

[← SVN 标签](#)

TortoiseSVN 使用教程

TortoiseSVN 是 Subversion 版本控制系统的一个免费开源客户端，可以超越时间的管理文件和目录。

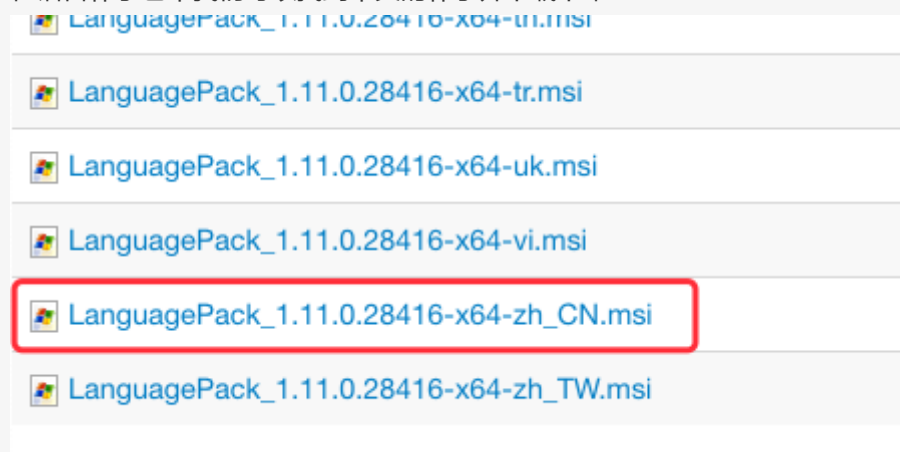
TortoiseSVN 安装

下载地址：<https://tortoisesvn.net/downloads.html>, 页面里有语言包补丁的下载链接。

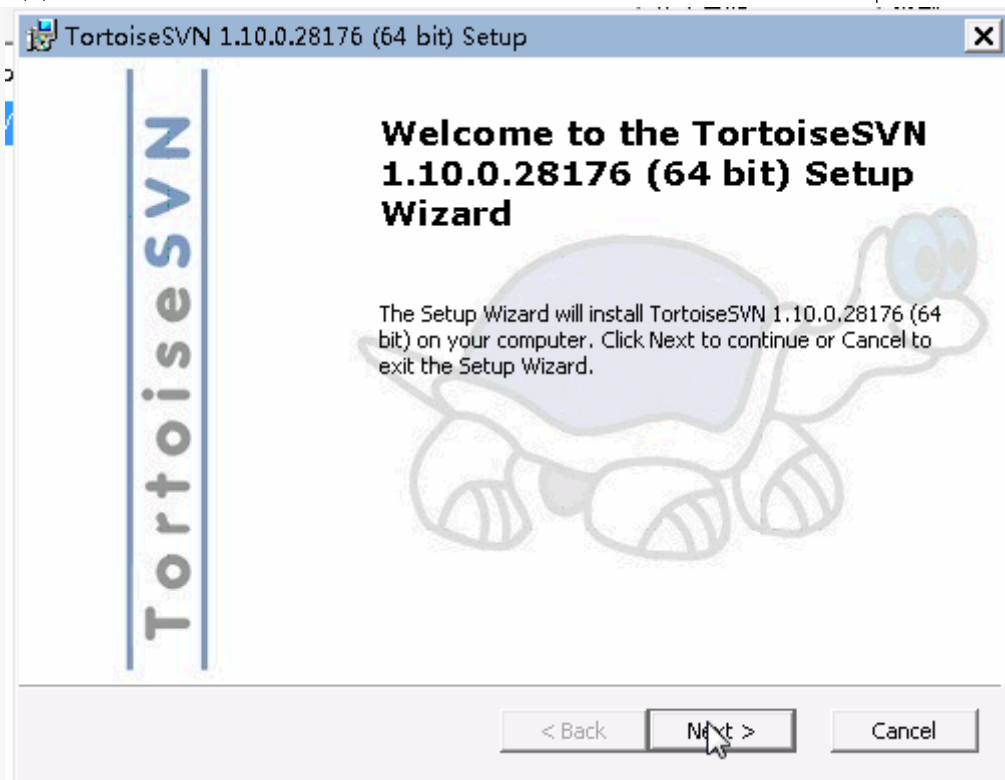
目前最新版为 1.11.0 下载地址：<https://osdn.net/projects/tortoisesvn/storage/1.11.0/>



在语言补丁包中我们可以找到中文的补丁并下载下来：



运行下载的 TortoiseSVN 安装程序



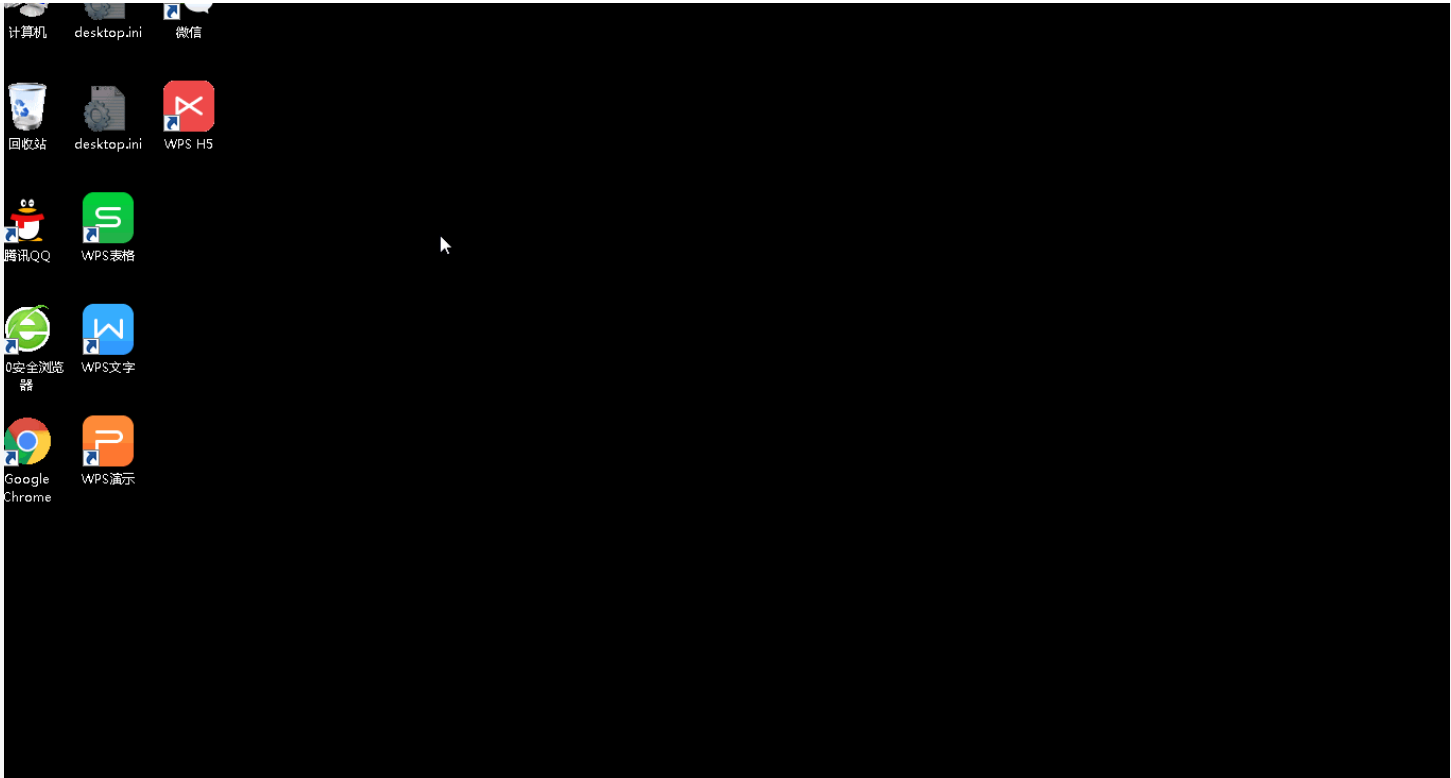
运行下载的 TortoiseSVN 中文语言包



正确安装后，应该进行一次的重开机，以确保 TortoiseSVN 的正确无误。

修改 TortoiseSVN 默认语言

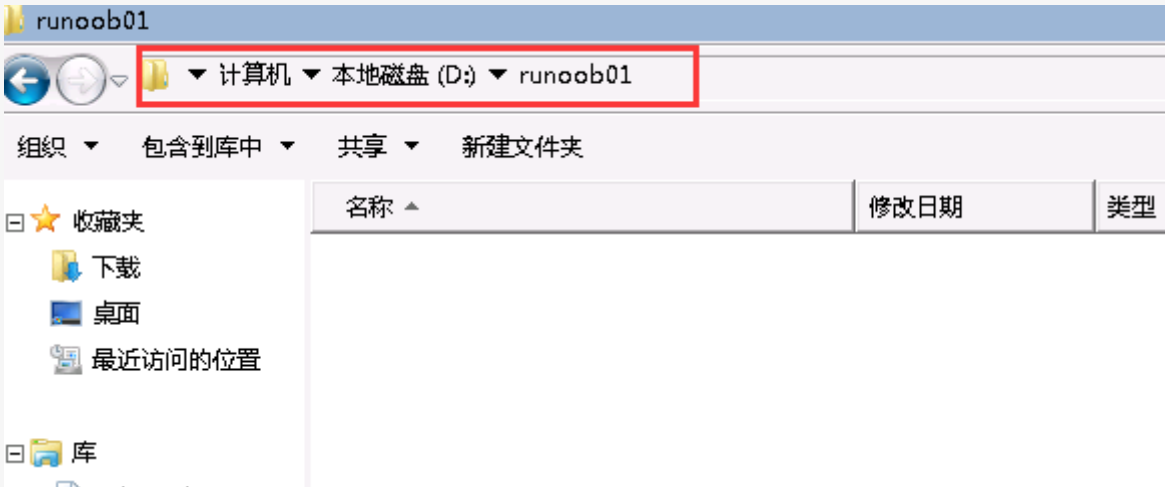
TortoiseSVN 安装完后默认的界面是英文的，我们可以通过设置修改成已安装语言



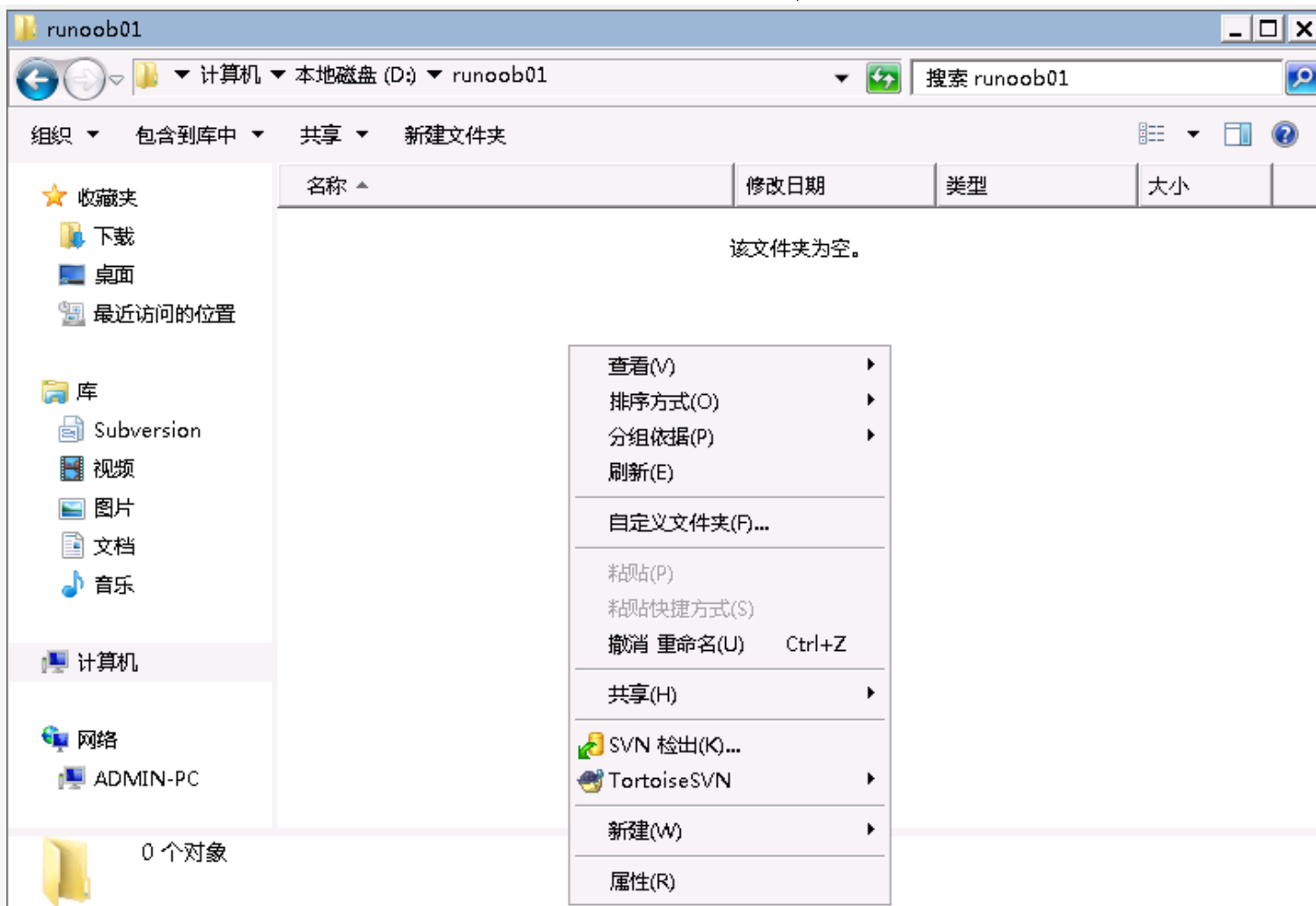
TortoiseSVN 的使用

建立一个 runoob01 的工作目录

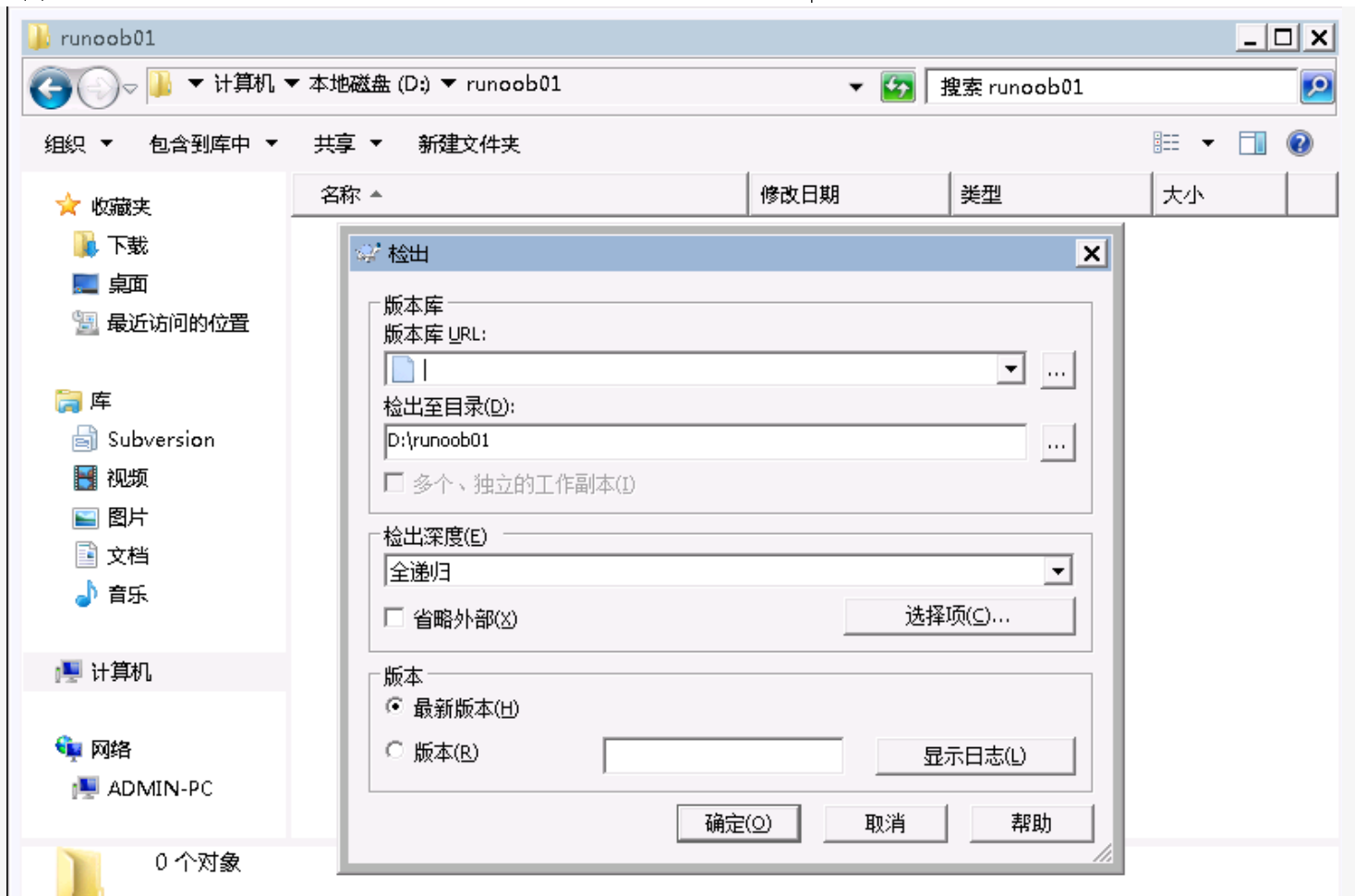
所谓的 runoob01 目录其实就是您平常用来存放工作档案的地方。通常会等到自己的工作做的一个段落的时候再进行备份。所以我们平常都是在 runoob01 目录下面工作，等到适当时机在 commit 到 repository 中。举例来说，我们想在 D 盘下面建立一个名为 runoob01 的目录。首先先把这个目录建立出来。



进入创建的目录在空白处按下右键后(您可以在 MyWork 目录的 icon 上按，也可进入 MyWork 目录后，在空白的地方按)，选择 **SVN checkout**。

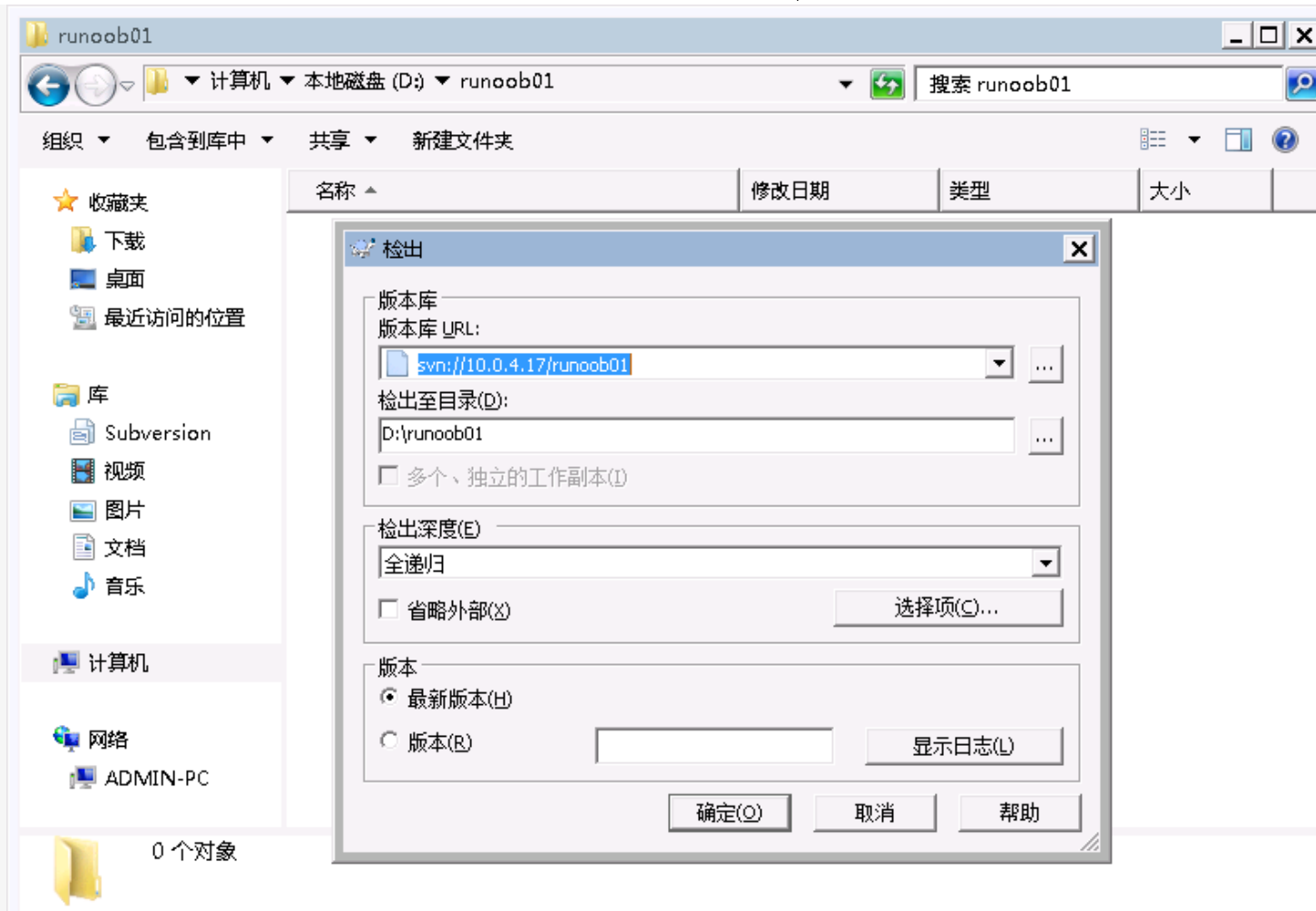


接着您可以看到如下的画面：



首先我们要填入的是 repository(版本库)的位置，对于 SVN 来说，repository 的位置都是 URL。版本库 URL 这里填入我们测试的版本仓库地址 **svn://10.0.4.17/runoob01**。

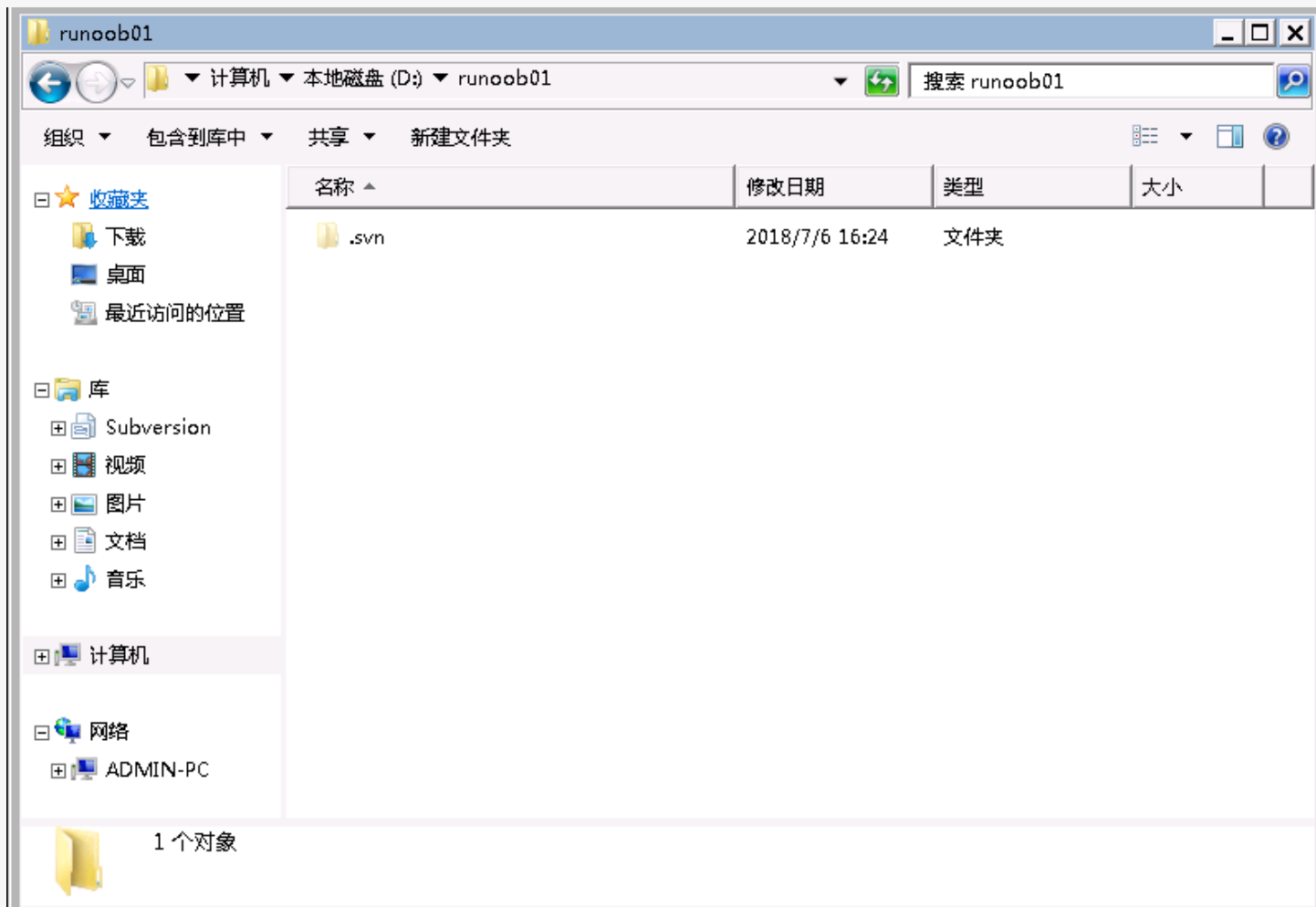
接着，稍微看一下 Checkout directory(检出至目录)，这个字段应该要指向您的 runoob01 目录。



确认后，按下 OK 按钮，您应该可以看到如下的讯息窗口。



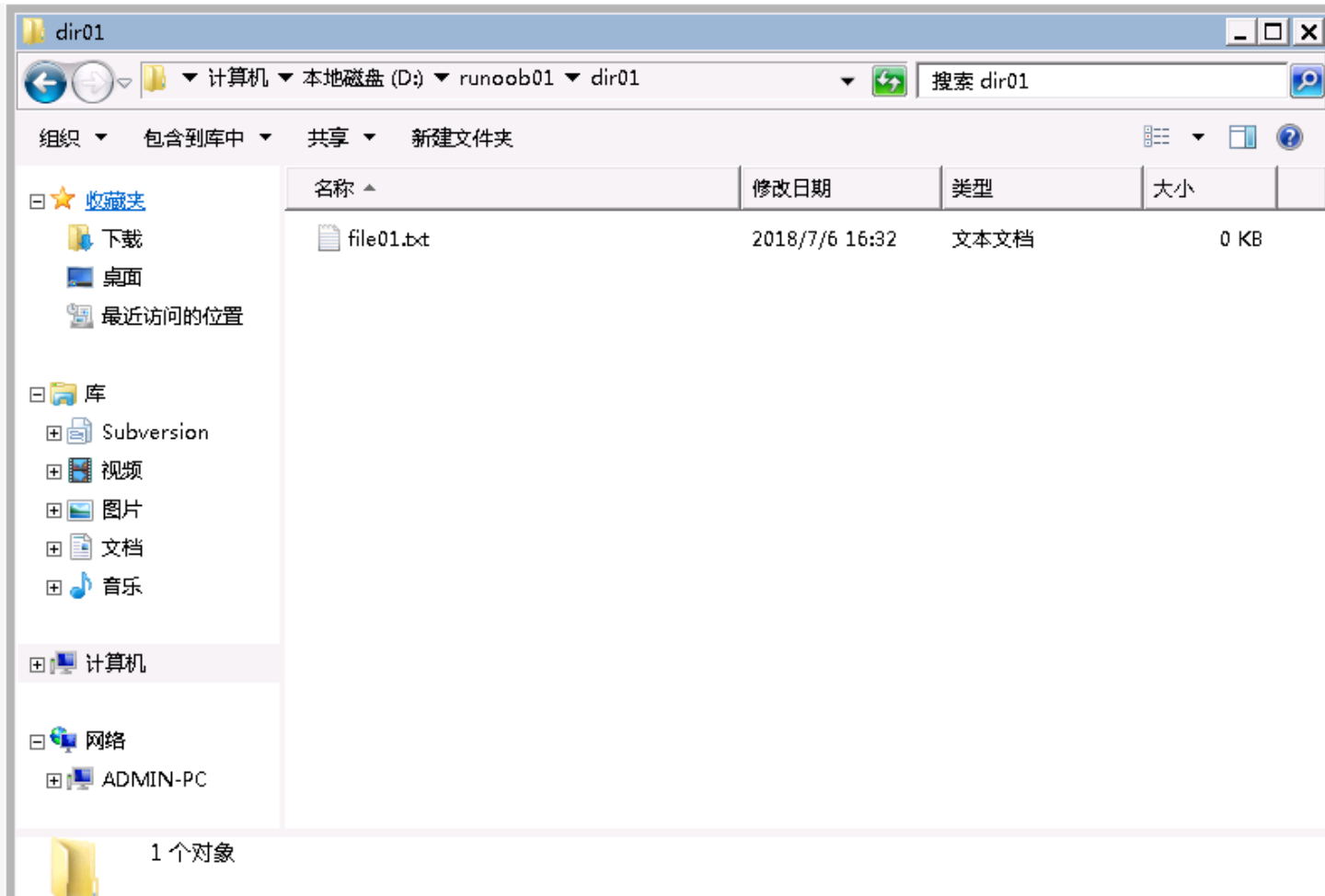
这样就表示动作完成。按下 OK 按钮后，再到您刚刚建立的目录下。您将会看到 MyWork 目录下面多了一个名为 `.svn` 的目录 (这个目录是隐藏的，如果您的档案管理员没有设定可以看到隐藏目录，您将无法看到它)。



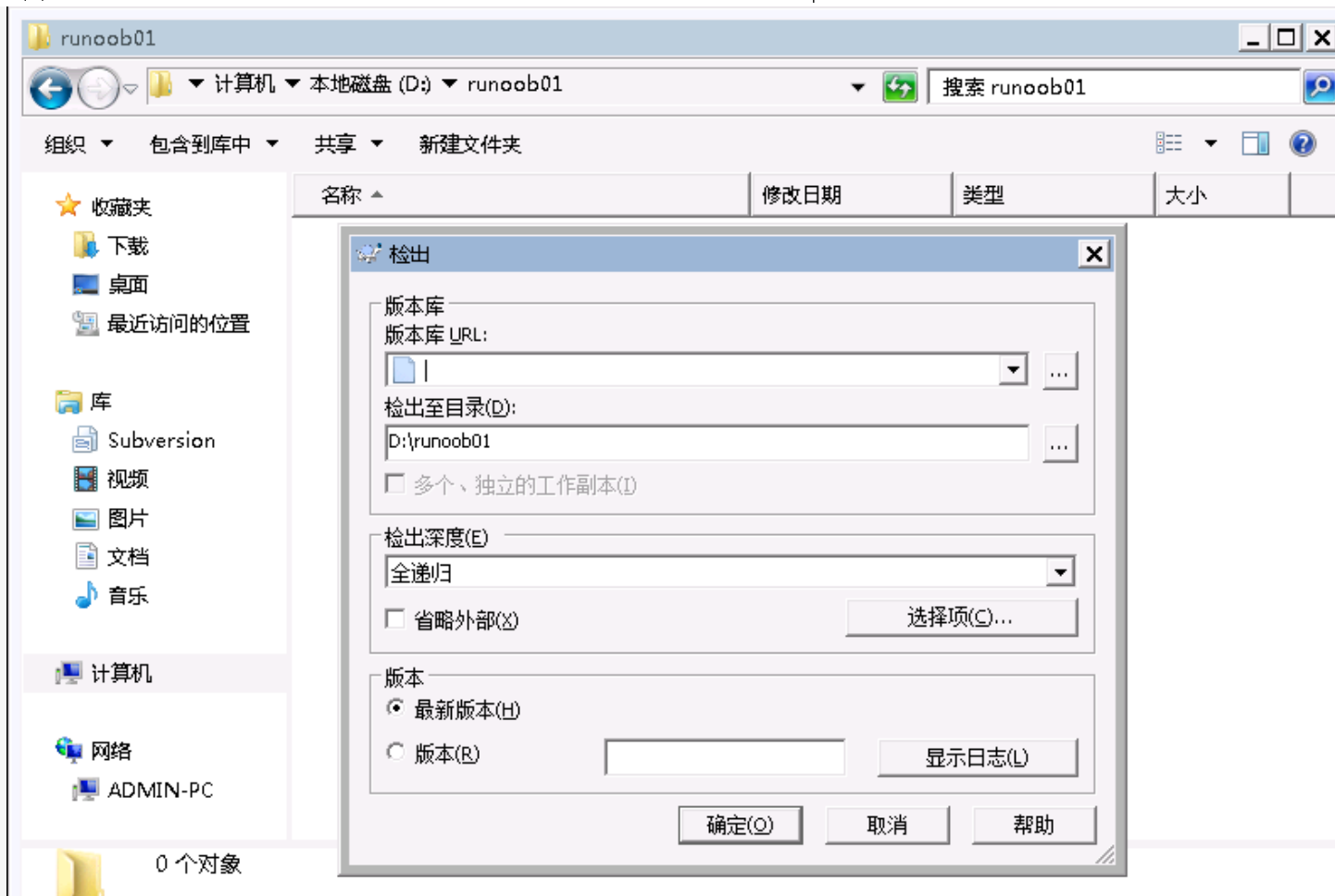
如果您要在一个已经存在的 SVN Server 上面 checkout 出上面的档案，您只需要给定正确的 SVN URL 以及要 checkout 目录的名称。就可以取得指定的档案及目录了。

新增档案及目录到 Repository 中 add commit

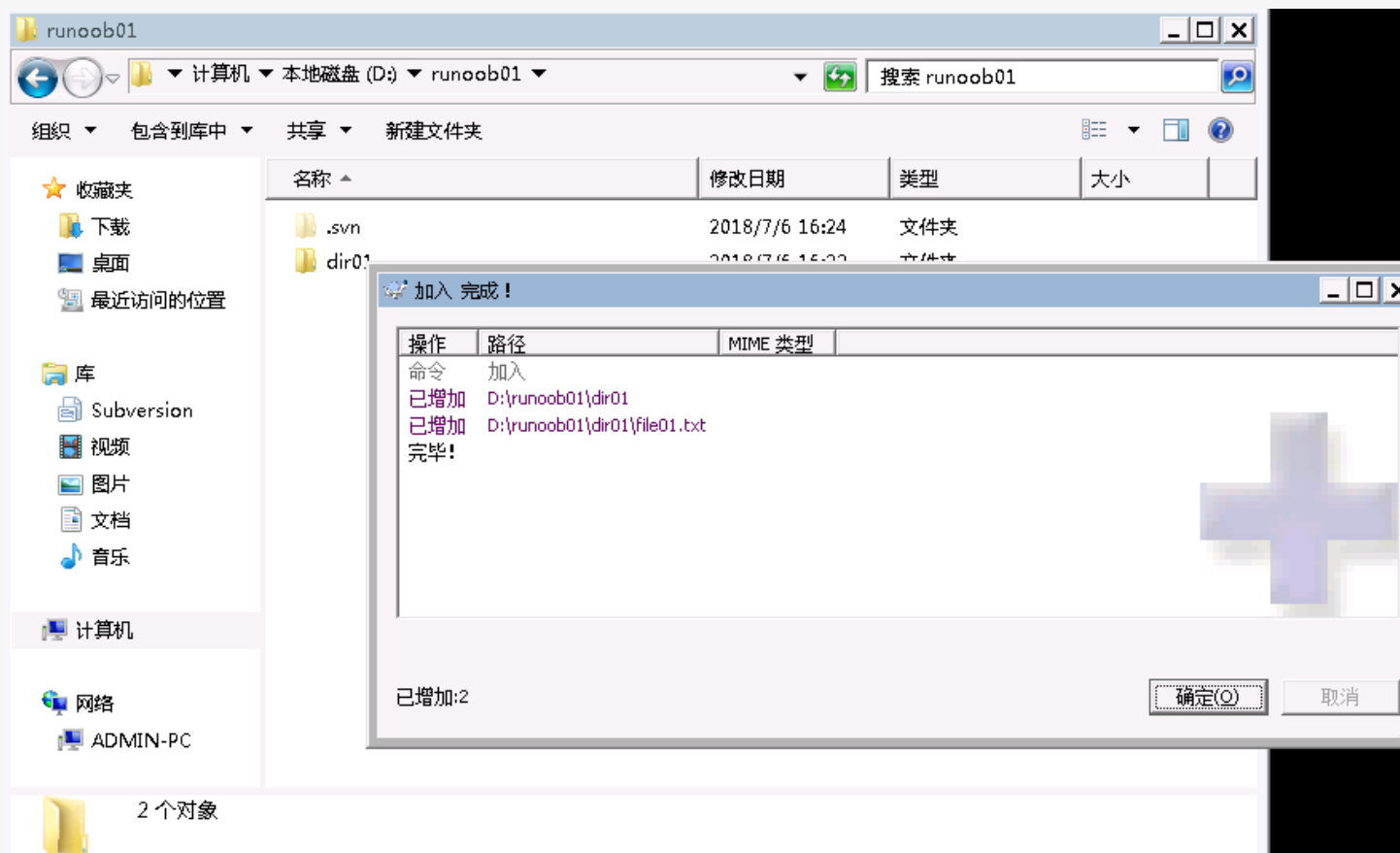
创建目录 dir01, 在目录里新增文件



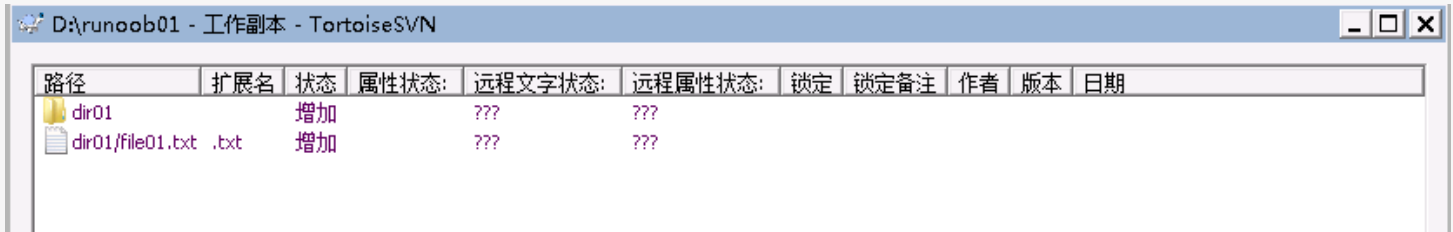
将新增的文件加入到 SVN 版本控制中，TortoiseSVN 会把准备要加入的档案及目录，勾选需要加入的文件。



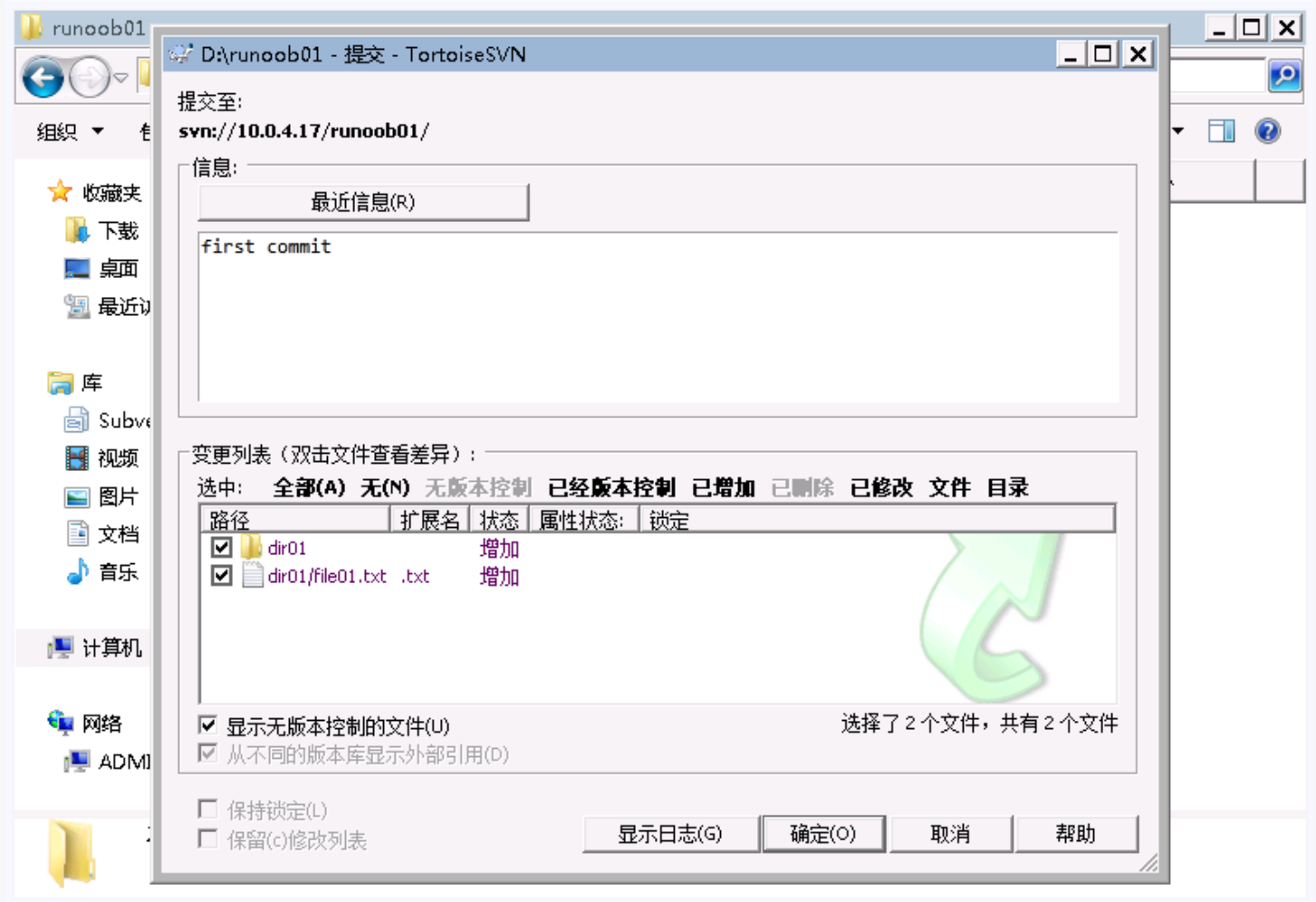
按下 OK 后，您将会看到如下的讯息窗口：



这个 Add(增加)的动作并未真正的将档案放到 Repository 中。仅仅是告知 SVN 准备要在 Repository 中放入这些档案。此时的文件状态为：

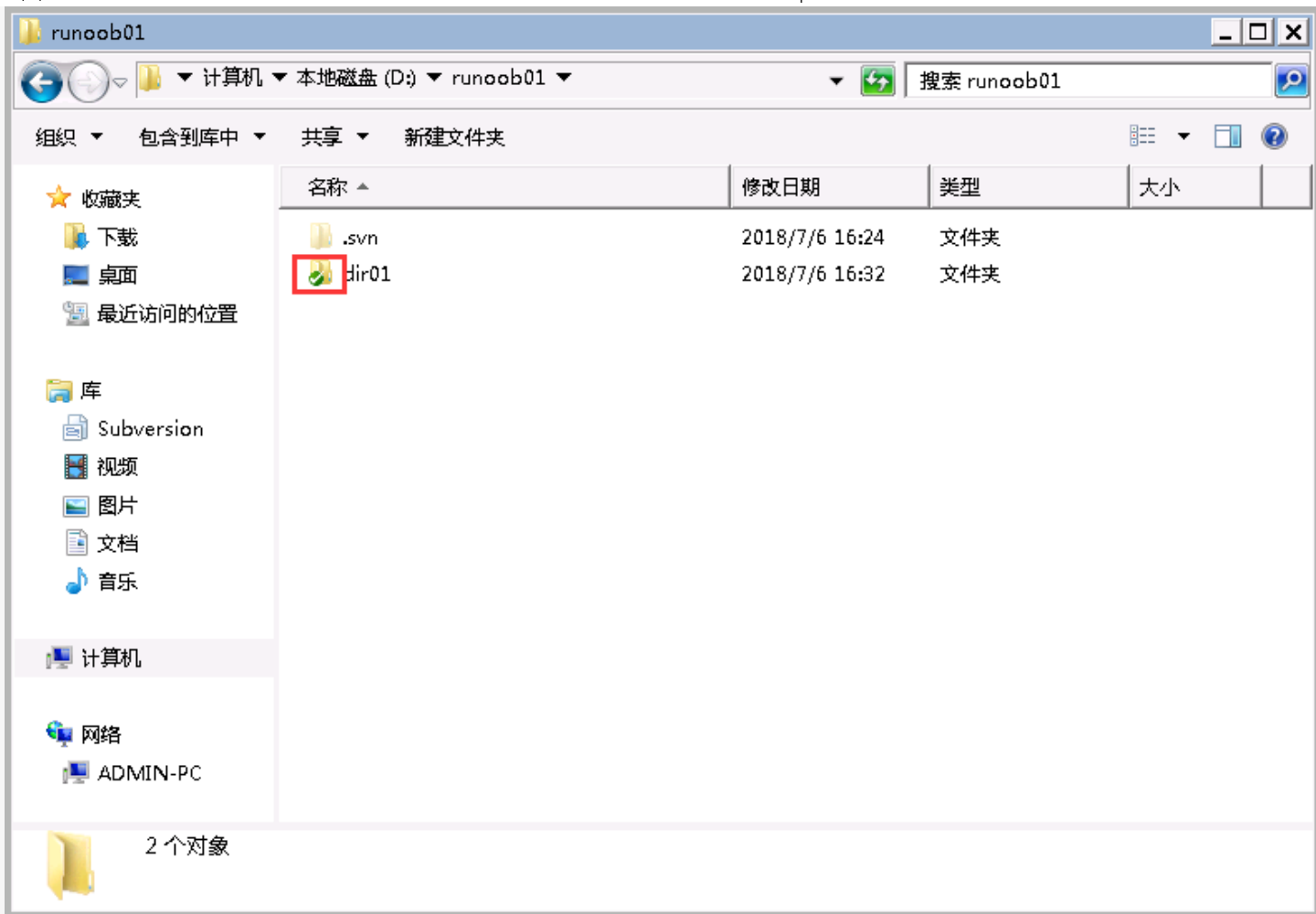


这些档案真正的放入到 Repository 中，空白处右键选择 SVN commit(提交) 紧接着，您将会看到如下的窗口出现：



在这里可以清楚地了解到哪些档案要被 commit 到 repository (版本库) 中。同样的，如果您有档案不想在这个时候 commit 到 Repository，您可以取消选取的档案，这样他们就不会被 commit 到 Repository 中。在"信息"文本框中可以写入对本次 commit 的说明。

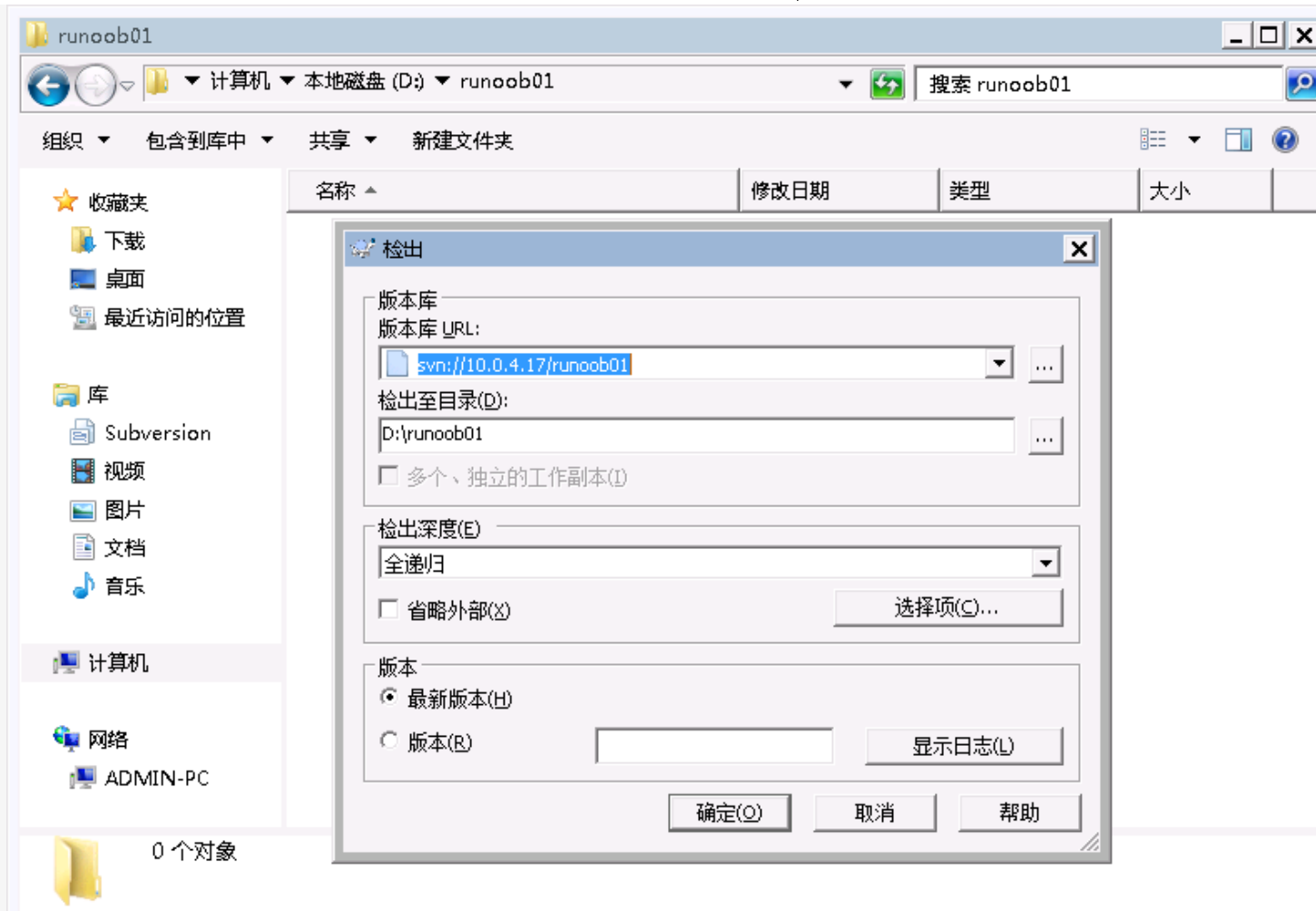
点击"确认"后完成 commit 动作，然后您可以到 runoob 目录中，确定是否所有的档案 icon 都有如下的绿色勾勾在上面，这样代表您的档案都正确无误的到 repository 中。



更新档案及目录 update

由于版本控制系统多半都是由许多人共同使用。所以，同样的档案可能还有人会去进行编辑。为了确保您工作目录中的档案与 Repository 中的档案是同步的。建议您在编辑前都先进行更新的动作。

在想要更新的档案或目录 icon 上面按下鼠标右键。并且选择 SVN Update。

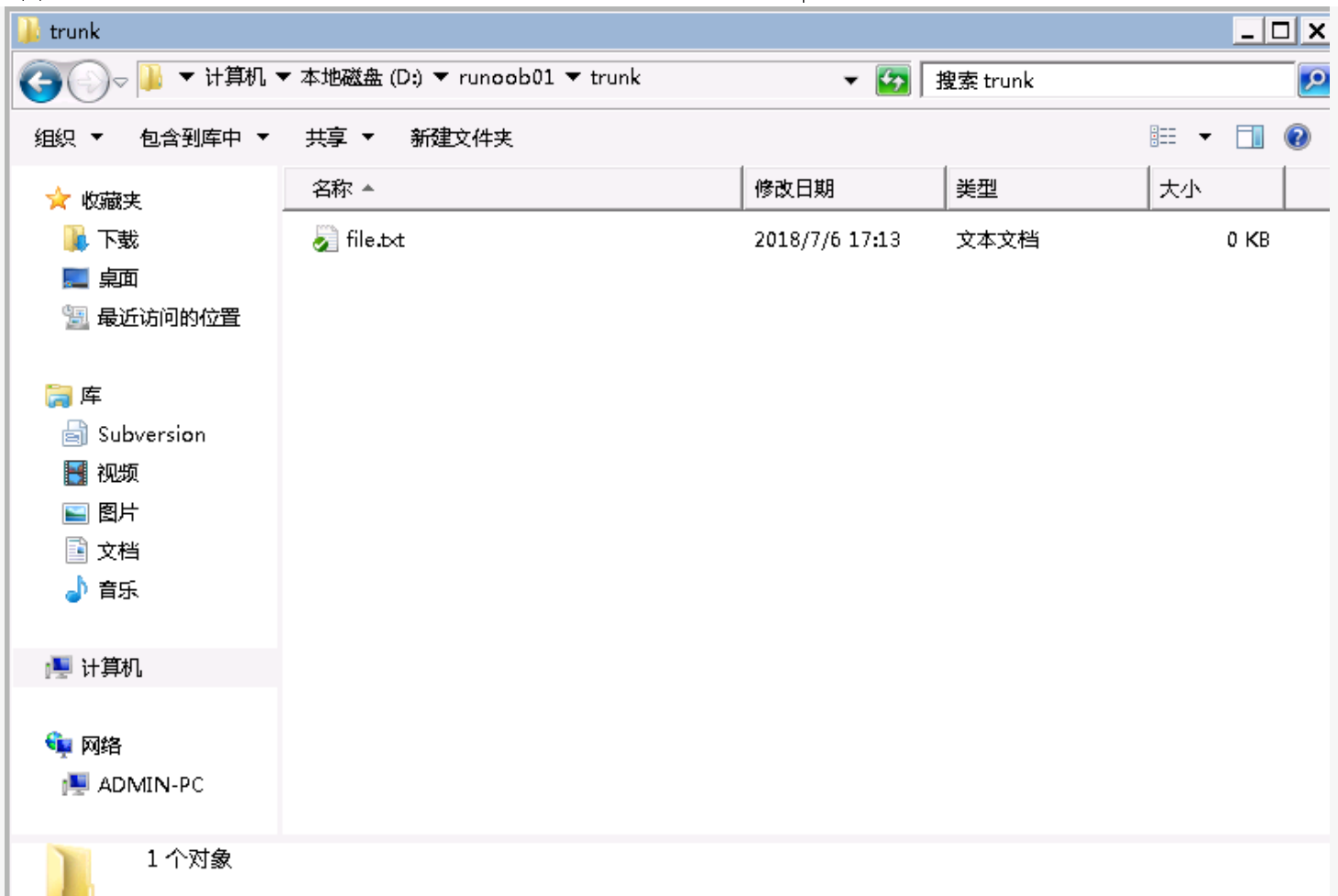


有时我们需要回溯至特定的日期或是版本，这时就可以利用 SVN 的 Update to revision 的功能。在想要更新的档案或目录 icon 上面按下鼠标右键。并且选择 TortoiseSVN->Update to revision(更新至版本)。

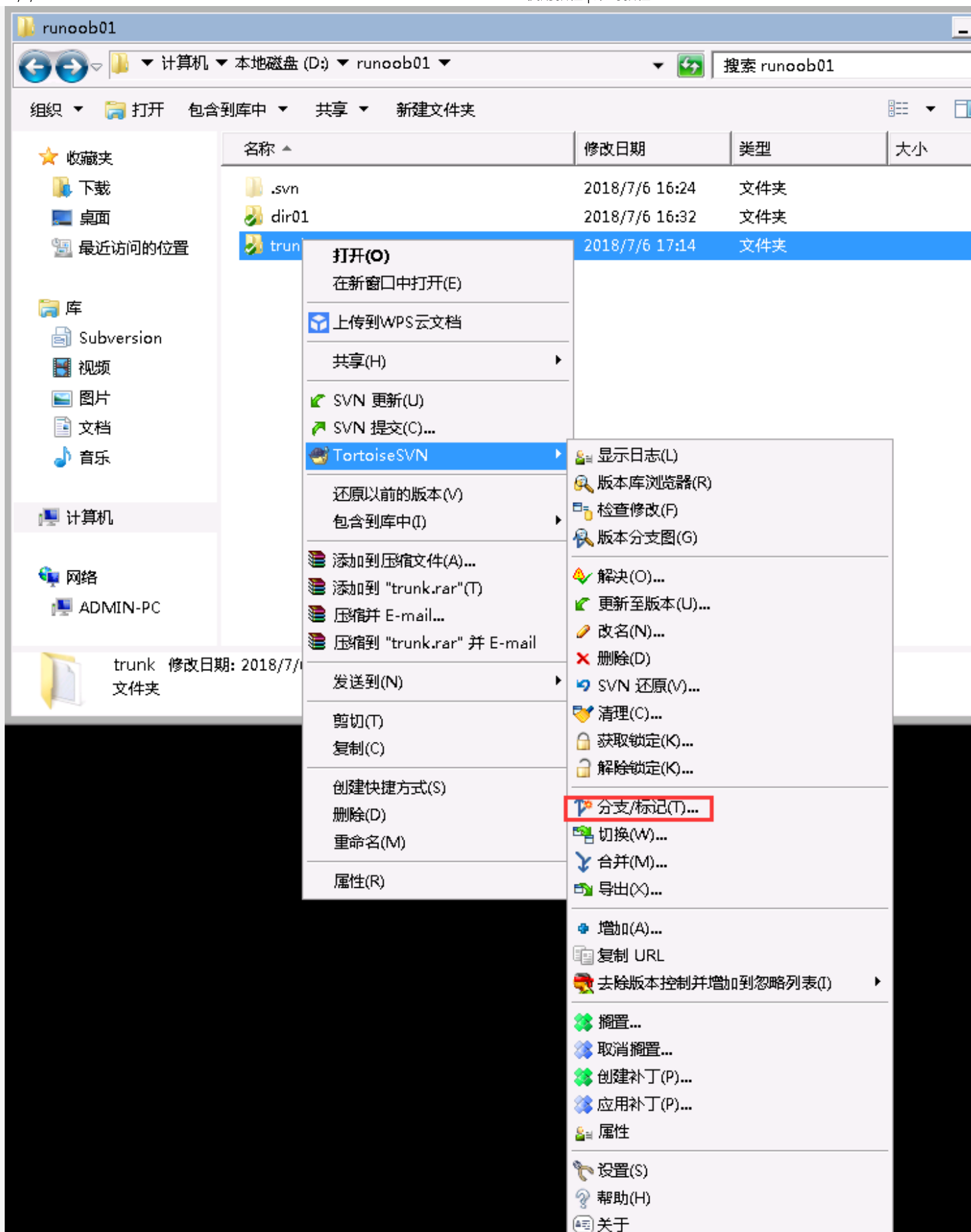


复制档案及目录 branch

很多时候您会希望有另外一个复制的目录来进行新的编修。等到确定这个分支的修改已经完毕了，再合并到原来的主要开发版本上。举例来说，我们目前在runoob01/trunk下面有如下的目录及档案：



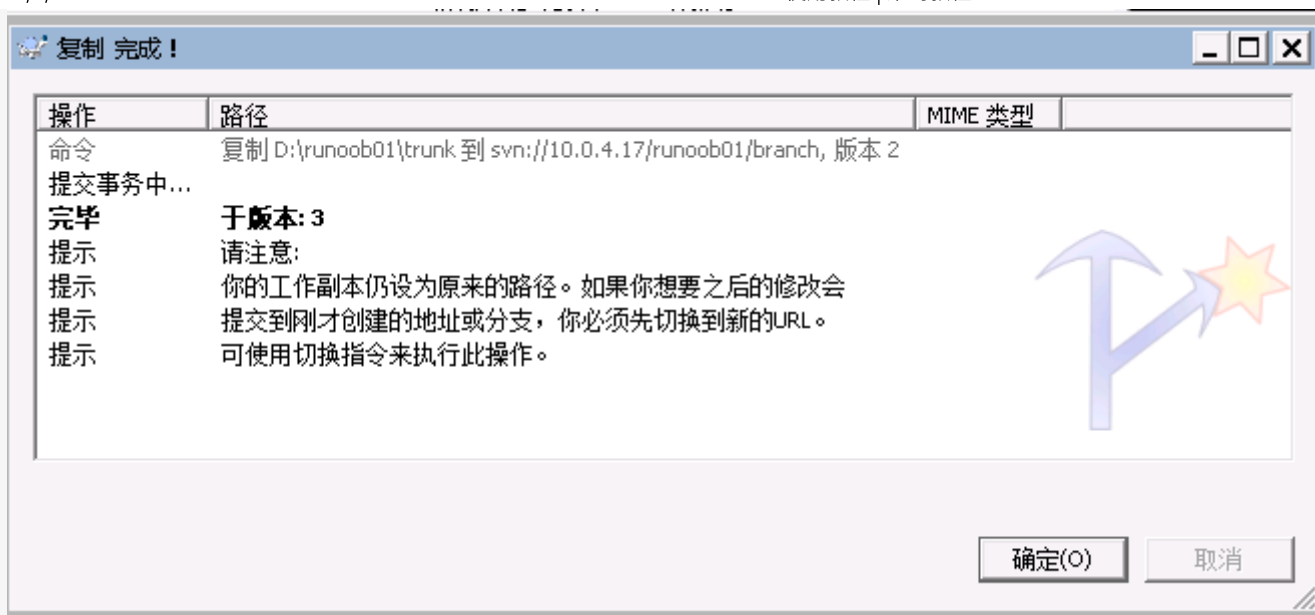
现在，我们要为 trunk 这个目录建立一个 branch。假设我们希望这个目录是在 D:\runoob01\branch。首先我们可以在 trunk 目录下面的空白处，或是直接在 trunk 的 icon 下面按下鼠标右键选择 Branch/Tag...(分支/标记)这个选项，您将会看到如下的对话框出现。





请先确认 From WC at URL(从工作副本/URL): 中的目录是您要复制的来源目录。接着, 在 To URL(至路径)中输入您要复制过去的路径。通常我们会将所有的 branch 集中在一个目录下面。以上面的例子来说, branch 档案都会集中在 branch 的子目录下面。在 To URL 中您只需要输入您要的目录即可。目录不存在时, 会由 SVN 帮您建立。特别需要注意的是 SVN 因为斜线作为目录分隔字符, 而非反斜线。接着在 Log message(日志信息)输入您此次 branch 的目的为何。按下 OK 就可以了。

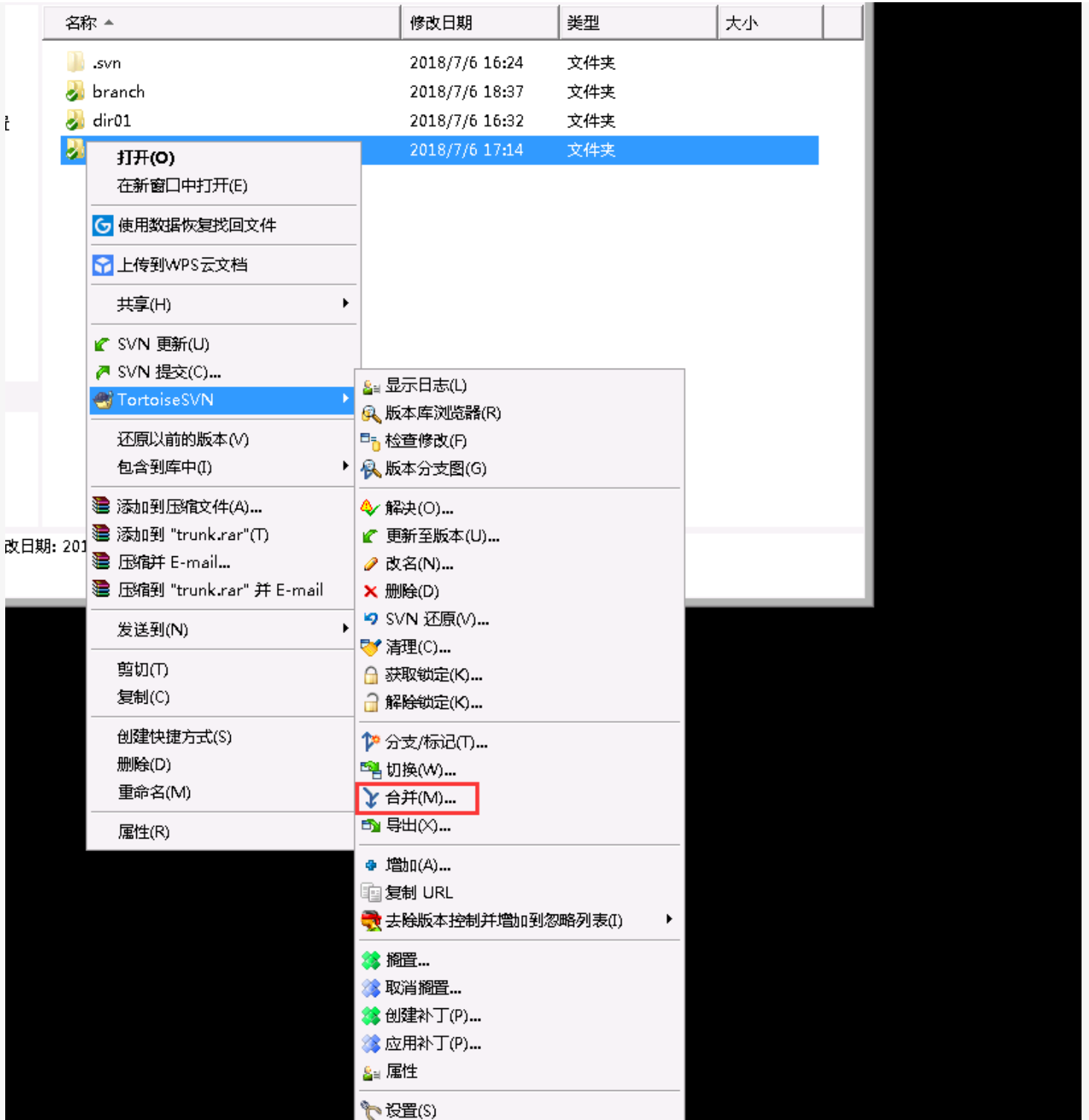
如果成功, 将可以看到下面的画面:



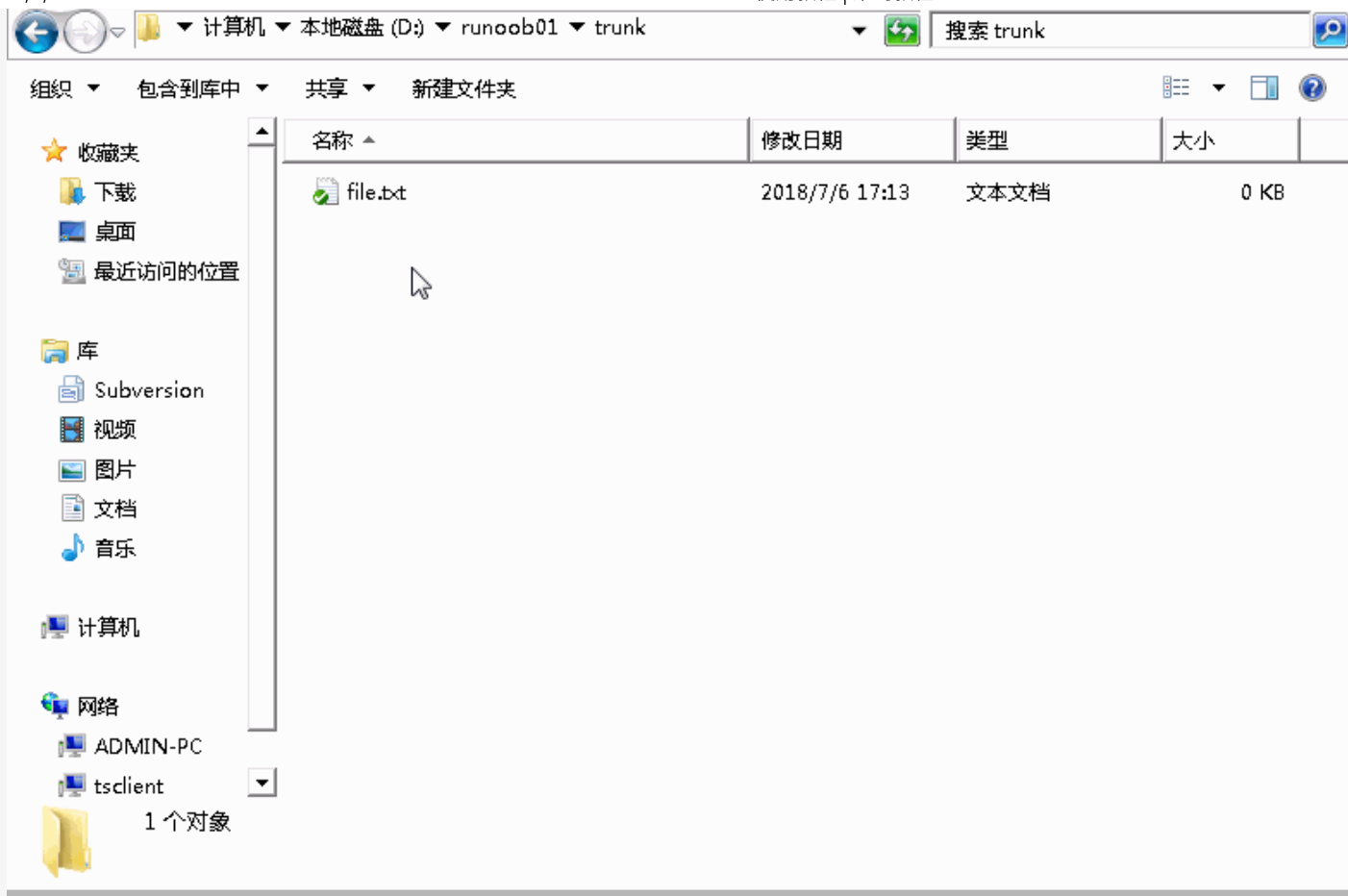
按下 OK 就可以关闭这个窗口了。如果您此时立刻去 runoob01 目录的 branch 子目录下面，您将会失望的发现在该目录下面并没有刚刚指定的目录存在。这是因为您 runoob01 目录的部份还是旧的，您只需要在 branch 子目录下面进行 SVN update 就可以看到这个新增的目录了。新增的目录就与原来的目录无关了。您可以任意对他进行编辑，一直到您确认好所有在 branch 下面该做的工作都完成后，您可以选择将这个 branch merge 回原来的 trunk 目录，或者是保留它在 branch 中。

合并动作 merge

假如我们在 branch 分支中对文件进行了修改或增加了文件，要 merge 回 trunk 目录中，方法很简单。以上面的例子来说，我们在 D:\runoob01\trunk目录空白处，按下鼠标右键，选择 Merge(合并):



这个画面主要分为三个部份，前面的 From: 与 To: 是要问您打算从 Branch 中的哪个版本到哪个版本，merge 回原来的 trunk 目录中。因此，From 跟 To 的 URL 字段应当都是指定原来 branch 的目录下。剩下的就是指定要 merge 的 revision 范围。以 上面的例子而言，我们从 Branch 的 Revision 7 开始 merge 到 Branch 下面的最新版本。您可以透过，Dry run 按钮，试作一 次 Merge。这个 merge 只会显示一些讯息，不会真正的更新到 trunk 的目录去。只有按下 Merge 按钮后，才会真正的将 branc h 的档案与 trunk 的档案合并起来。

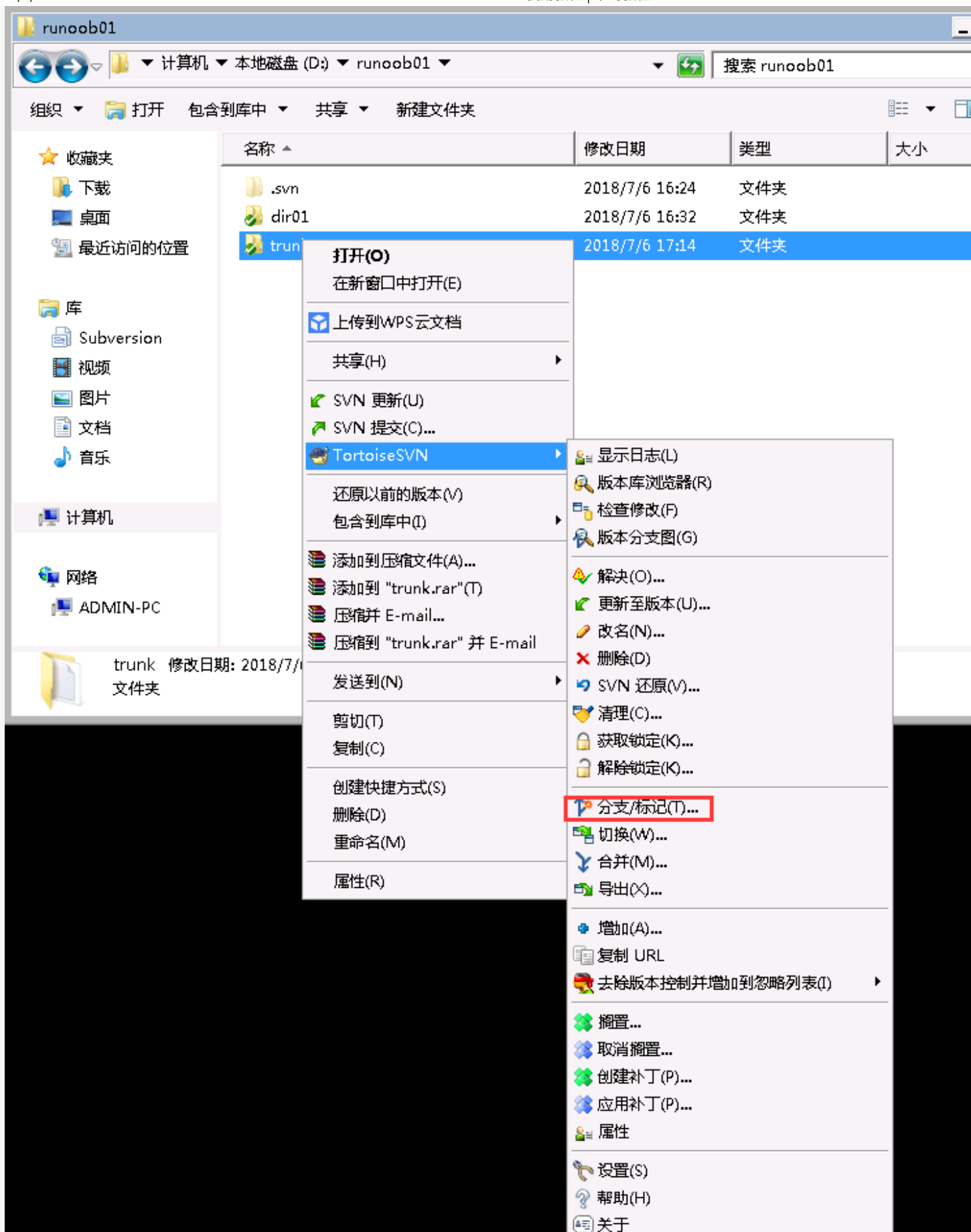


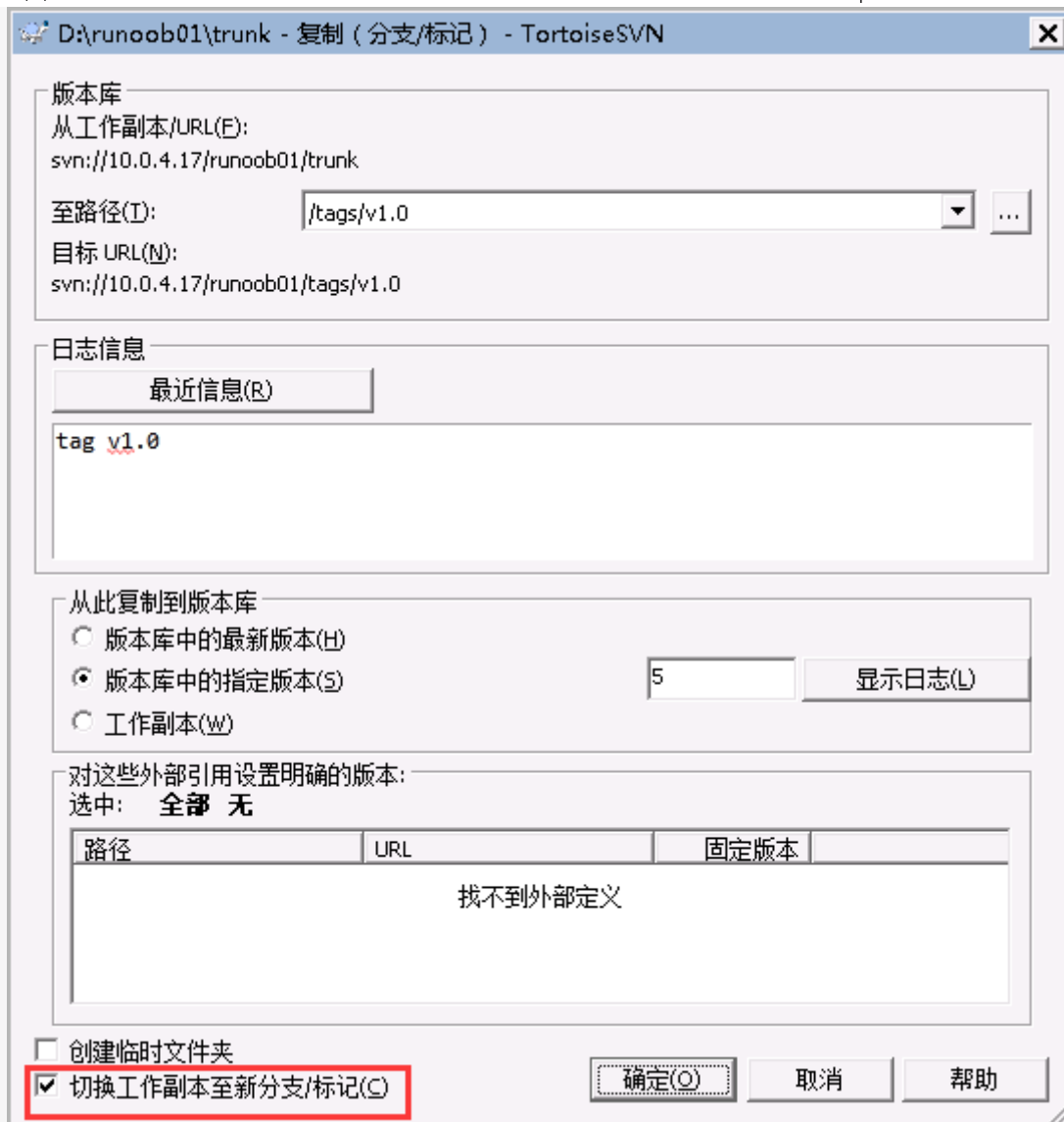
如果您确认这次的 merge 没有问题，您可以直接使用 commit 来将这两个被修改的档案 commit 回 SVN repository 上。如果有问题，您可以直接修改这两个档案，直到确认 ok 了，再行 commit。

制作 Tag 或是 Release

所谓的 Tag 或是 Release 就是一个特别的版本，因为这个版本可能有特别的意义。例如：这个版本是特别的 Milestone 或是 release 给客户的版本。其实，Tag 与 Release 的作法与 Branch 完全相同。只是 Branch 可能会需要 merge 回原来的 trunk 中，而 tag 及 release 大部分都不需要 merge 回 trunk 中。

举例来说，今天我们的 trunk 做了一版，这个版本被认定是软件的 1.0 版。1.0 版对于开发来说是一个非常重要的里程碑。所以我们要特别为他做一个标记，亦即 Tag。假设，这个 1.0 版是要正式 release 给客户或是相关 vendor，我们要可以为他做一个 Release 的标记。基本上，SVN 只有目录的概念，并没有什么 Tag 的用法。所以您会看到在 SVN 的选单上面，Branch 与 Tag 是同一个项目。以这个 1.0 的例子来说，我们在 runoob01 目录下创建 tags 目录用于存放打 tag 的版本，并提交到版本库，然后在 Trunk 上面，按下鼠标右键，选择 Branch/Tag 的项目：





成功的话，您就在对应的 Tag 目录下面建立了一个 v1.0 的目录。当然，如果您这时到 Tag 的目录下面去，会看不到这个目录，您需要在 Tag 目录下面 update 一下，才能看到它。

