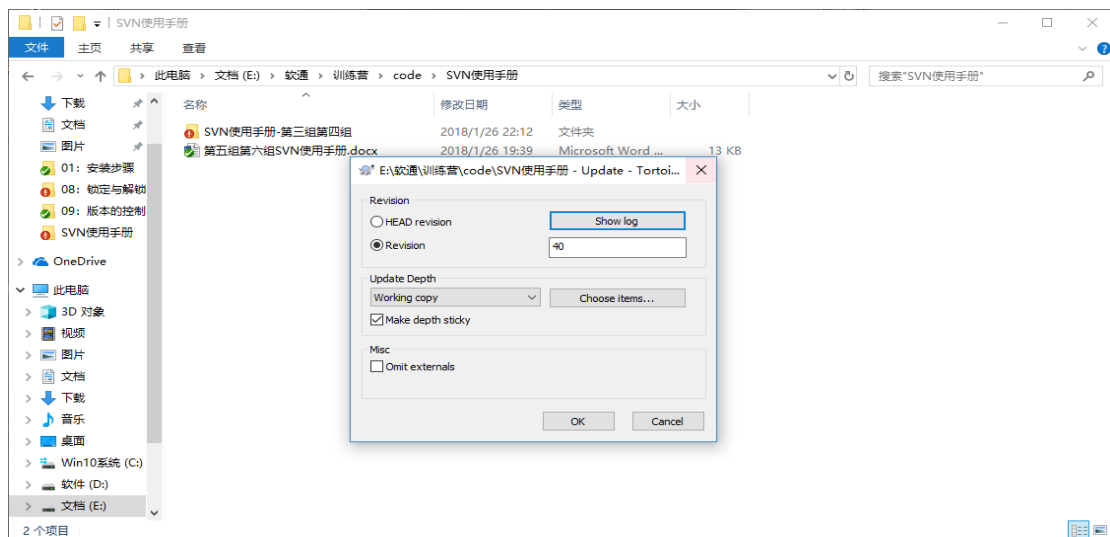


Step2.点击 show log，选择想要回退到的版本，点击 ok。



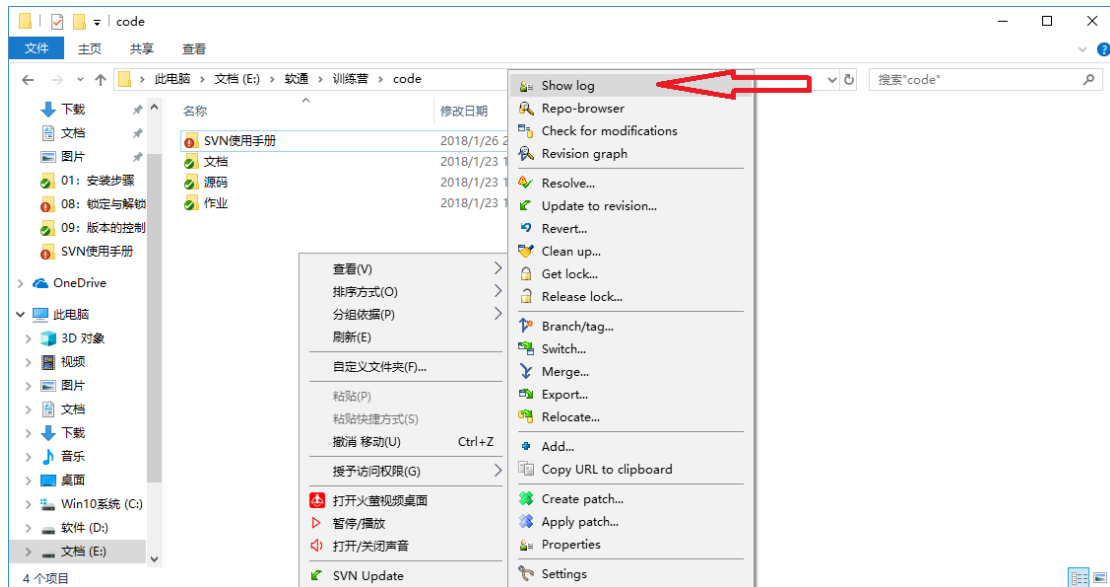


Step3.再次点击 ok，此时版本已经回退到你选择的版本，右键 SVN Update 并且解决冲突，这时你便可以在此版本上进行编辑。

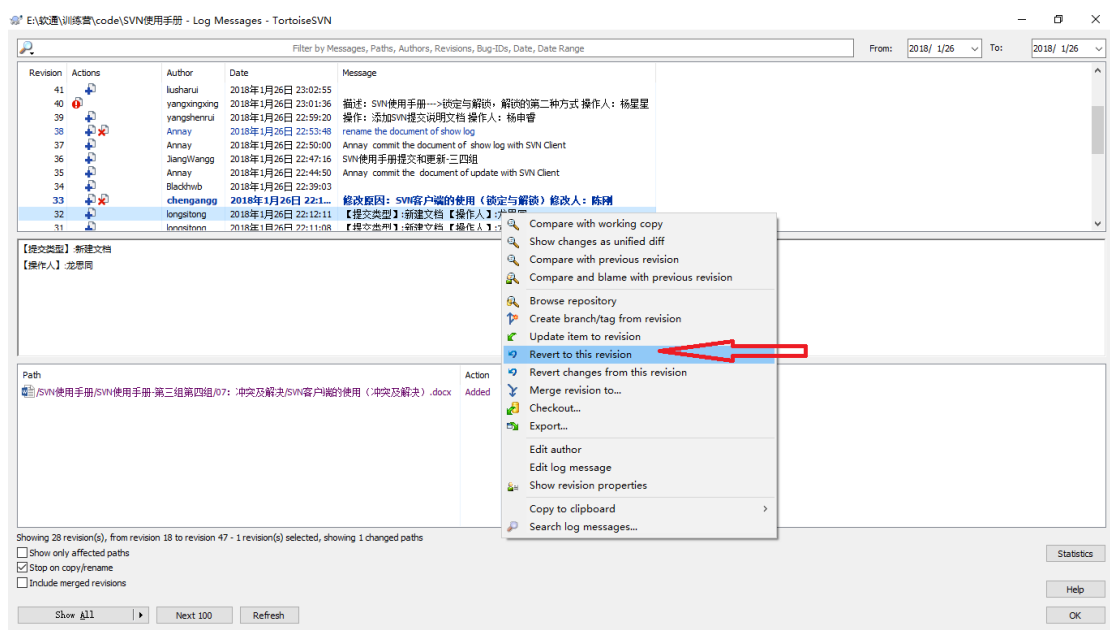


## 9.2.5. 方法五

Step1.若想直接回退到之前的某个版本，右键 TortoiseSVN>>Show log。

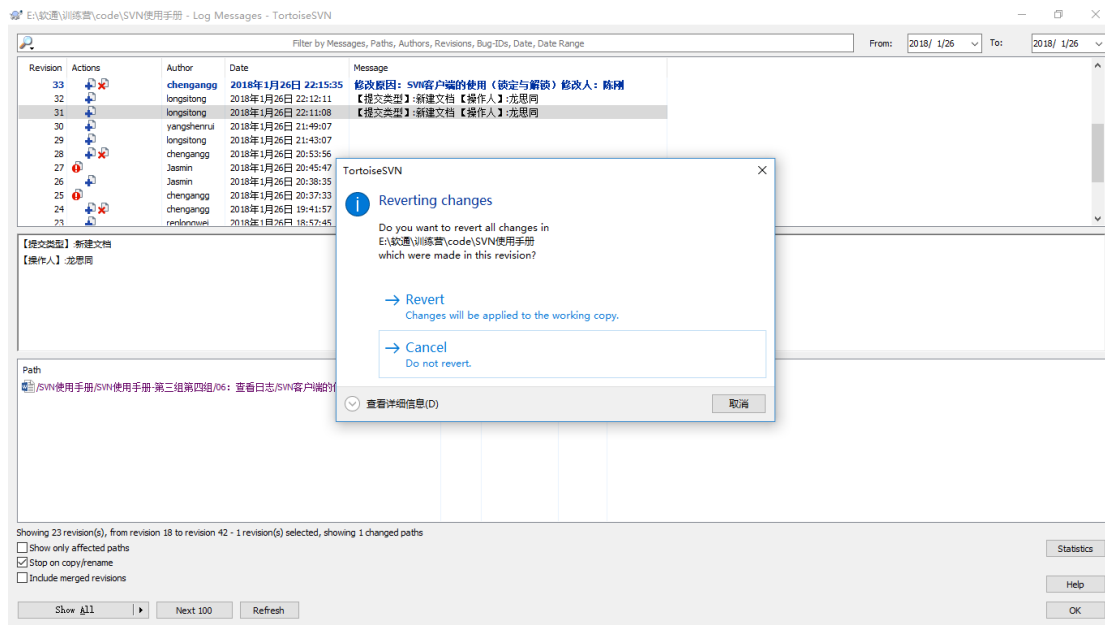


Step2.在想要回退的版本处右键，选择 Revert to this revision



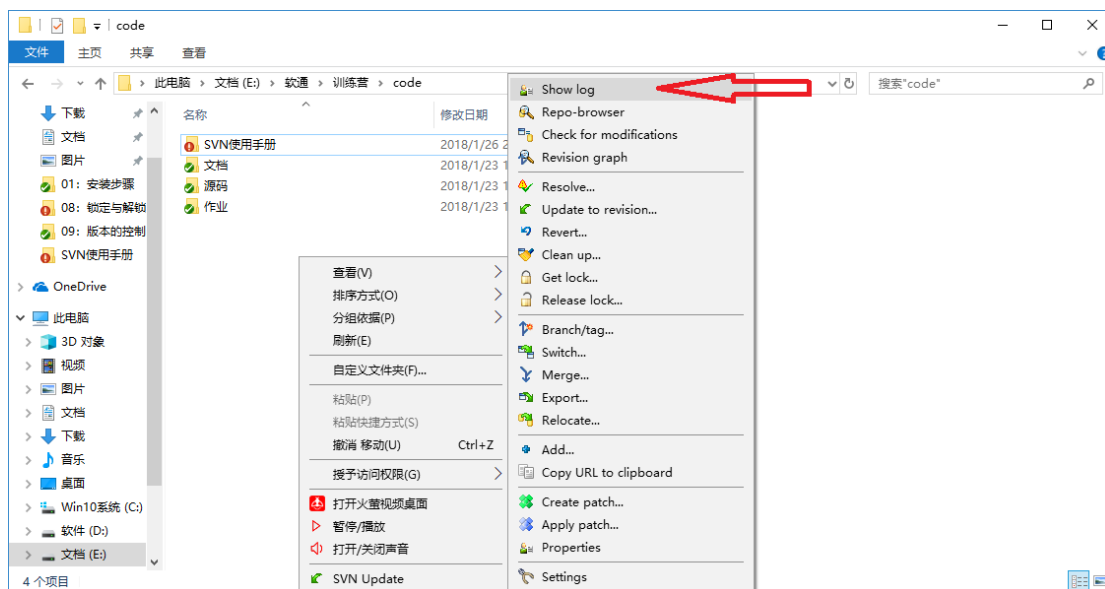
Step3.点击 Revert。此时你的版本已经回退到你选择的版本。



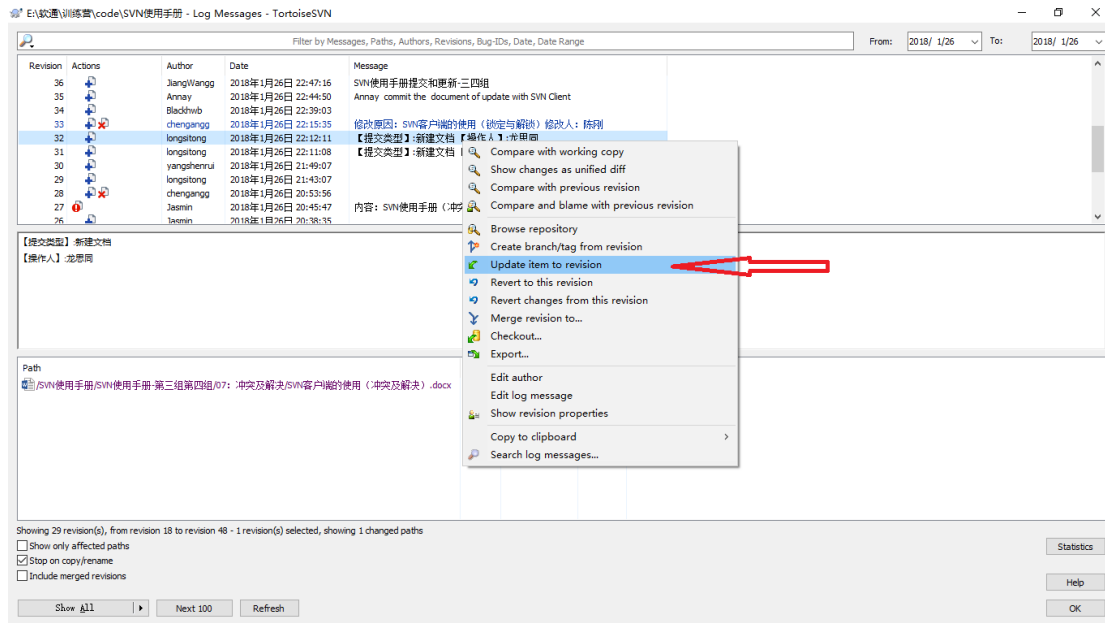


## 9.2.6. 方法六

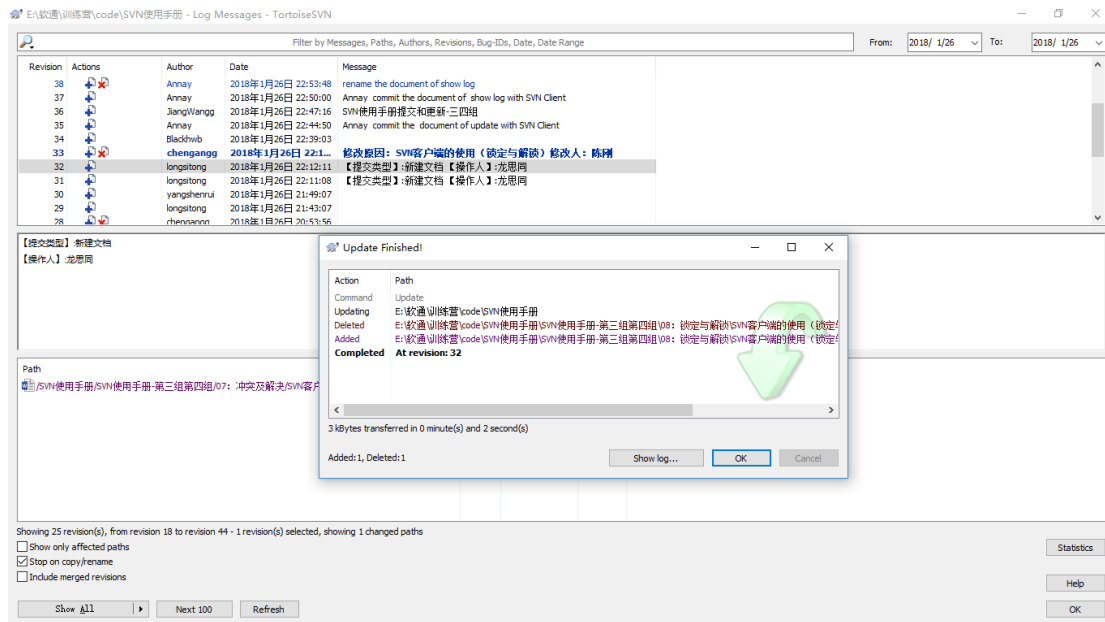
Step1.在文件夹空白处 TortoiseSVN>>Show log。



Step2.在想要会退的版本处右键，选择 Update item to reversion



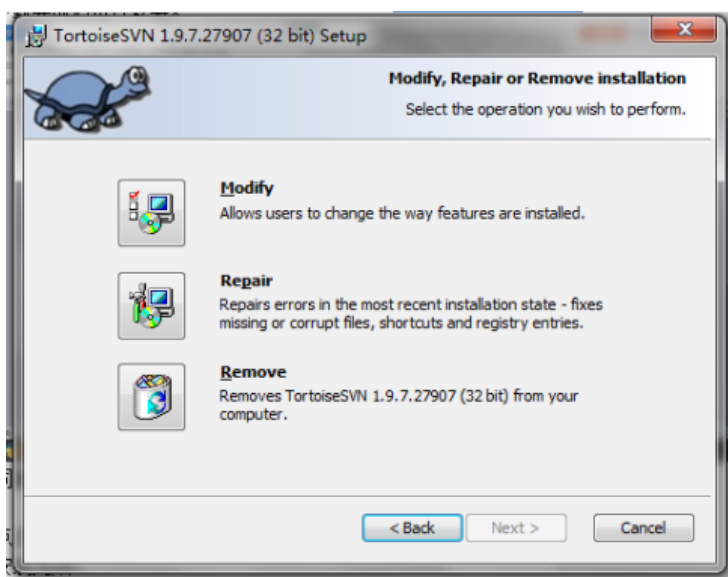
Step3.点击 ok，此时版本已经回退到你选择的版本，右键 SVN Update 并且解决冲突（若不执行，则此版本不能编辑），这时你便可以在此版本上进行编辑。



## 10 svn 使用注意事项

### 10.1. 安装

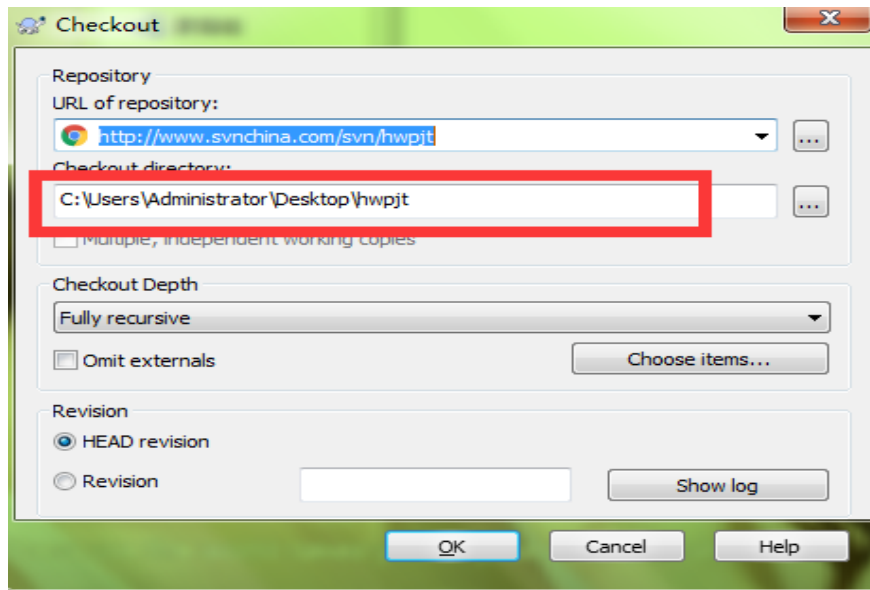
SVN 为常用的版本控制器之一，其安装步骤非常简单。下载好后直接 install 进行安装，一步步直接 next 就好。在安装出错后再次进行安装事，会有三个选项供选择，如图所示：



其中，modify 为改变安装路径等；repair 为修复 SVN；Remove 为卸载，移除。

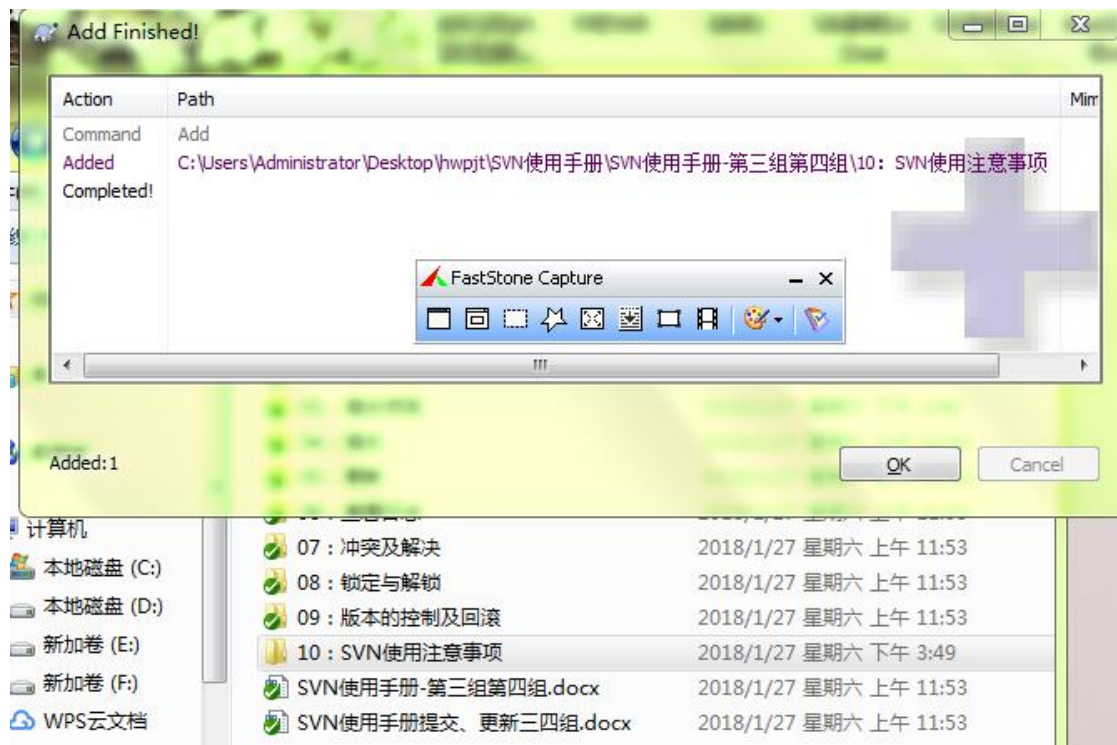
### 10.2. 检出

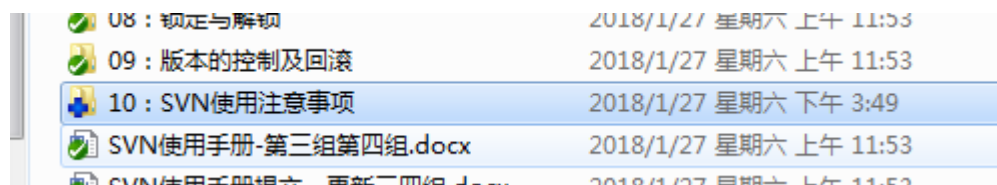
检出时注意选择好自己的检出路径，自己指定一个特定的文件夹，方便日后查找使用。



### 10.3. 导入

导入又可以叫做添加或者新增，应该先 Add,待其变为蓝色加号后再选中右键 cmmit 进行提交。提交后，有时反应较慢需要刷新下，由蓝色图标变为绿色对勾，则为正常。





U8: 锁定与解锁	2018/1/27 星期六 上午 11:53
09: 版本的控制及回滚	2018/1/27 星期六 上午 11:53
10: SVN使用注意事项	2018/1/27 星期六 下午 3:49
SVN使用手册-第三组第四组.docx	2018/1/27 星期六 上午 11:53
SVN使用手册-第三组第四组.docx	2018/1/27 星期六 上午 11:53

## 10.4. 提交

提交前一定要注意更新操作,因为有可能在你自己编写代码的这段时间有人对同一个文件做了操作。如果不更新,就会造成冲突。当然,如果对方使用了锁,你则无法提交,直到对方释放锁,你才可以对此文件进行操作。

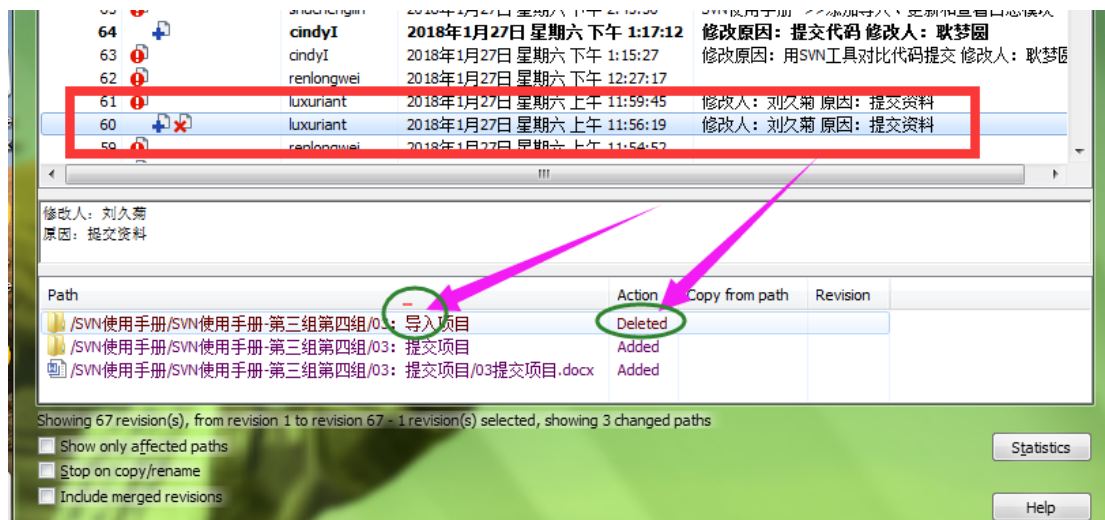
为此,我们每天上班开工的时候,第一件事都应该是从 SVN 上更新最新的代码下来,这样你那个有效的较少冲突的发生。

## 10.5. 更新

更新有时代码量较大,或者你长时间不更新。SVN 一次性更新时间较长,在你更新的同时有可能另一边有人提交同时发生,所以有时一次性更新不到最新的版本,需要多次更新。

## 10.6. 查看日志

查看日志的目的是为了方便用户解决冲突等异常情况,及时准确的定位到某某同事对那一个部分做了那一步操作,造成了什么后果,如下图:



## 10.7. 冲突与解决方案

冲突是我们每个人都不愿意看到的事情，但也是难以避免的，总有操作失误的时候，因此，遇到冲突，不要慌张，该用比较工具的用比较工具，该回滚的回滚，该删除了重新更新的重新更新。

一般情况下，当自己提交发生冲突后始终解决不了冲突，可以这样解决：

- ①：把自己写的内容备份一份放在外边的目录中；
- ②：选中起冲突的文件或文件夹，右键删除（注意不是从 svn 中删除，而是从本地删除）；
- ③：在本地删除文件的目录下右键更新。这时就会从 SVN 中检出你删除的 SVN 中的文件；
- ④：那比较工具比较先前自己备份的文件和刚从 SVN 中更新的文件作比较进行筛选剔除，最后记得保存。
- ⑤：在保存后的本地文件中。选中保存后的文件，commit 就提交成功了（这是望不得已的办法，也是最笨最可行的方法）

## 10.8. 锁定与解锁

在修改完文件后要记得释放锁，让其他同事有权限访问。

## 10.9. 版本的控制及回滚

在目前的版本无法解决冲突时，才需要回滚到上一版本操作。

## 10.10. 经验总结

目前为止：在 SVN 上提交代码，特别是前台，一个模块应该把整个模块的文件都提交（直接提交你所需要提交所需文件的上成文件夹），这样不会引起冲突报错。而耽搁提交会引起冲突报错。