



北京动力节点教育科技有限公司

动力节点课程讲义

DONGLIJIEDIANKECHENGJIANGYI

www.bjpowernode.com

SVN 讲义

第1章 SVN 简介及服务器安装

1.1 SVN 简介

1.1.1 为什么要使用 SVN

程序员在编写程序的过程中，每个程序员都会生成很多不同的版本，这就需要程序员有效的管理代码，在需要的时候可以迅速，准确取出相应的版本。

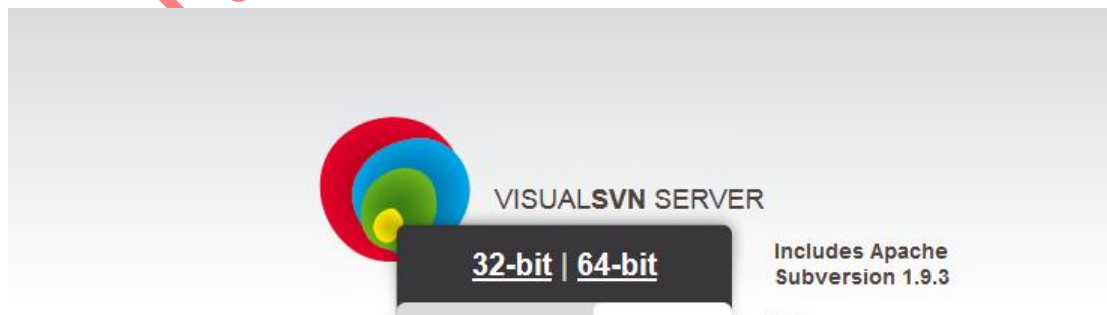
1.1.2 Subversion 是什么

它是一个自由/开源的版本控制系统，一组文件存放在中心版本库，记录每一次文件和目录的修改，Subversion 允许把数据恢复到早期版本，或是检查数据修改的历史，Subversion 可以通过网络访问它的版本库，从而使用户在不同的电脑上进行操作。

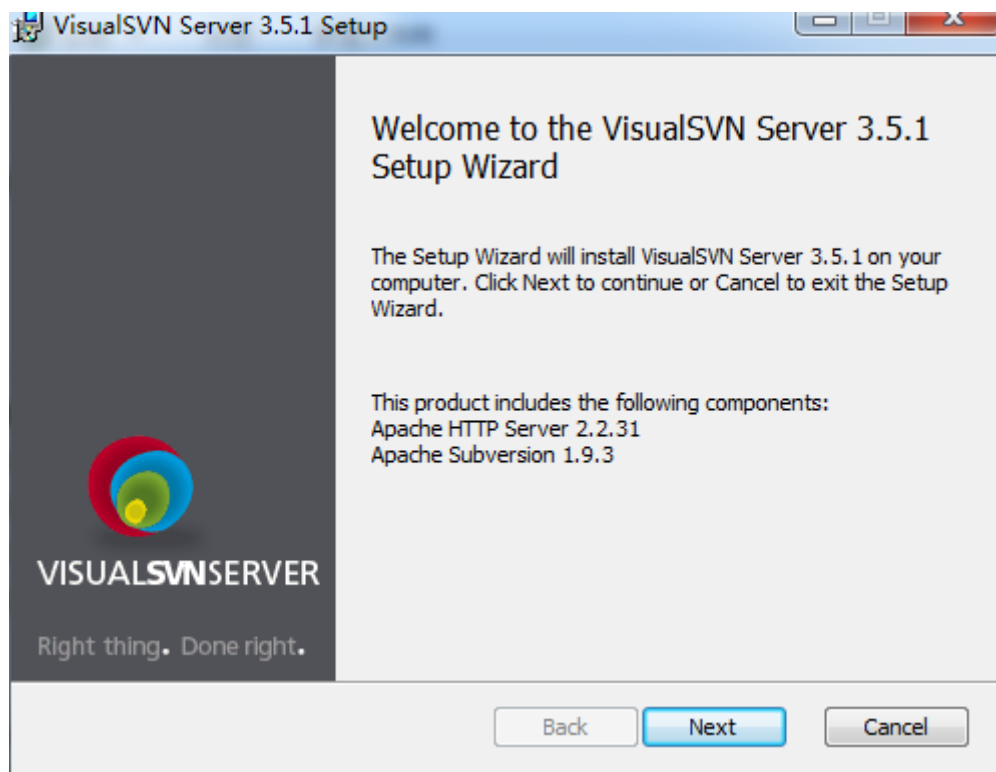
1.2 SVN 服务器搭建和使用

1.2.1 首先来下载和搭建 SVN 服务器

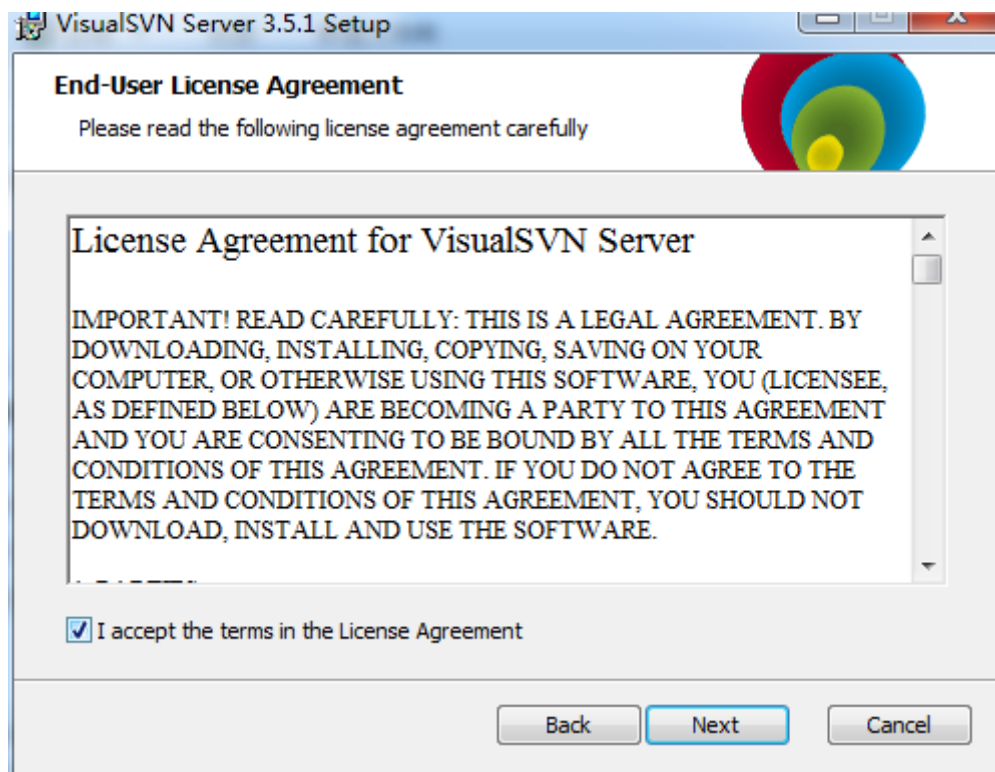
下载地址如下：<https://www.visualsvn.com/server/download/>，根据自己操作系统的版本选择合适的 SVN 版本进行下载



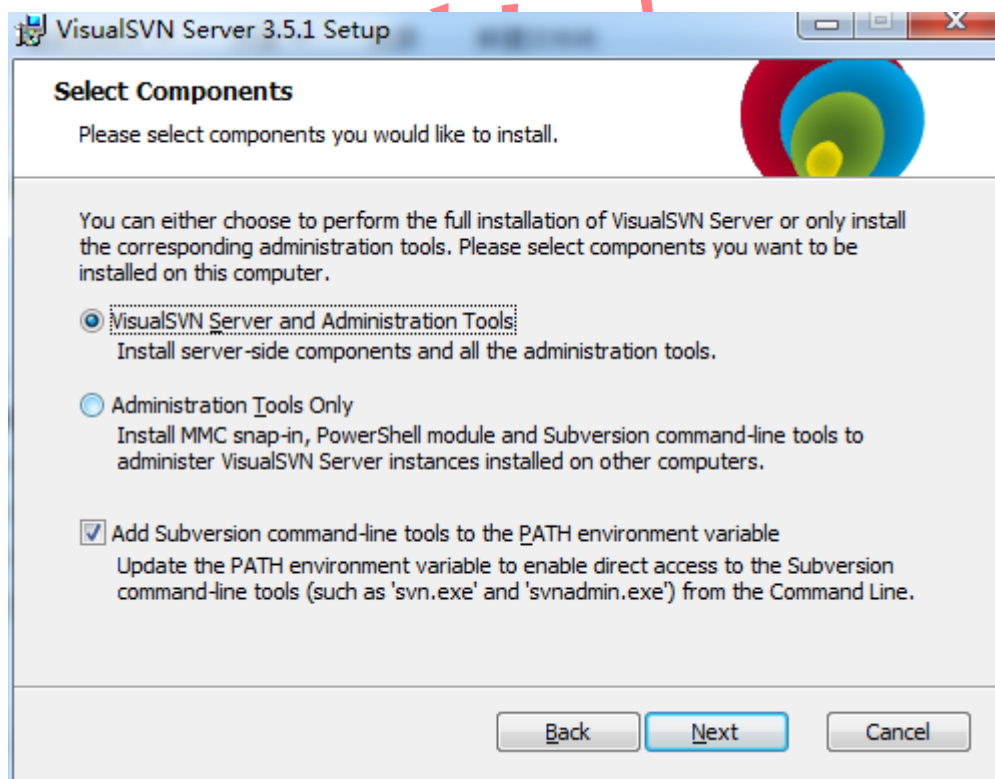
个人认为最好用 VisualSVN server 服务端和 TortoiseSVN 客户端搭配使用。 点开上面的 VisualSVN 连接, 下载 VisualSVN server, 下载完成后双击安装, 如下图:



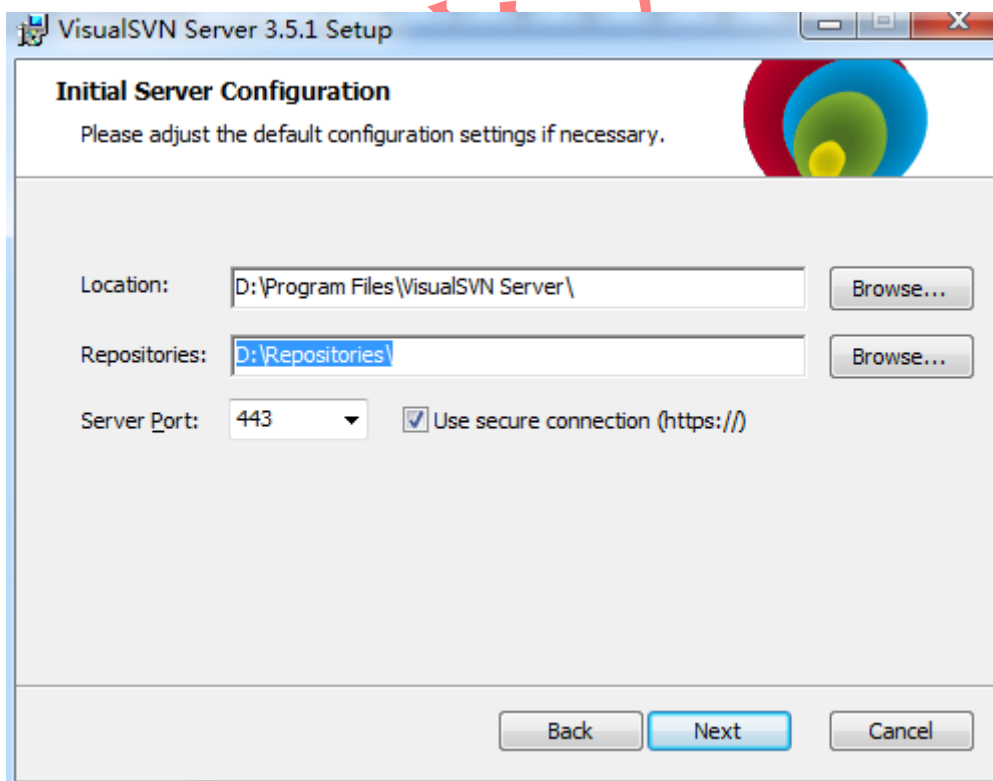
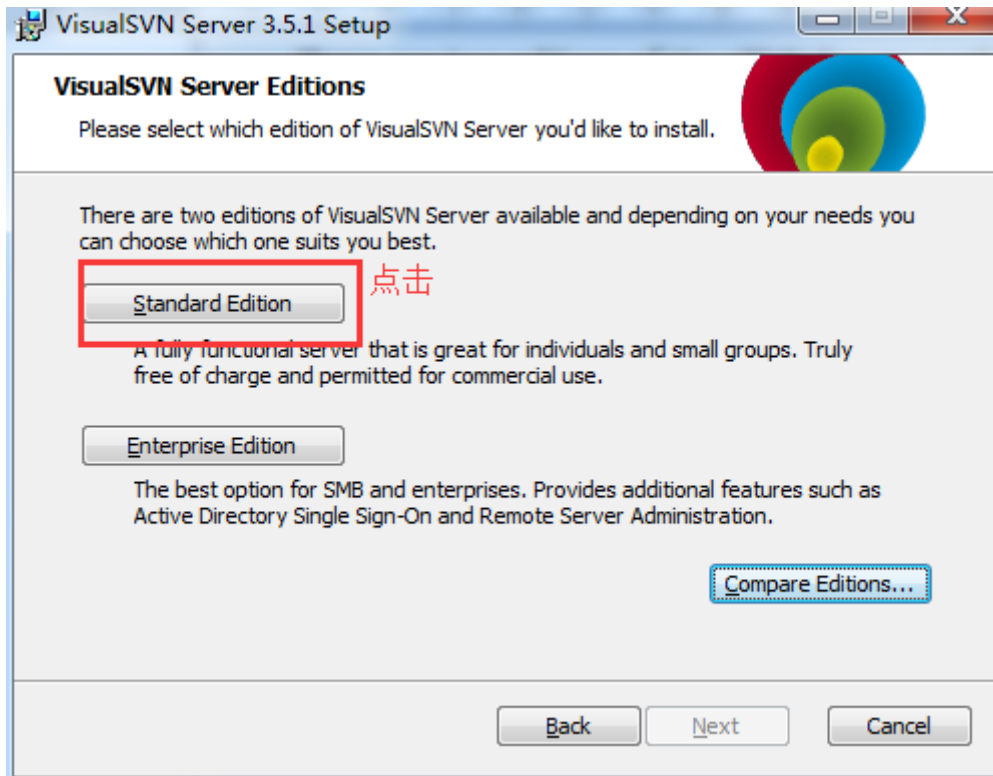
点击 Next 下一步, 如下:



然后再点击 Next 项，下一步，如下：

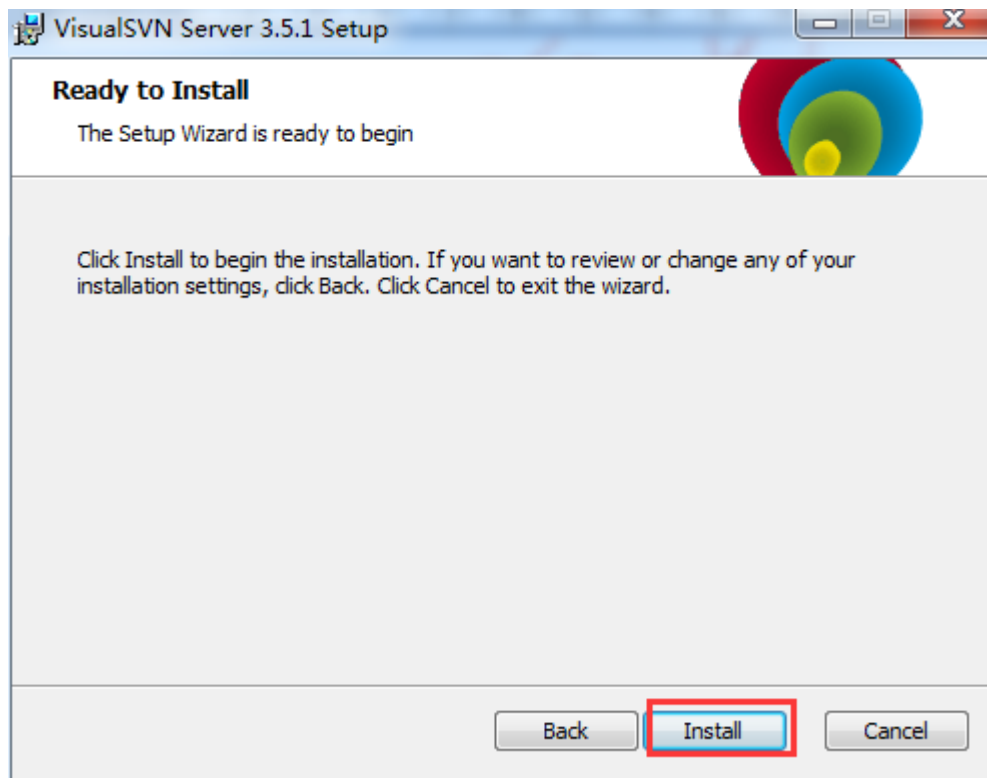


点击【Next】 如下：

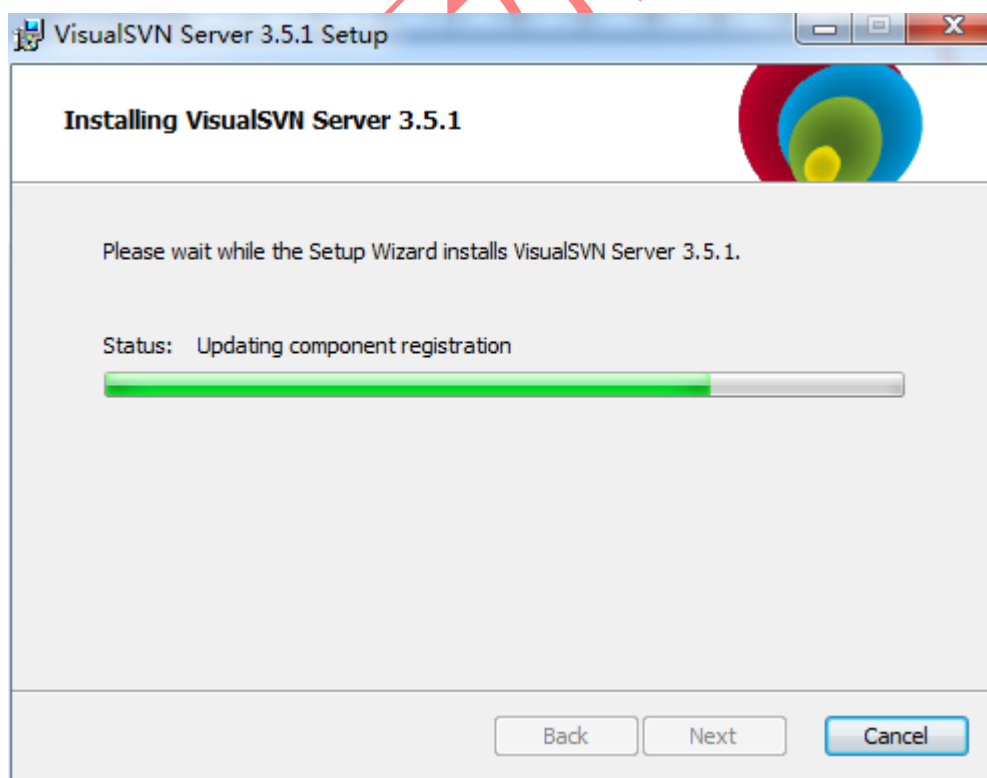


Location 是指 VisualSVN Server 的安装目录, Repositorys 是指定你的版本库目录. Server Port 指定一个端口, Use secure connection 勾选表示使用安全连接,

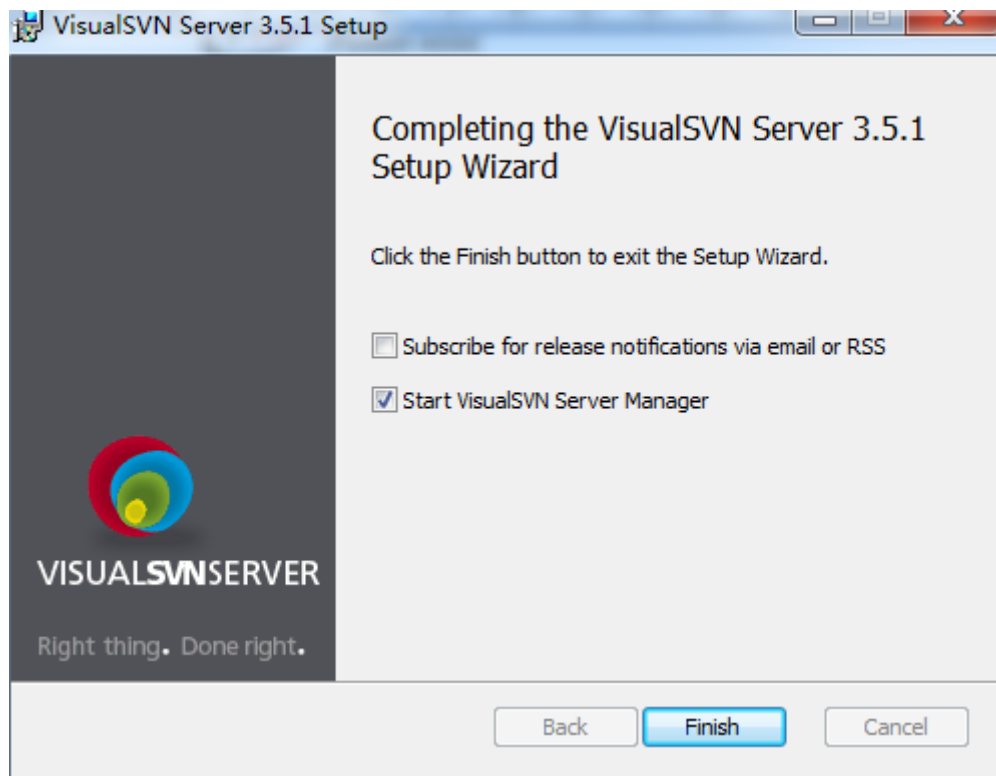
点击 Next, 进入下一步, 如下图:



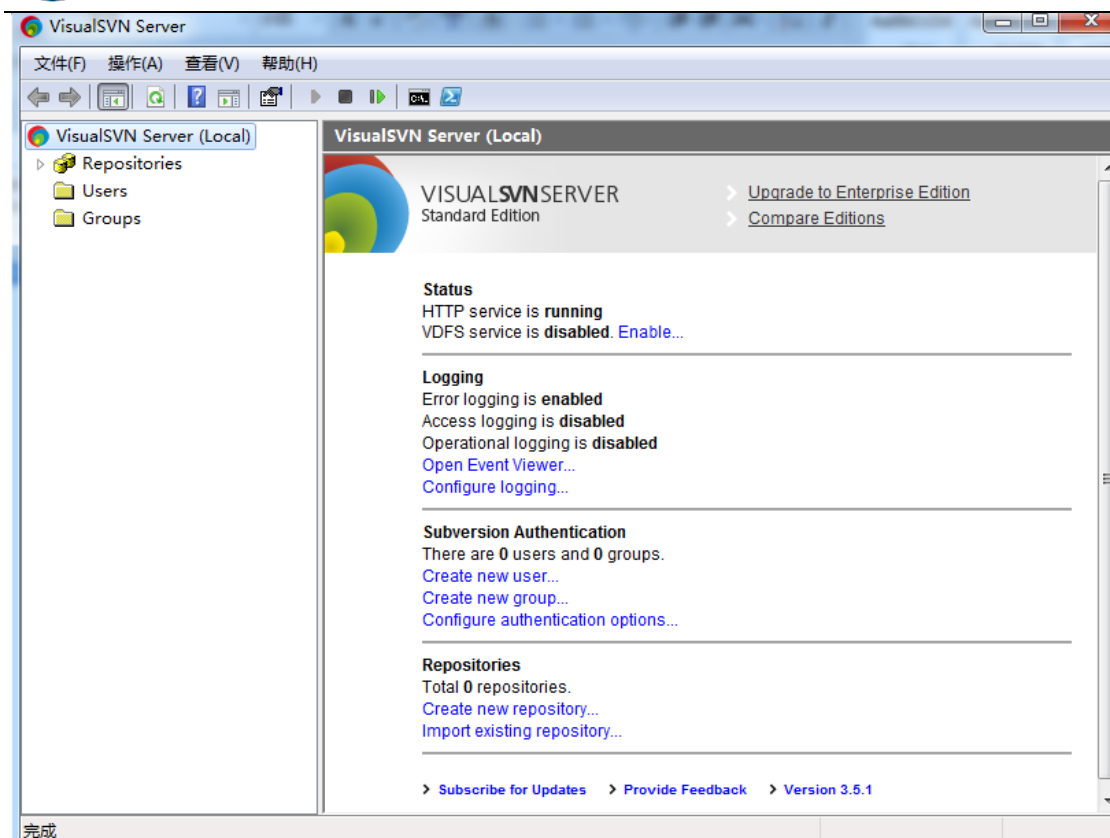
再点击【Install】, 进入如下安装图:



等待安装完成后，点击【next】，进入下一步：如下图



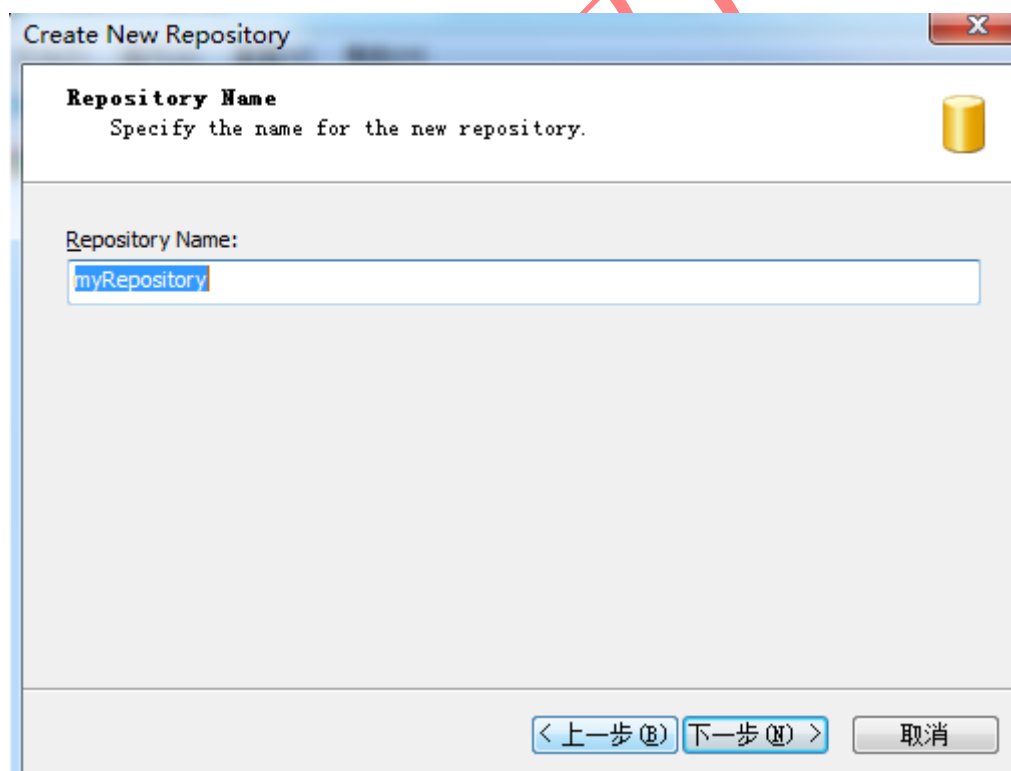
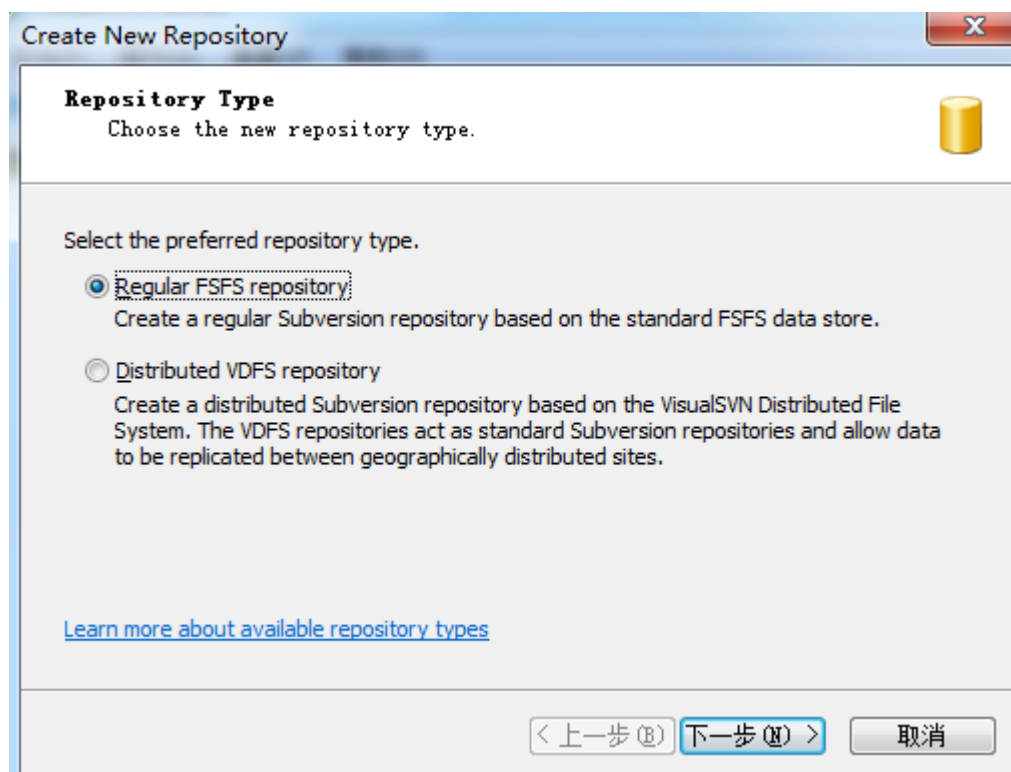
点击【Finish】即可完成安装。安装完成后，启动 VisualSVN Server Manager，如图：



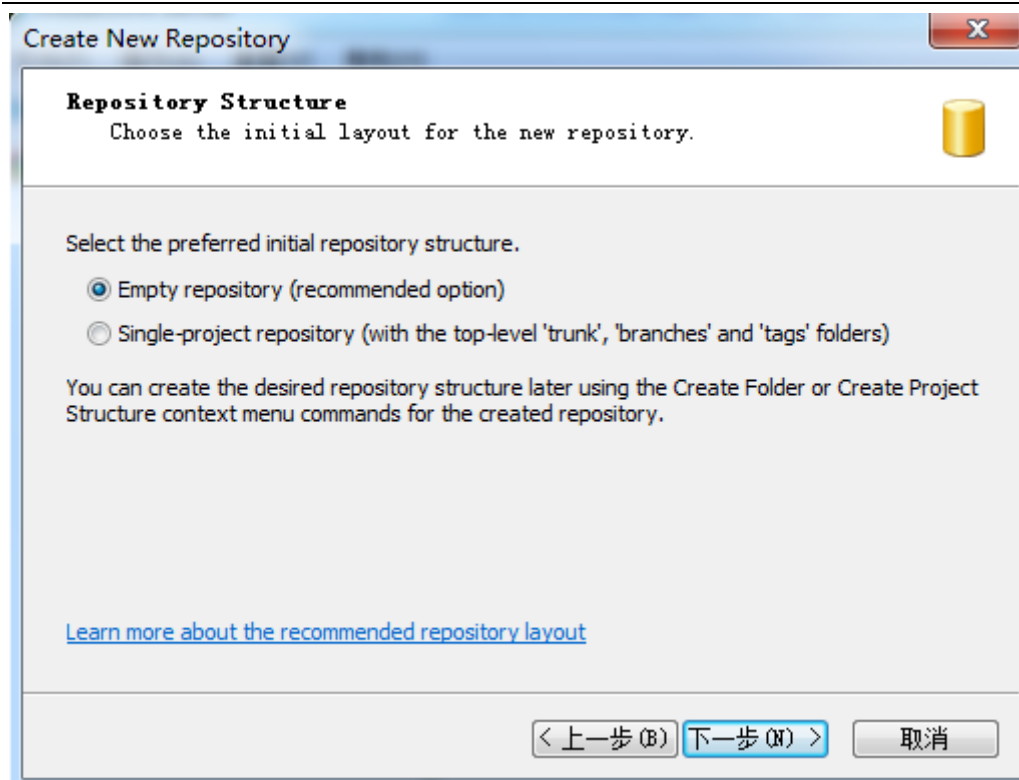
可以在窗口的右边看到版本库的一些信息, 比如状态, 日志, 用户认证, 版本库等.
要建立版本库, 需要右键单击左边窗口的 Repositories, 如下图:



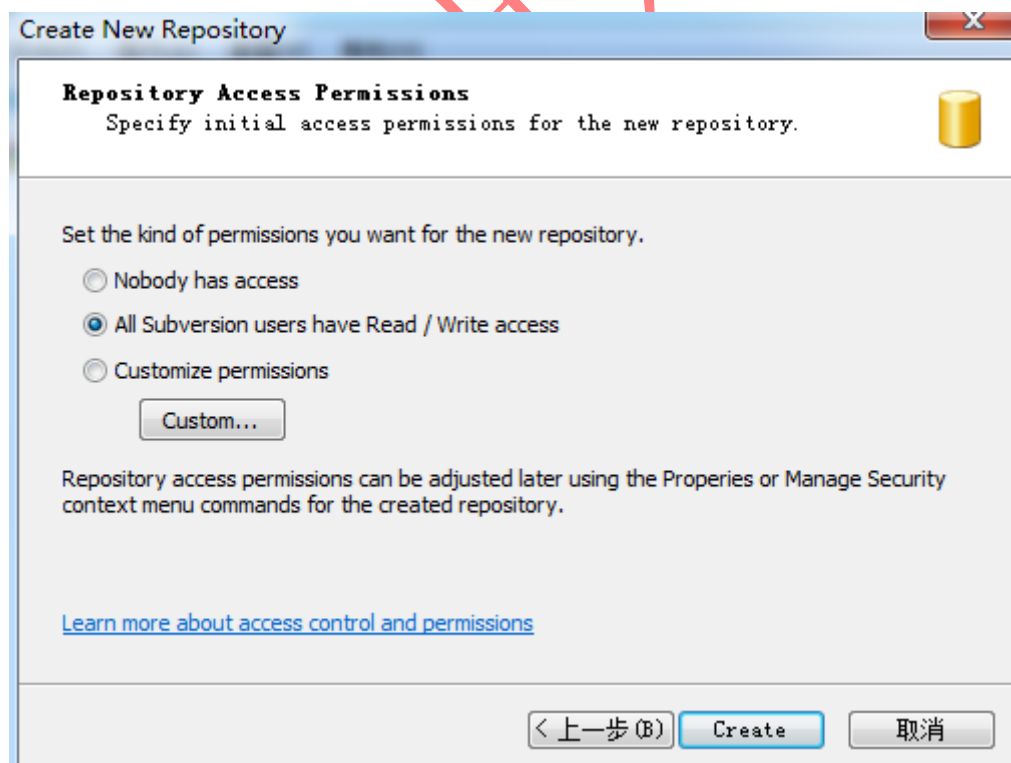
在弹出的右键菜单中选择 Create New Repository 或者新建->Repository:



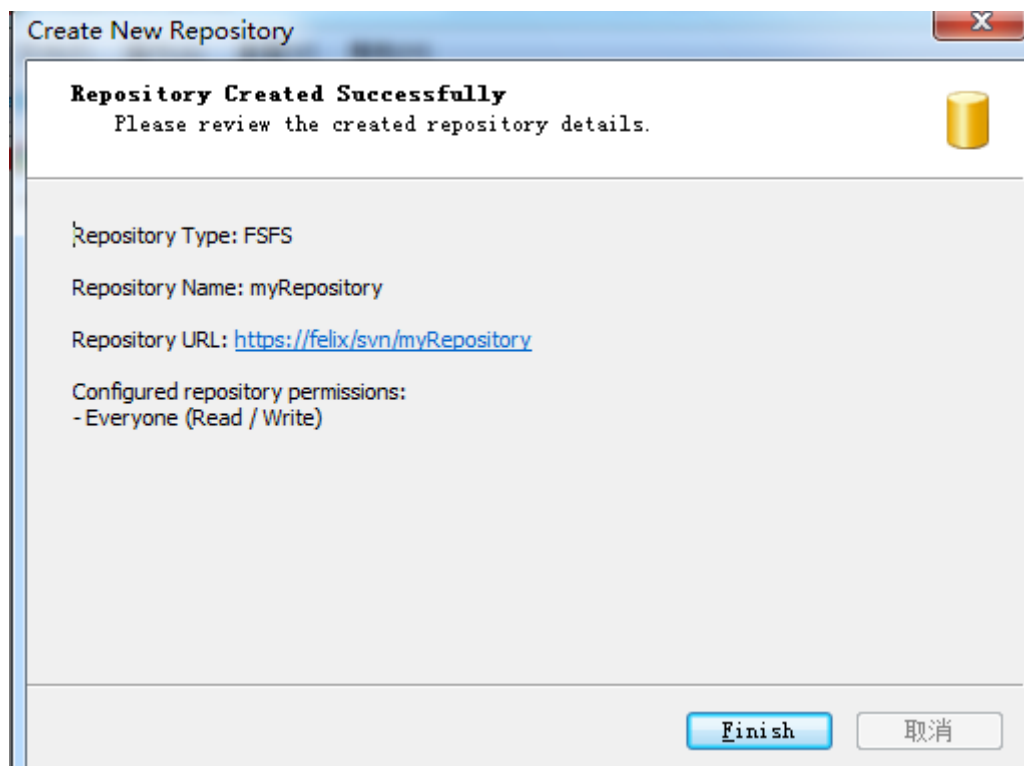
进入下一步，如下图：



点击【下一步】，如下图：



点击【create】，如下图：



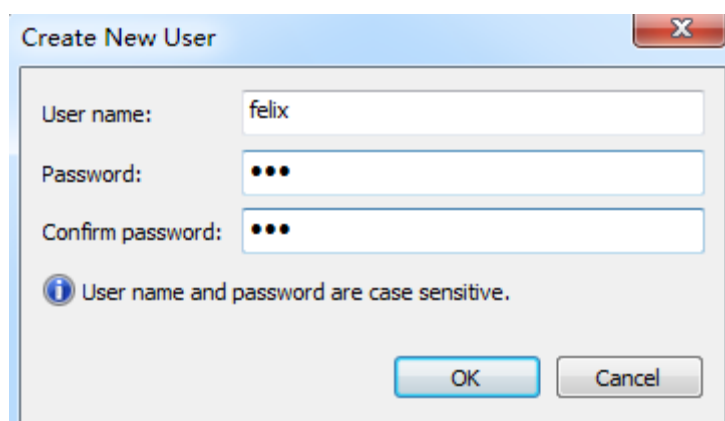
点击【Finish】即可完成基本创建。

1.2.2 需要建立用户

在 VisualSVN Server Manager 窗口的左侧右键单击 Users, 选择 Create User 或者新建->User, 如图:



点击 User 后，进入如下图：



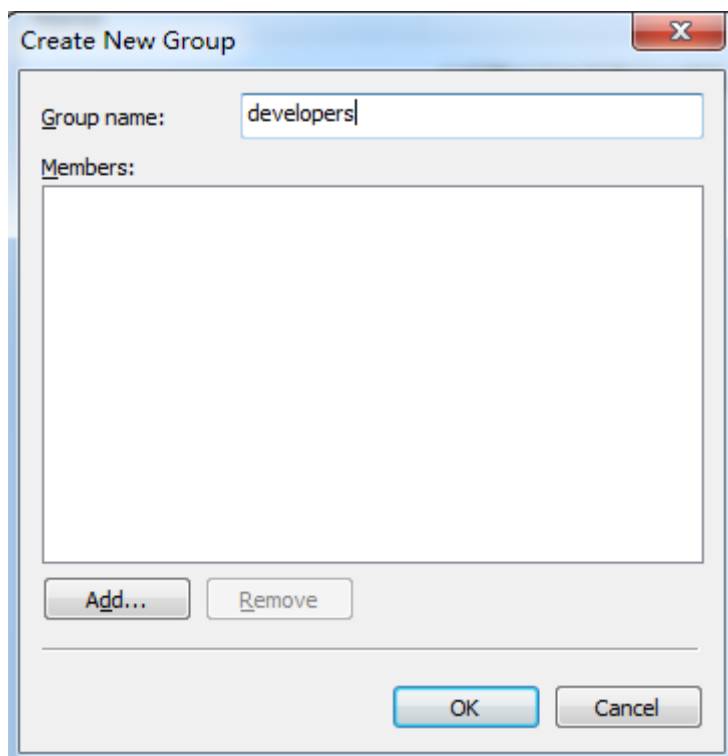
填写 username 和 password 后，点击 ok 按钮后，用户创建成功

1.2.3 建立用户组并分配权限

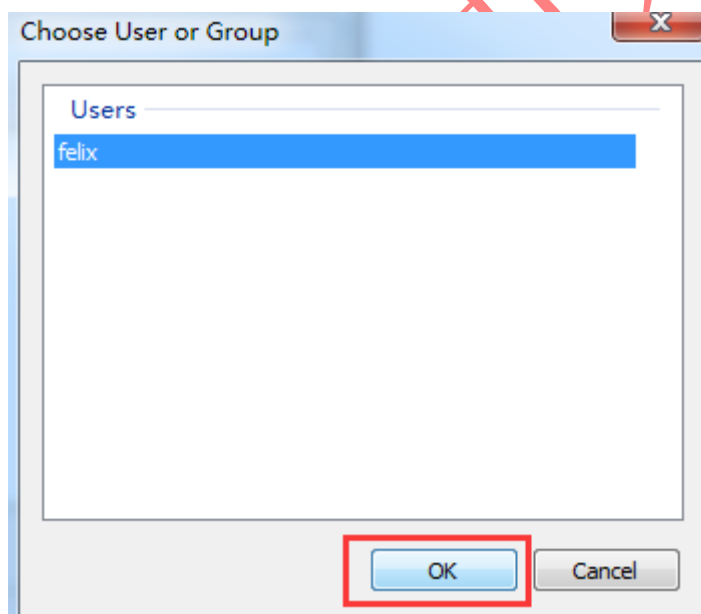
在 VisualSVN Server Manager 窗口的左侧右键单击 Groups 组，选择 Create Group 或者新建->Group，如图：



点击【Group】按钮后，进入如下图：

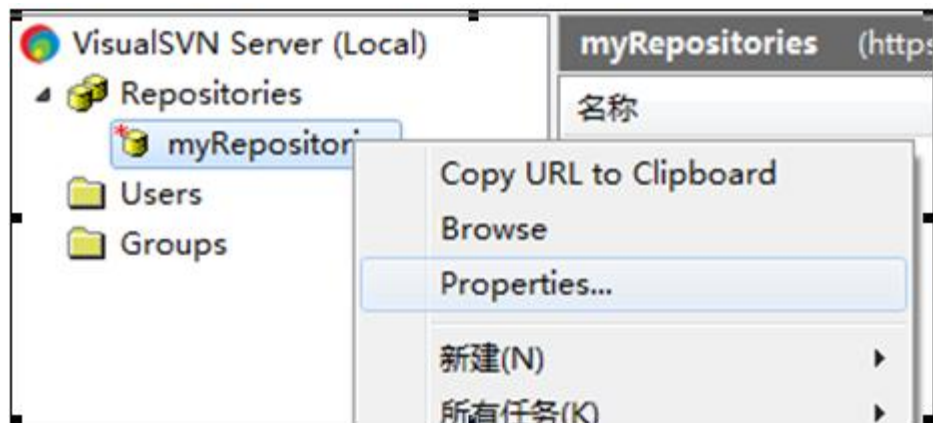


在弹出窗口中填写 Group name 为 Developers, 然后点 Add 按钮, 在弹出的窗口中选择 felix, 加入到这个组, 然后点 Ok.

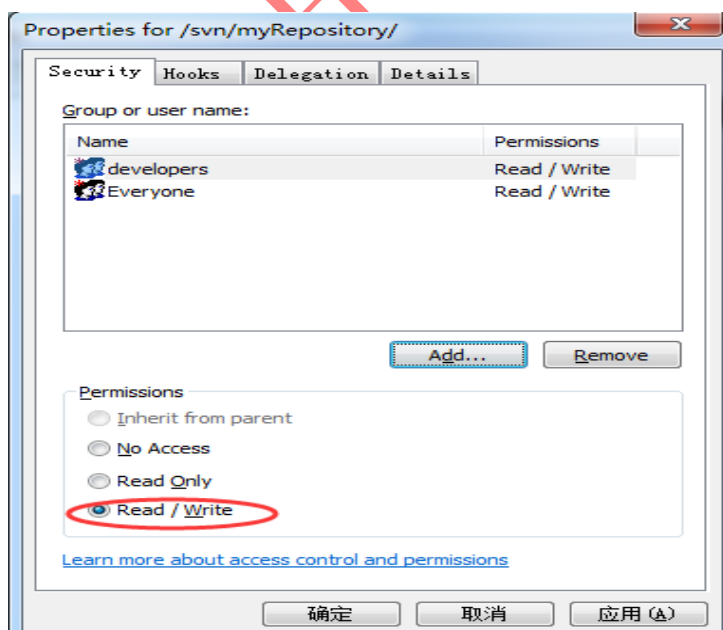
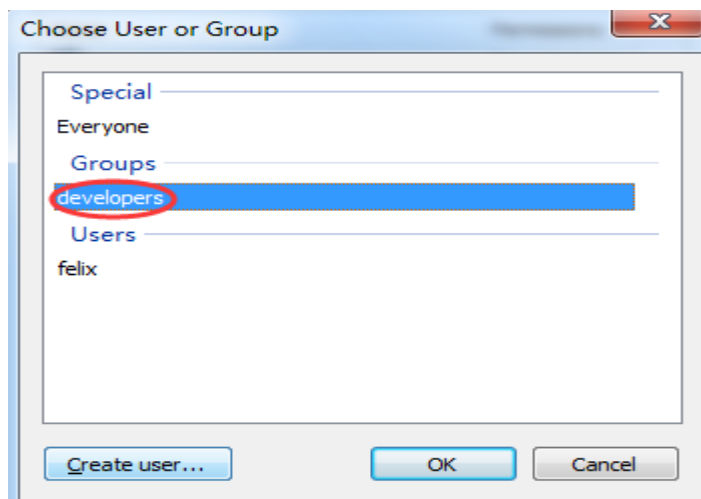


如果有多个用户, 操作一样

接下来我们需要给用户组设置权限, 在 myRepository 上单击右键, 选择属性, 如图:



在弹出的对话框中, 选择 Security 选项卡, 点击 Add 按钮, 选中 Group->developers 或者 Users->felix 给组或者用户分配权限, 然后添加进来, 权限设置为 Read/Write, 如下图:




点击【确定】按钮即可。

第2章 TortoiseSVN 客户端安装

2.1 首先我们需要下载 “svn 小乌龟”后进行安装

在安装的时候根据自己的操作系统（32 位/64 位）选择合适的客户端工具, 例如我下载的版本为

 TortoiseSVN_1.9.3.27038_win32.1454137896.msi ，直接下一步进行安装即可

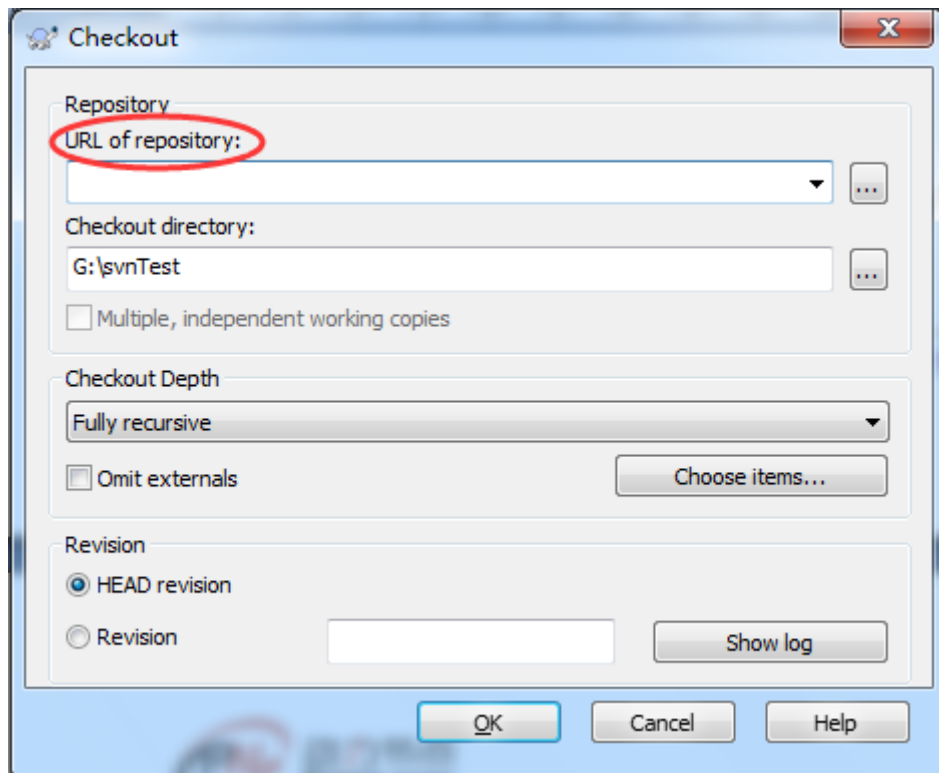
安装完成后，单击鼠标右键就可以看到如下内容：



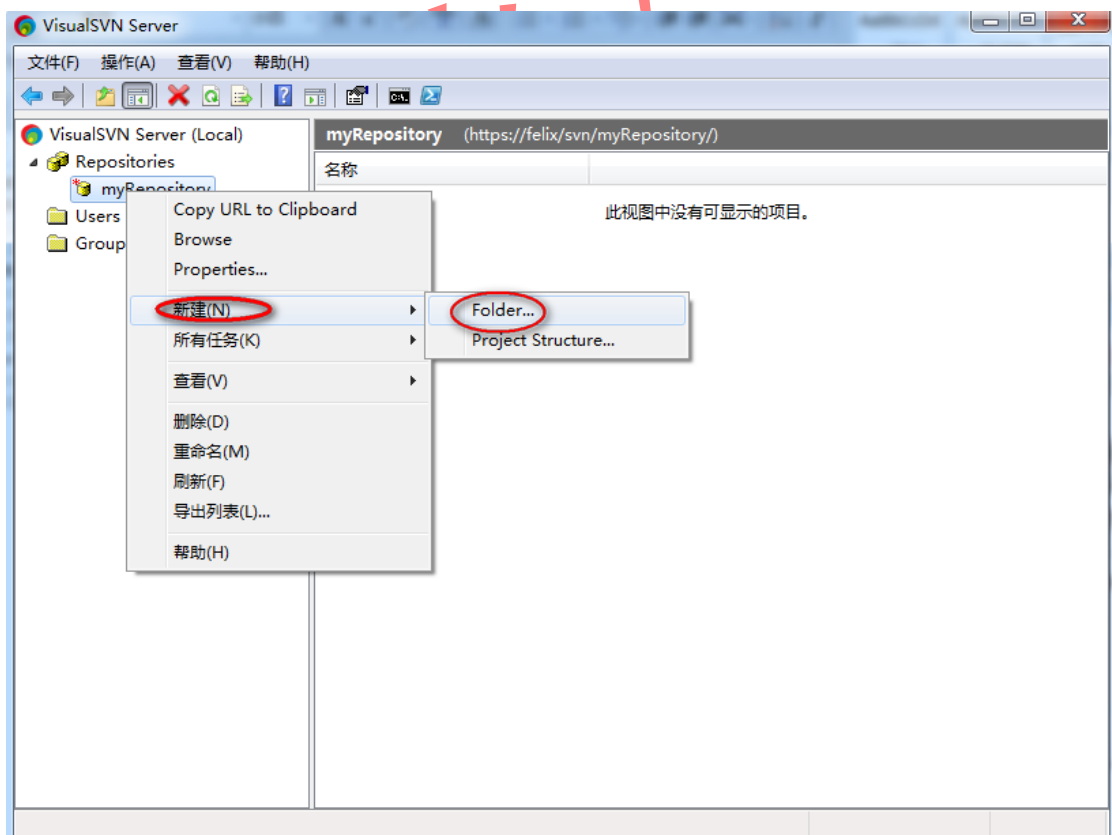
说明小乌龟客户端工具已经安装成功了, 如果没有看到效果，需要重启一下机器

2.2 checkout 项目文件

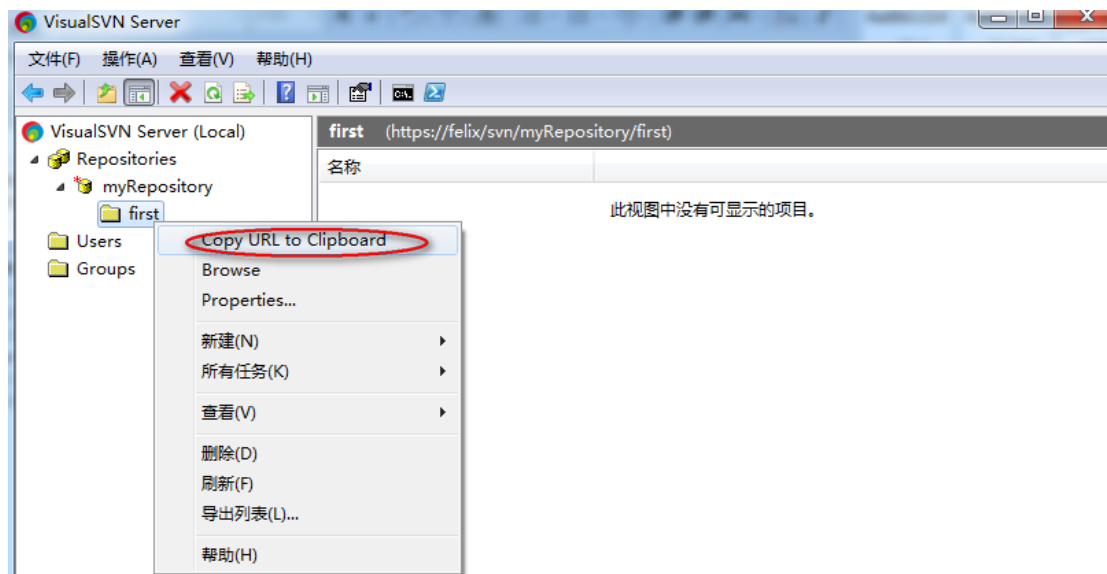
自己创建一个目录（一般是自己存放项目相关文档的目录），我这里创建目录名称叫 svnTest，右键 --> svn Checkout



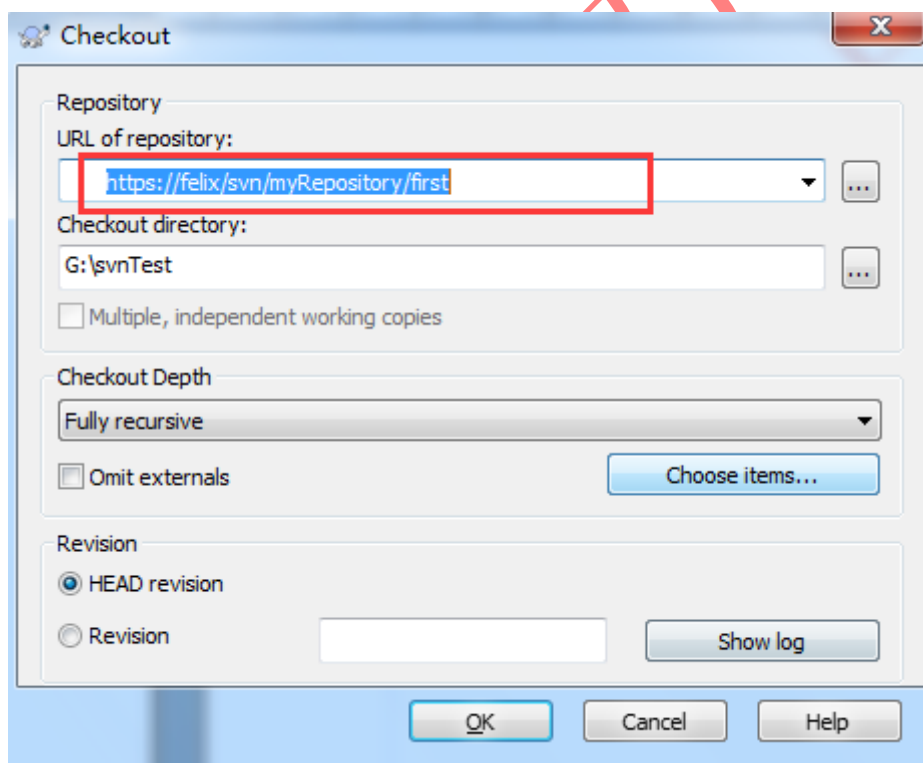
其中 URL 我可以在 SVN 服务器获取到，我在 myRepository 下右键新建文件



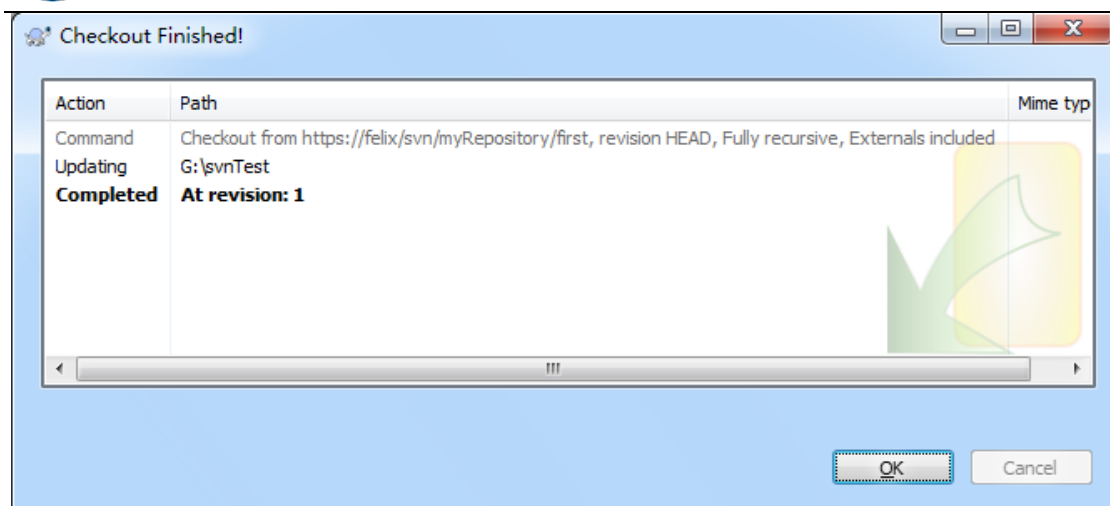
例如我创建的文件夹名字为first, 然后在该文件夹上这样右键 --> Copy URL to Clipboard



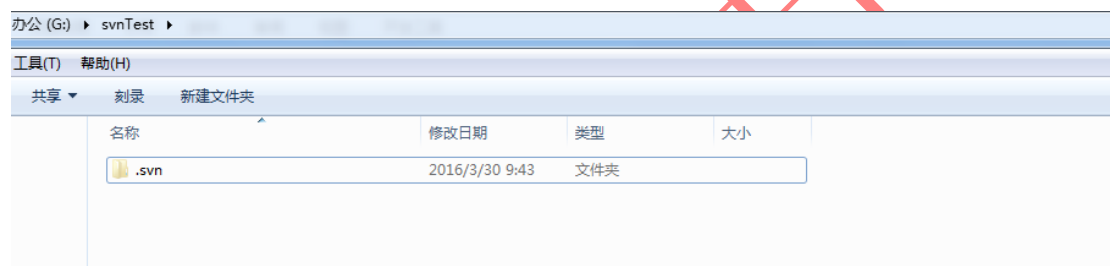
将复制的版本库 URL 粘贴在小乌龟客户端上, 如下图:



点击【ok】按钮后, 需要输入上面分配的用户名和密码, 就可以检索出来, 如下:



如下图：

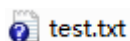


注意事项：

.svn 这个隐藏目录记录着两项关键信息：工作文件的基准版本和一个本地副本最后更新的时间戳，千万不要手动修改或者删除这个.svn 隐藏目录和里面的文件!!，否则将会导致你本地的工作拷贝(静态视图)被破坏，无法再进行操作。

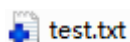
2.3 TortoiseSVN 图标介绍

2.3.1 蓝色问号



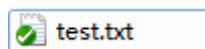
暂时还没受到版本的控制，但是没有被忽略。

2.3.2 蓝色加号



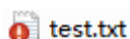
意味着这个文件或文件夹已经被计划加入到版本控制之下

2.3.3 绿色对勾



一个新检出的工作副本使用绿色的对勾重载，表示 Subversion 状态正常。

2.3.4 红色叹号



在你开始编辑一个文件之后，状态就变成了已修改，而图标重载已变成了红色感叹号。通过这种方式，你可以很容易地看出那些文件从你上次更新工作副本被修改过，且需要提交。

2.3.5 黄色叹号

如果在提交的过程中出现了冲突，图标就会变成了黄色感叹号。

2.4 TortoiseSVN Client 基础操作

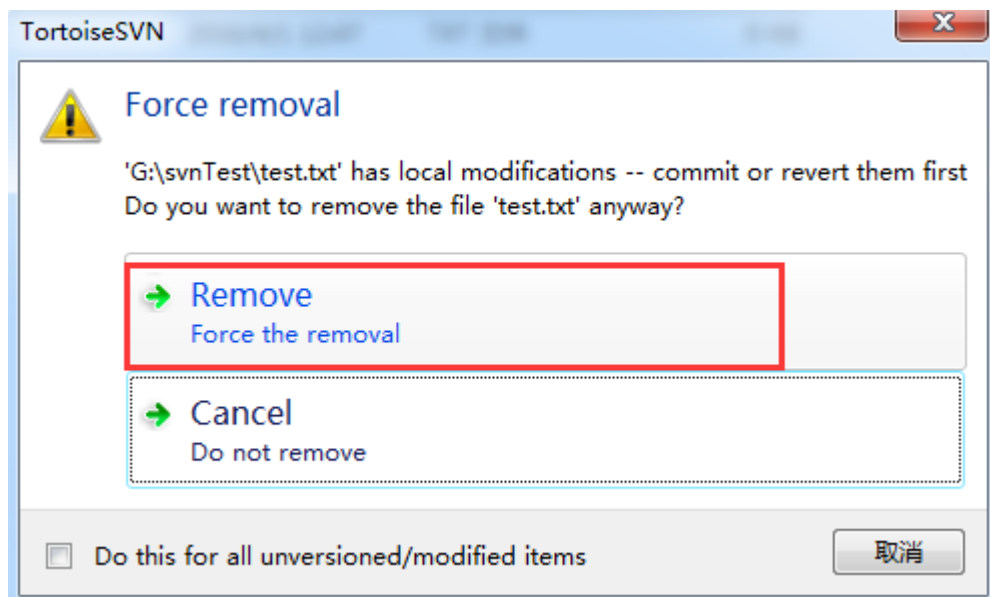
2.4.1 增加(Add)

选中要提交的文件，右键 SVN 菜单执行“Add“操作提交到”变更列表中”，然后右键 SVN 菜单执行“SVN Commit”提交到版本库中

2.4.2 删除(Delete)

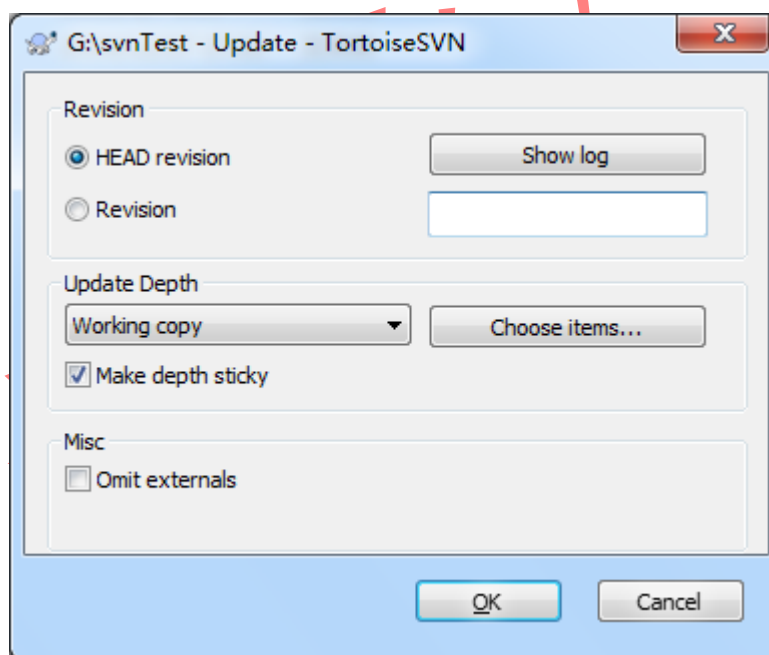
如果被删除的文件还未入版本库，则可以直接使用操作系统的删除操作删除该文件

如果被删除的文件已入版本库，则删除的方法如下：选择被删除文件，右键 svn 菜单执行“delete”操作，然后选择被删除文件的父目录，右键 svn 菜单执行“SVN Commit”。



2.4.3 SVN 还原(SVN Revert)

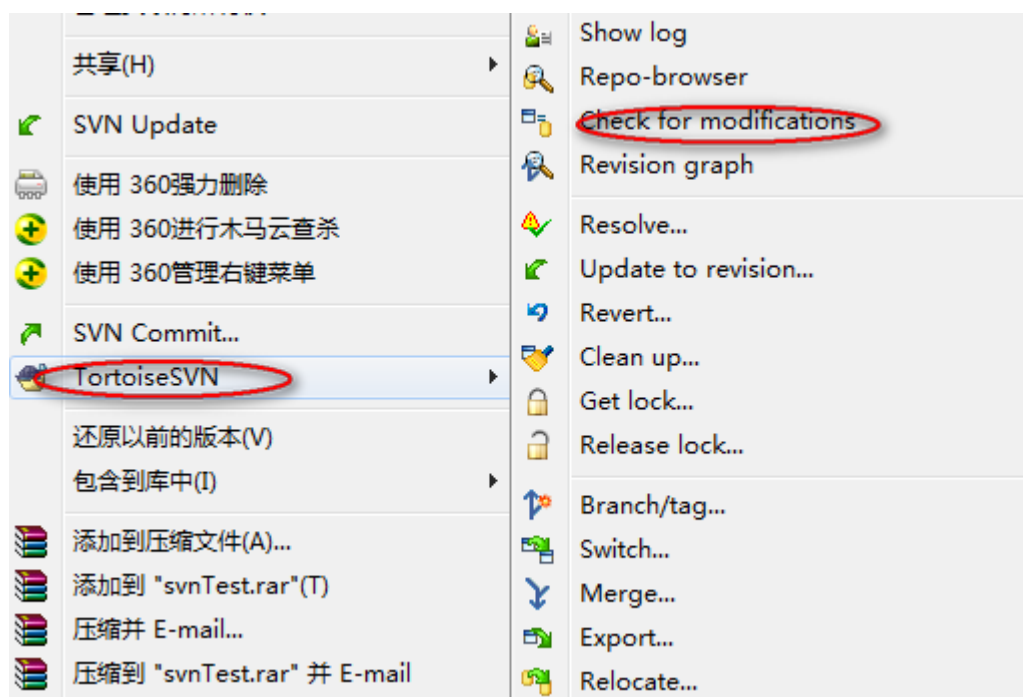
右击想要回退的文件或者文件夹，在 TortoiseSVN 弹出菜单中选择” Update to reversion...” 然后会弹出一个窗口，如下：



比如说我们要回退到第 10 个版本只需要在 Revision 中填写相应的版本号，然后点击 ok 即可。

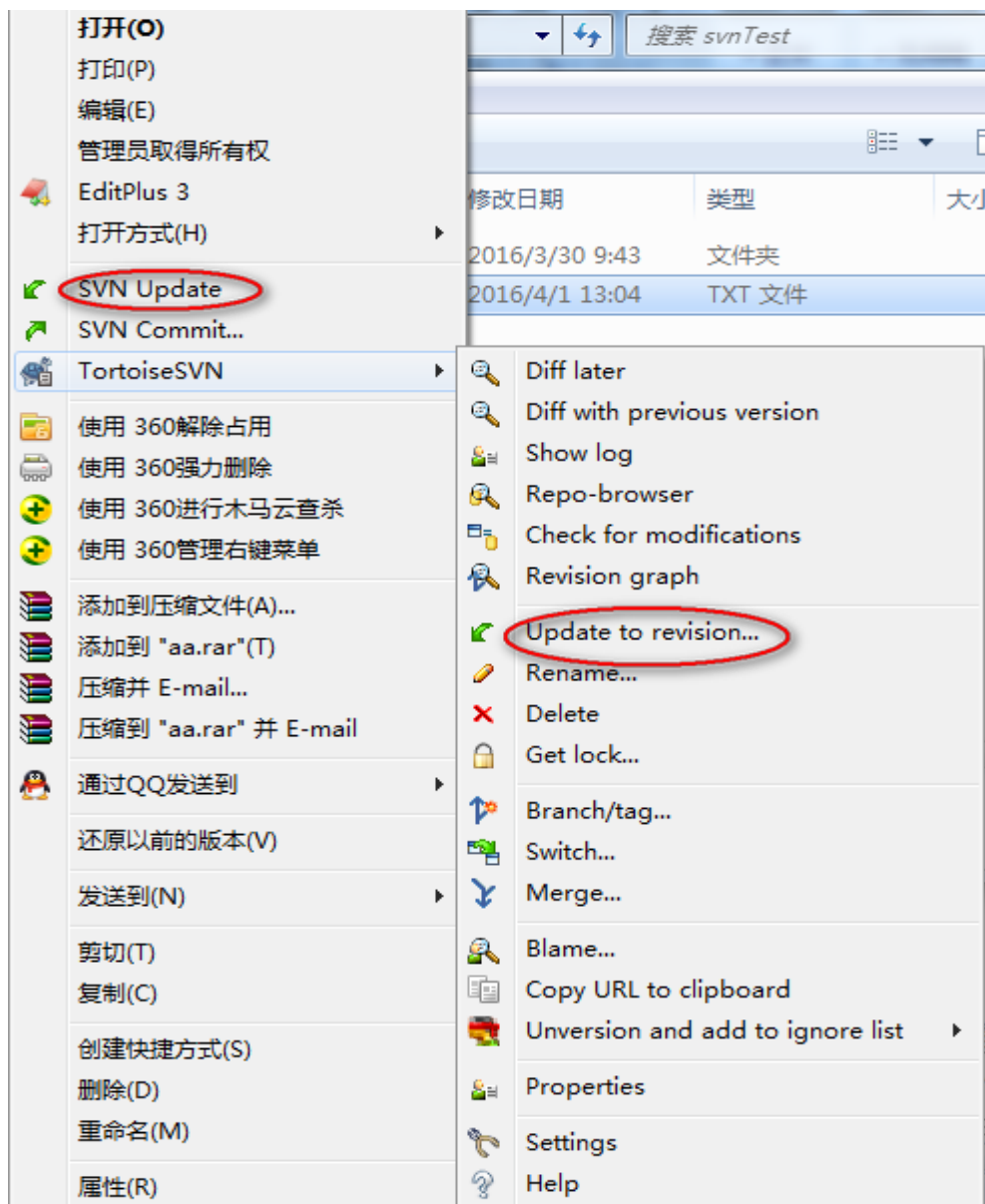
2.4.4 检查更新(Check for modifications)

此功能可以显示你所做的修改有哪些还没有提交的，此功能不光能看到对文件的修改变化，所有的变化都能看到，包括增加文件或者目录，删除文件或者目录，移动文件或者目录等，如果你点击了检查版本库，那你还可以看到版本库里的改动，既别人提交了哪些文件的改动，你还没更新到本地，如下：



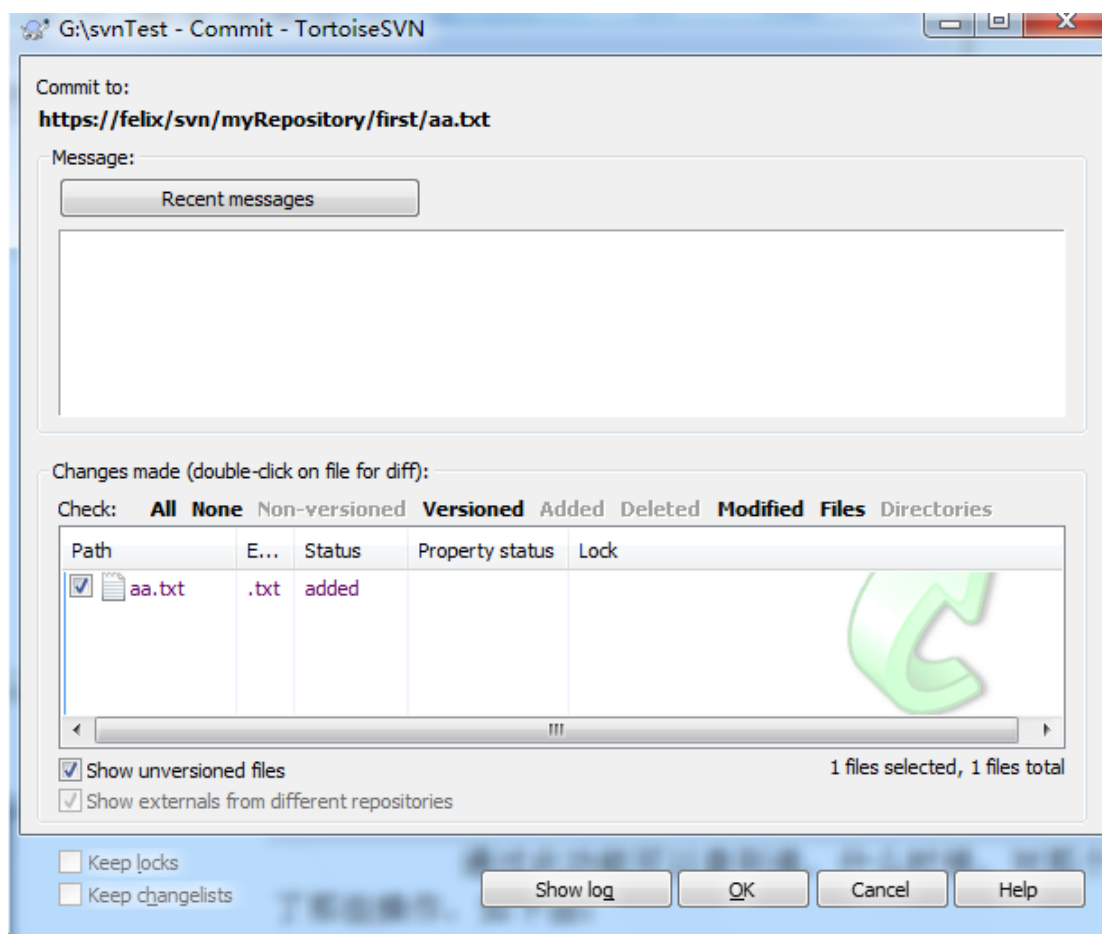
2.4.5 SVN 更新(SVN Update)

更新本地代码与 SVN 服务器上最新的版本一致，只要在需要更新的文件夹上点击右键或者在文件下空白处点击右键，选择”SVN Update”； 如果想获取指定版本中的内容，点击右键执行 SVN 菜单中的”Update to reversion “，就可以了。



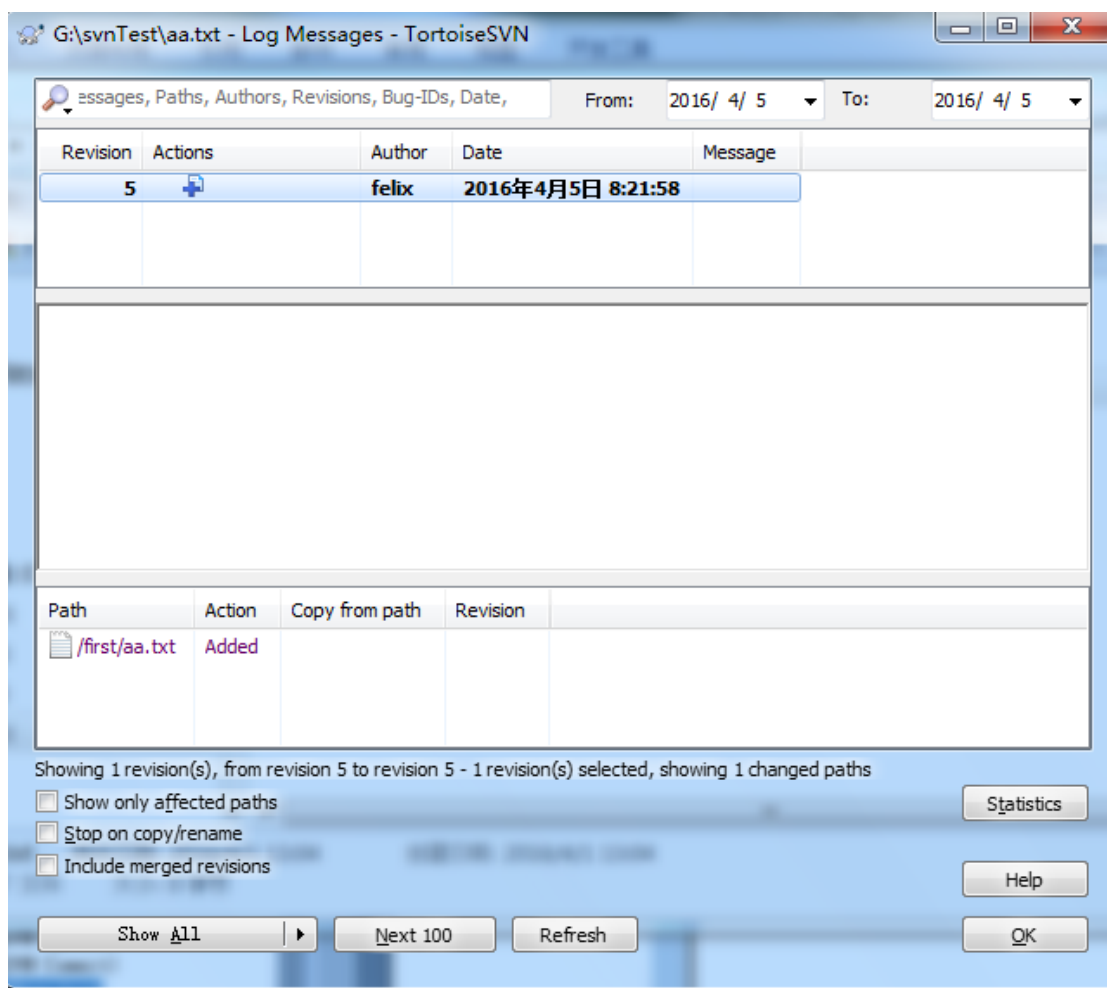
2.4.6 SVN 提交(SVN Commit)

Svn 的提交是将在工作空间做的修改进行提交，包括文件内容的修改，文件或目录的添加，删除，命名，移动等操作，具体操作选中要提交的文件或者目录，右键，SVN Commit，然后出现如下图所示内容：



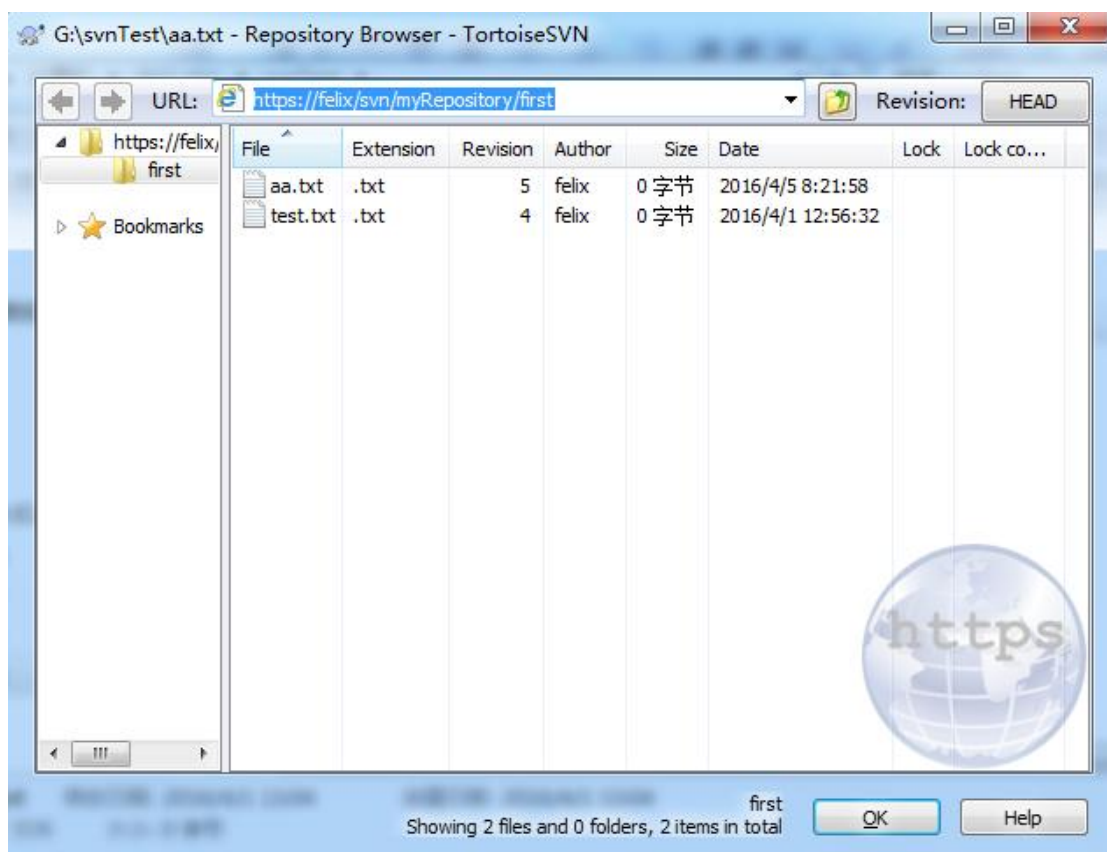
2.4.7 显示日志(Show log)

通过此功能可以查到谁，什么时候，对那个目录下的那些文件进行了那些操作，如下图：



2.4.8 版本库浏览(Repo-browser)

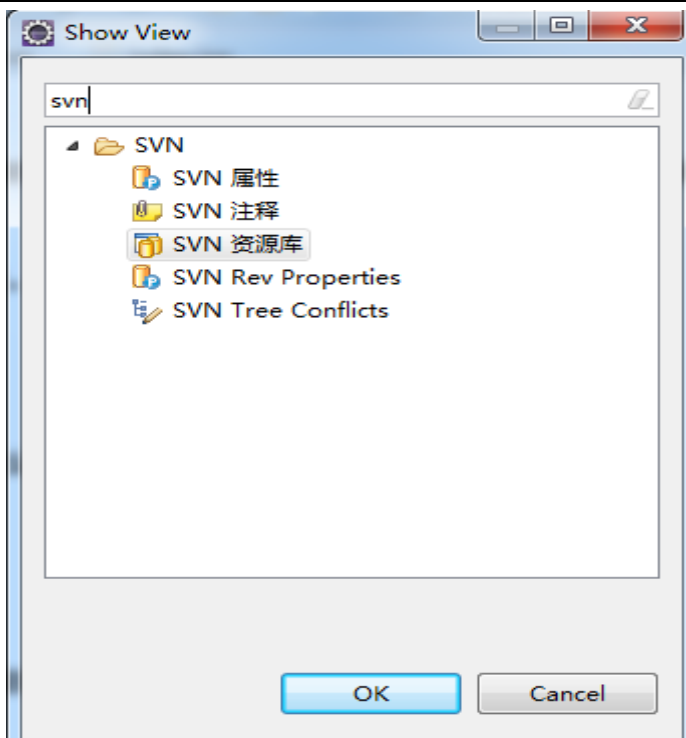
此功能是用来浏览需要查看的资料库，在本地文件夹下点击右键，选择 TortoiseSVNRepo-browser，在弹出的对话框中输入资料库地址，再输入用户名和密码，就能查看到你看到版本库的内容，在这你还能看到那些文件被谁锁定了，如下图：



第3章 使用 EclipseSVN 插件解决冲突

3.1 EclipseSVN 插件的安装

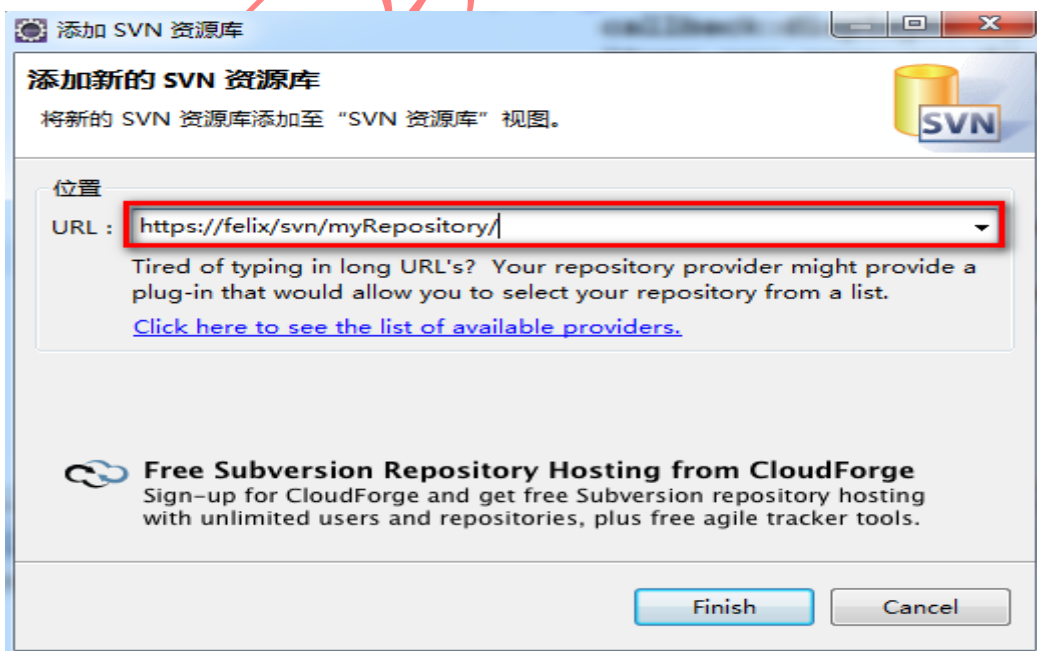
将 svn 插件目录拷贝的 eclipse 安装目录的 dropins 目录下,这是最简单的一种安装方式,如果安装成功,在 eclipse->Window->Show View 下找 svn,如果出现下图内容说明安装成功



然后选择资源库,显示资源库视图

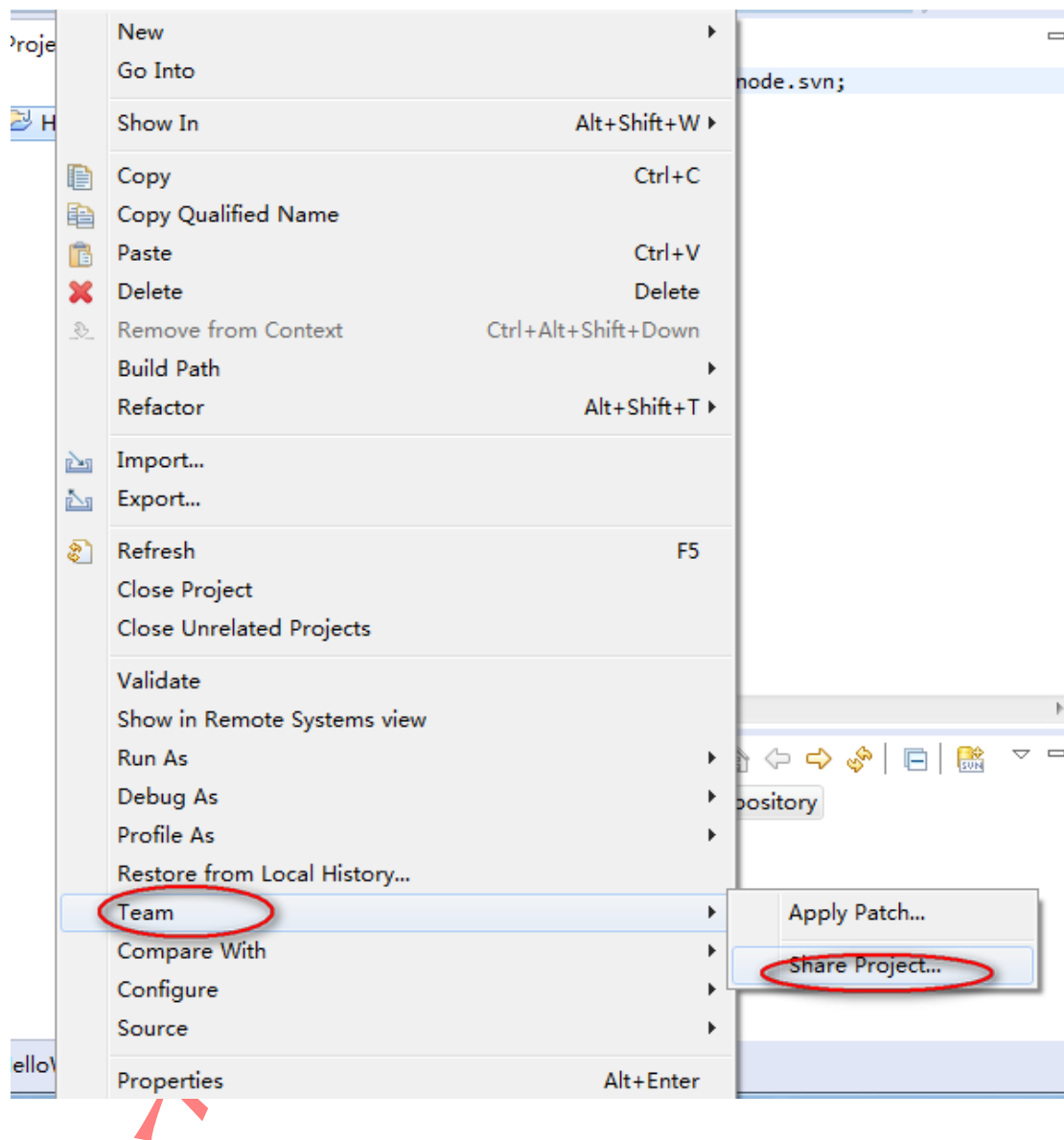
3.2 新建资源库位置

在资源库视图下,右键,新建资源库位置,url 为你要连接的 svn 资源库路径

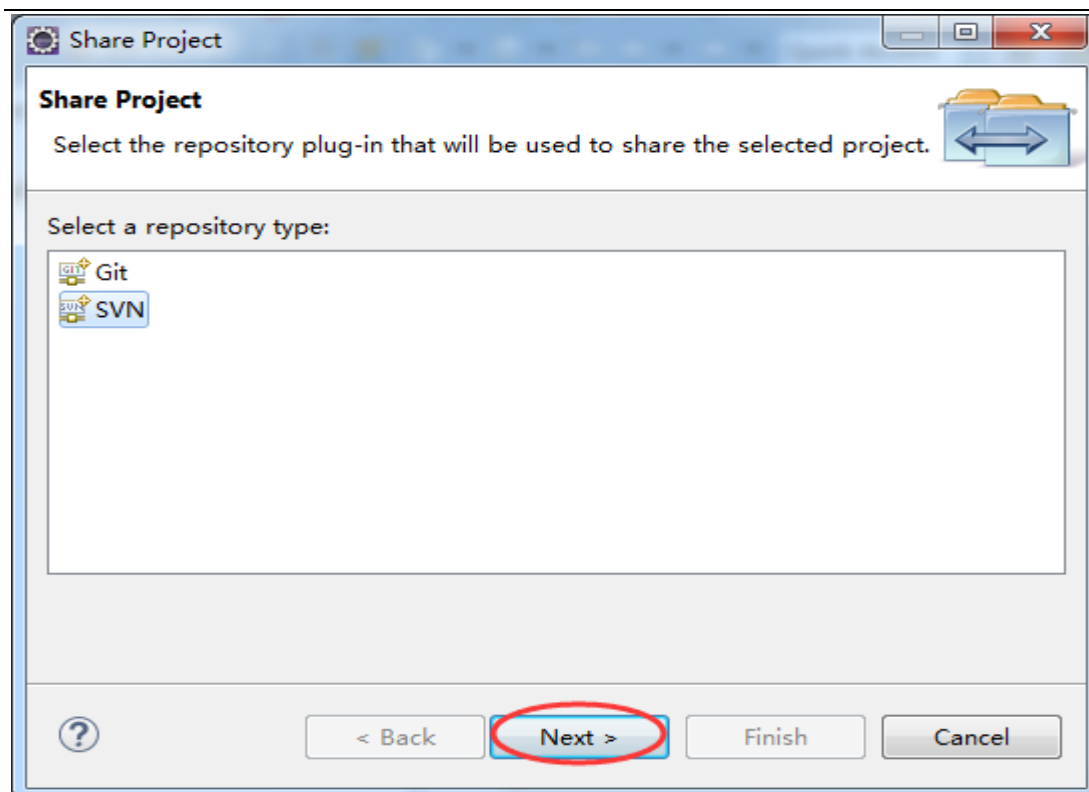


3.3 将 Eclipse 中的工程和 svn 资源库建立关系(共享)

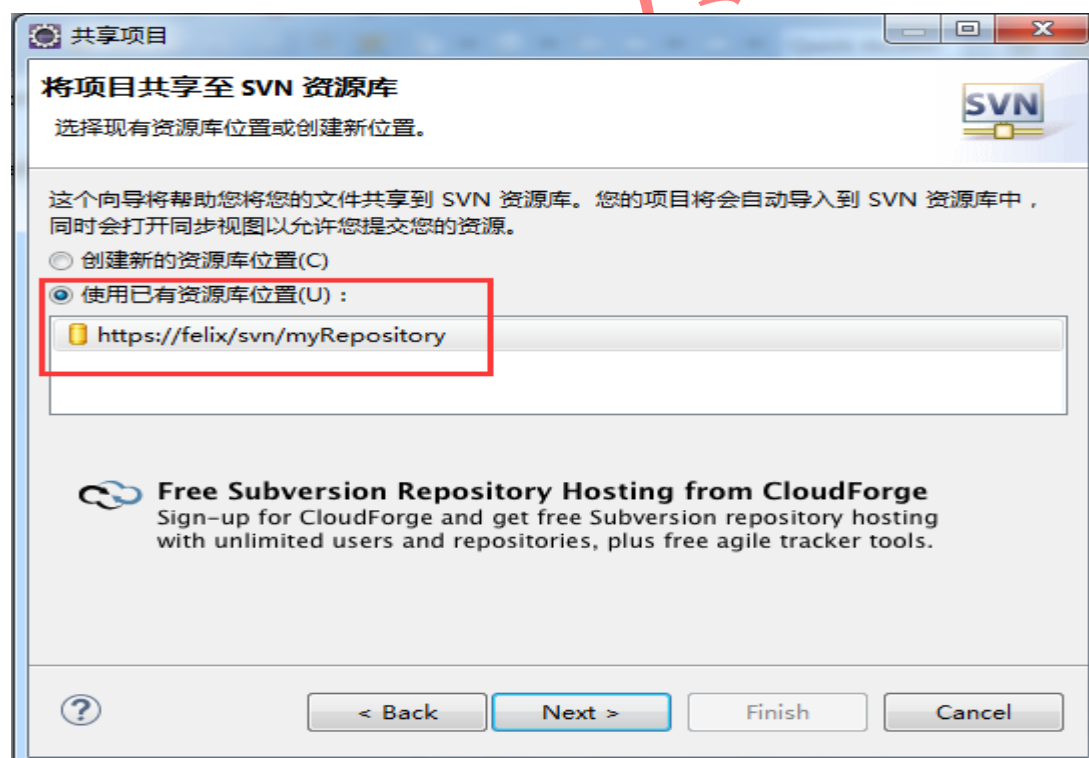
选择要和 SVN 建立关系的工程,右键->Team->Share Project,如图



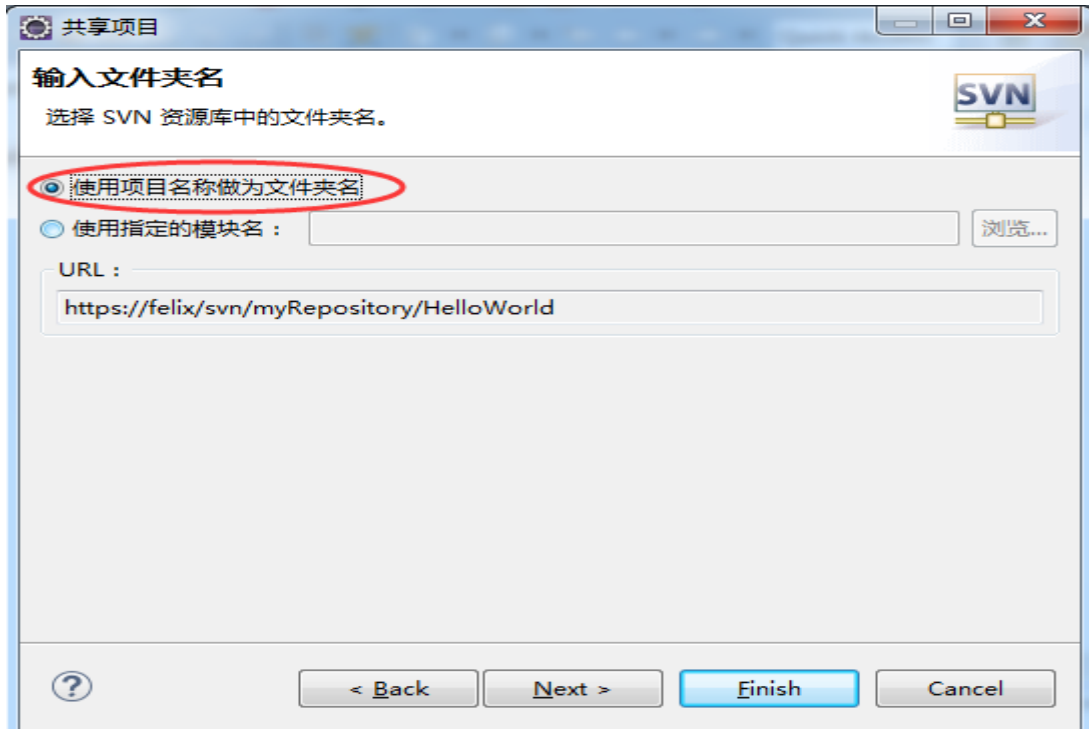
然后出现如下页面,选择 SVN,点击 next



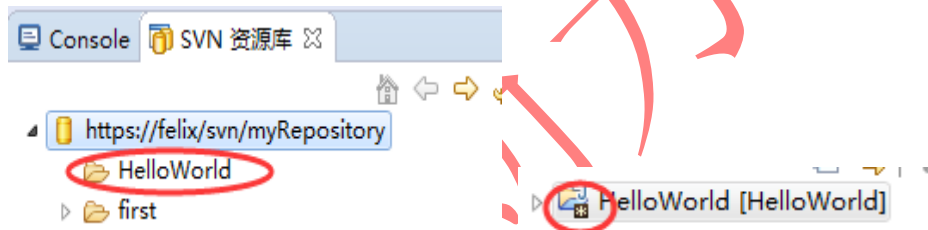
选择使用已有资源库位置



出现如下页面,选择使用项目名称作为文件夹名,点击 finish



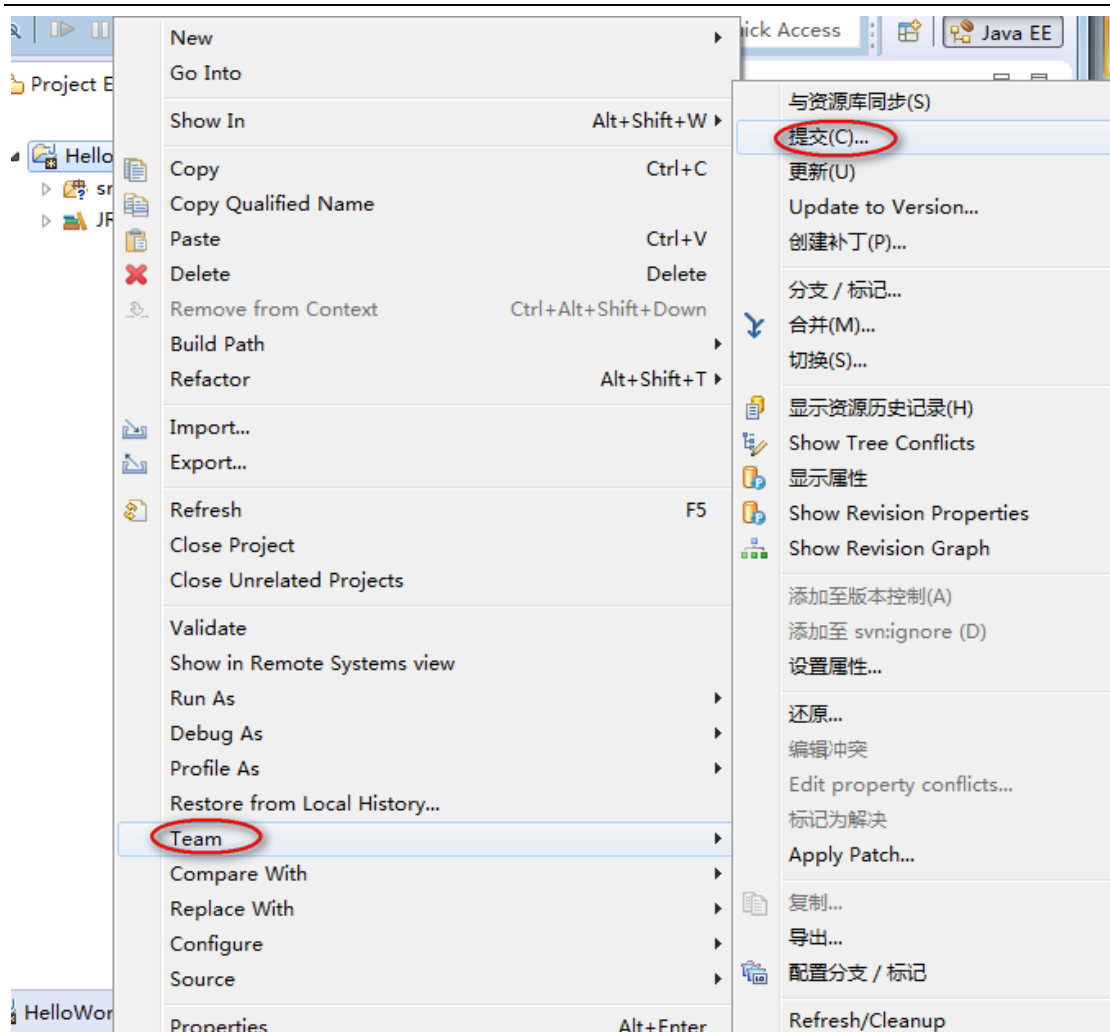
这个时候在 SVN 资源库视图,会出现 HelloWorld,项目前会出现如下图标



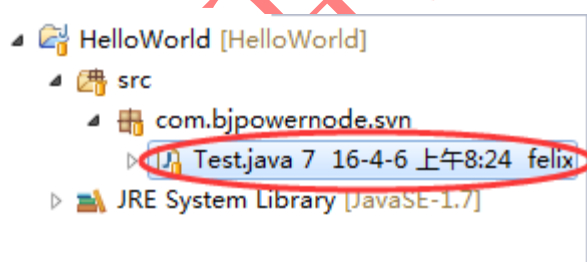
但是只是建立了连接,并没有将项目中的内容提交到 SVN 服务器上

3.4 提交工程中的内容

选择要提交内容的项目,右键->Team->提交,如图

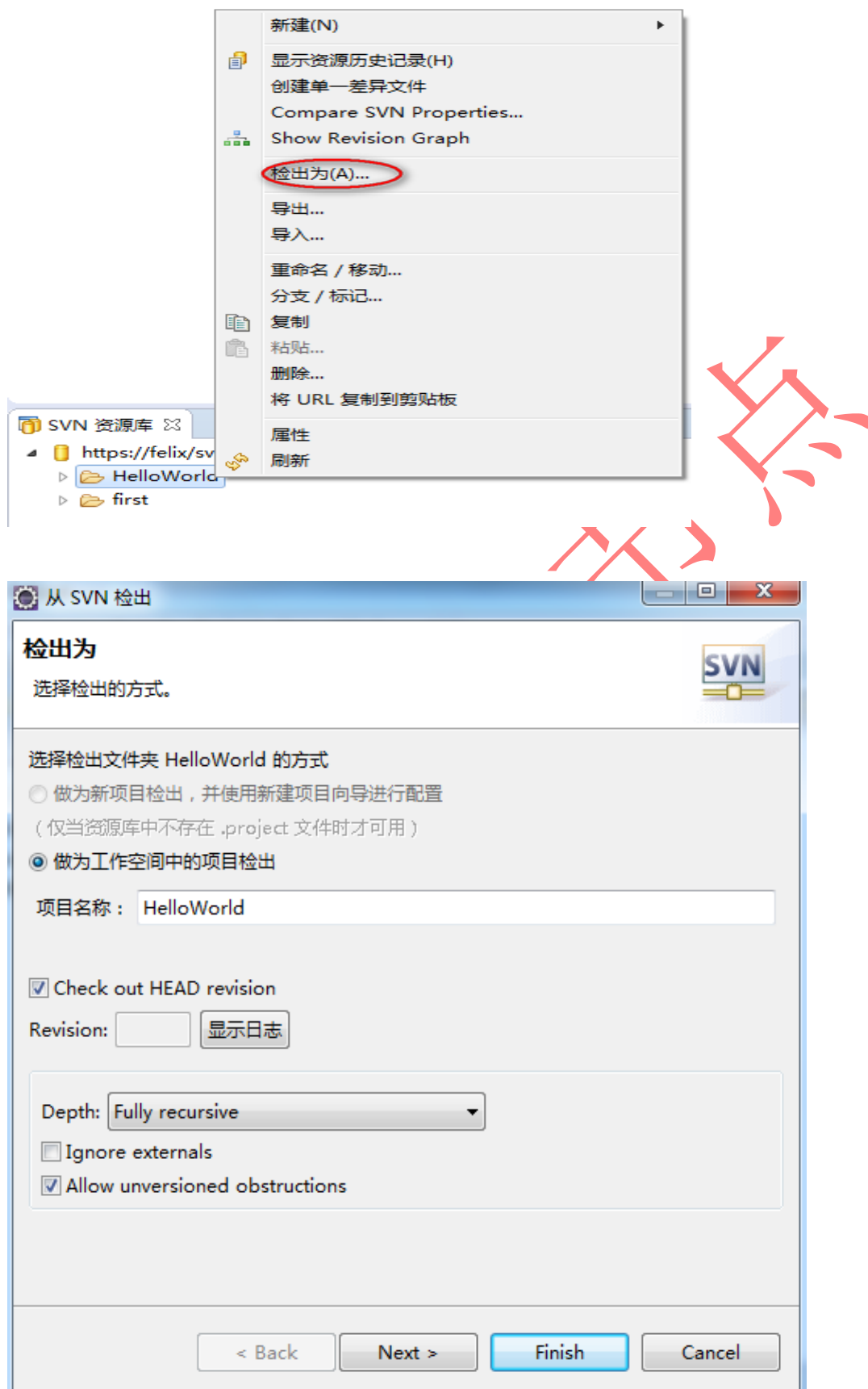


注意,不要提交本地生成的代码,提交的时候养成写注释的习惯,提交成功后,工程前的图标如下



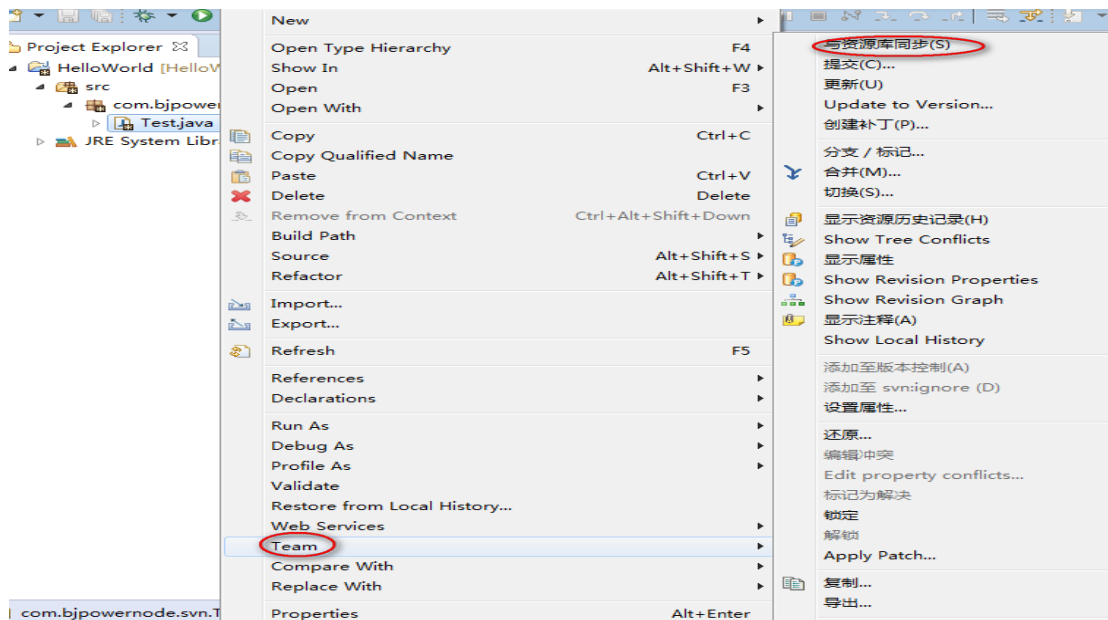
3.5 从 svn 上检出项目

在 SVN 资源库视图下,选中要更新的项目,检出为

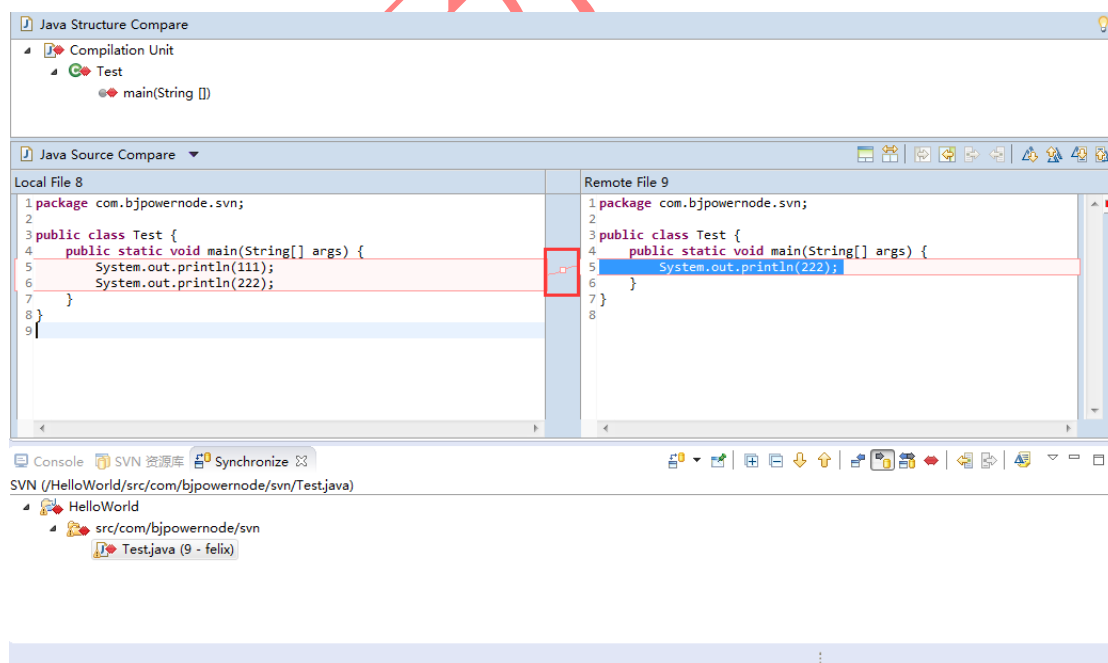


3.6 解决冲突

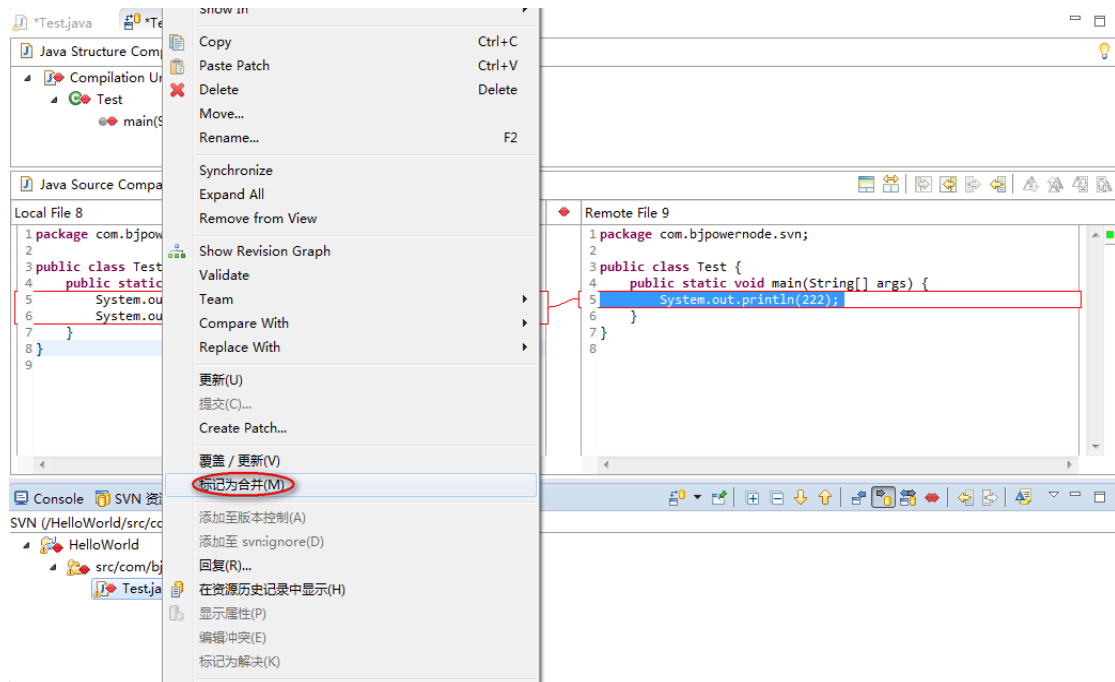
如果提交的过程中,出现了冲突,在发生冲突的文件上右键->Team->与资源库同步



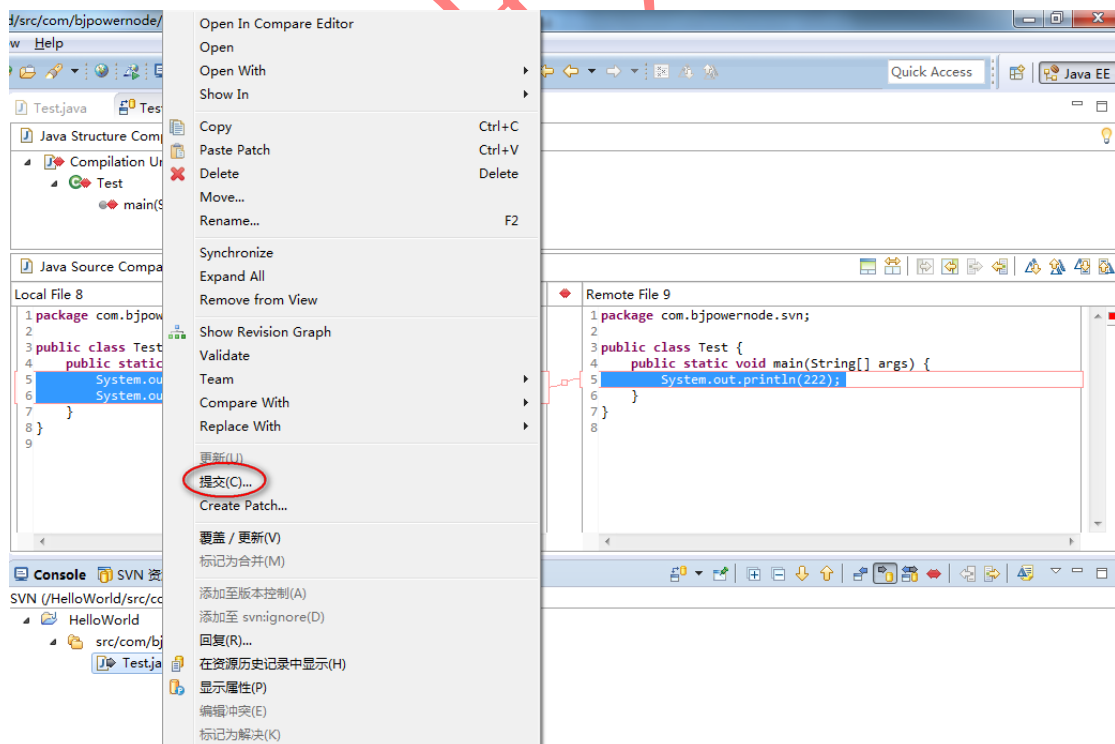
会出现如下的内容,左边为本地文件,右边为资源库上的文件,然后点击中间的框,将 svn 资源库的更新拷贝到本地



在发生冲突的文件上,右键->标记为合并



将 svn 资源库的文件的变化合并到本地后,提交代码



第4章 SVN 使用注意事项

4.1 提交之前先更新

- SVN 更新的原则是要随时更新，随时提交。当完成了一个小功能，能够通过编译并且自己测试之后，谨慎地提交。
- 如果在修改的期间别人也更改了 svn 的对应文件，那么 commit 就可能会失败。如果别人和自己更改的是同一个文件，那么 update 时会自动进行合并，如果修改的是同一行，那么合并时会产生冲突，这种情况就需要同之前的开发人员联系，两个人一起协商解决冲突，解决冲突之后，需要两人一起测试保证解决冲突之后，程序不会影响到其他功能。
- 在更新时注意所更新文件的列表，如果提交过程中产生了更新，则也是需要重新编译并且完成自己的一些必要测试，再进行提交。这样既能了解别人修改了哪些文件，同时也能避免 SVN 合并错误导致代码有错

4.2 保持原子性的提交

- 每次提交的间歇尽可能地短，以几个小时的开发工作为宜。例如在更改 UI 界面的时候，可以每完成一个 UI 界面的修改或者设计，就提交一次。在开发功能模块的时候，可以每完成一个小细节功能的测试，就提交一次，在修改 bug 的时候，每修改掉一个 bug 并且确认修改了这个 bug，也就提交一次。我们提倡多提交，也就能多为代码添加上保险。

4.3 提交时注意不要提交本地自动生成的文件

- 一般配置管理员都会将项目中一些自动生成的文件或者与本地配置环境有关的文件屏蔽提交（例如 eclipse 中的.classpath 文件等）。如果项目中没有进行这方面的配置来强行禁止提交这样的文件，请自觉不要提交这样的文件。提交了这样的文件后，别人在更新后就可能与本地的环境冲突从而影响大家的工作。

4.4 不要提交不能通过编译的代码

- 代码在提交之前，首先要确认自己能够在本地编译。如果在代码中使用了第三方类库，要考虑到项目组成员中有些成员可能没有安装相应的第三方类库。项目经理在准备项目工作区域的时候，需要考虑到这样的情况，确保开发小组成员在签出代码之后能够在统一的环境中进行编译。

4.5 不要提交自己不明白的代码

- 代码在提交入 SVN 之后，你的代码将被项目成员所分享。如果提交了你不明白的代码，你看不懂，别人也看不懂，如果在以后出现了问题将会成为项目质量的隐患。因此在引入任何第三方代码之前，确保你对这个代码有一个很清晰的了解。

4.6 提前协调好项目组成员的工作计划

- 项目经理应该合理分配工作计划。每个成员在准备开始进行某项功能的修改之前，如果有可能，先跟工作小组的成员谈谈自己的修改计划，让大家都能了解你的思想，了解你即将对软件作出的修改，这样能尽可能的减少在开发过程中可能出现的冲突，提高开发效率。同时你也能够在和成员的交流中发现自己之前设计的不足，完善你的设计。

4.7 对提交的信息采用明晰的标注

- 在一个项目组中使用 SVN，如果提交空的标注或者不确切的标注将会让项目组中其他的成员感到很无奈，项目经理无法很清晰的掌握工作进度，无法清晰的把握此次提交的概要信息。在发现错误后也无法准确的定位引起错误的文件。所以，在提交工作时，要填写明晰的标注，能够概要的描述所提交文件的信息，让项目组其他成员在看到标注后不用详细看代码就能了解你所做的修改。

4.8 慎用锁定功能

- 在项目中要慎用锁定的功能，在你锁定了一个文件之后别人就无法继续修改提交该文件，虽然可以减少冲突的发生率，但是可能会影响项目组中其他人员的工作。平时只有在编辑那些无法合并的文件（例如图片文件，flash 文件等）时，才适当的采用锁定操作。