# 一、svn的概念及使用

1. **svn的概念**

**1.1基本概念**

Svn是版本控制系统，可以帮助软件开发人员一起工作，让他们的工作也保持完整的历史。

1）允许开发同时工作。

2）不要覆盖对方的文件变化。

3）保持历史的每一个版本的一切。

（注意：每提交一次形成一个新的版本）

1.2下载与安装（略）

1. **维护工作文件**

**2.1与资源库同步**

与资源库同步的目的主要是比较工作副本与版本库里面的区别。

1. 查看工作副本修改错误，对比版本库，更新到本地工作副本
2. 提交之前对比与版本库的文件变化，确认无误时提交。

**2.2提交**

1）提交一次形成一个版本，对应一个版本号。（版本号有序递增）

2）提交需提交注释，以便其他人查看。

3）编译文件等一些系统生成的文件不要提交。

**2.3 更新**

1）因为多名开发人员共同开发同一个项目，有必要更新一些其他人提交的信息。

2）更新之前建议先与资源库同步，比较差异，以免覆盖自己有用的文件。

# 二、分支与合并

**1、创建分支的目的**

简单说，分支就是用于区分开发版本与当前发布版本的。

   1） 主干负责新功能的开发

    2）分支负责修正当前发布版本的bug（对于可以放入下个发布版本的改进性bug可以直接在主干上开发）

    3）分支上修改的bug，经常性merge到主干上，尽量及时merge（避免大面积红色区域）。

    4）只能分支往主干靠拢（merge），不能反向！

5）直到下个新版本发布，该分支停止修改

**2、分支与合并的基本概念**

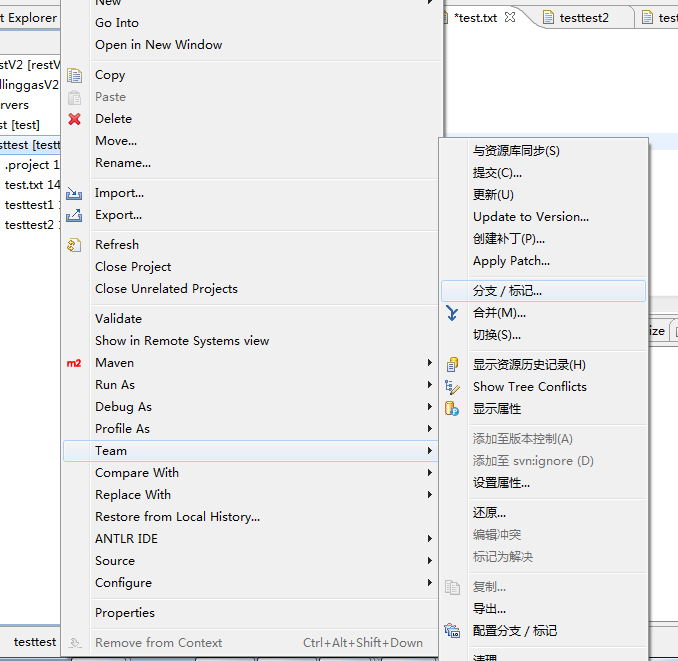
**分支：**版本控制系统的一个特性是能够把各种修改分离出来放在开发品的一个分割线上。这条线被称为分支。分支经常被用来试验新的特性，而不会对开发有编译错误的干扰。当新的特性足够稳定之后，开发品的分支就可以混合回主分支里(主干线)。

**合并：**分支用来维护独立的开发支线，在一些阶段，你可能需要将分支上的修改合并到最新版本，或者将最新版本的修改合并到分支。

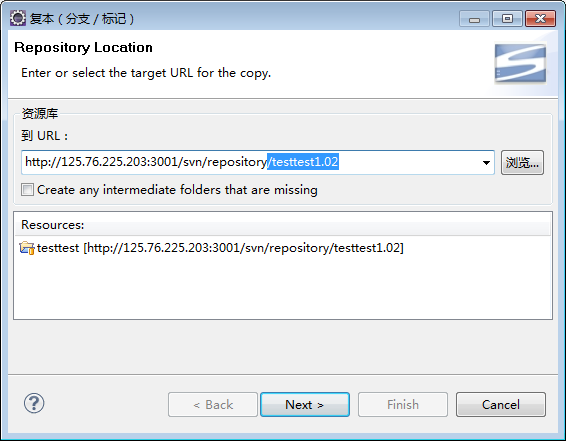
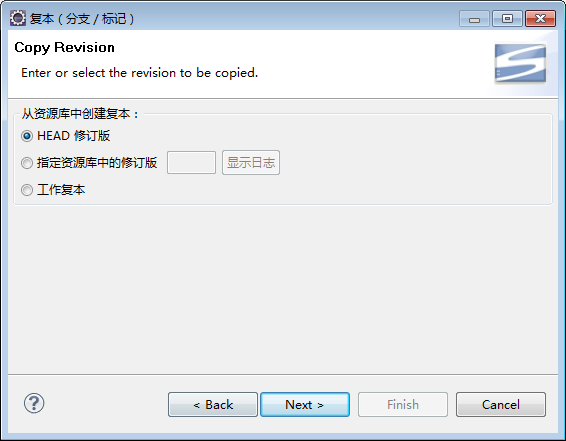
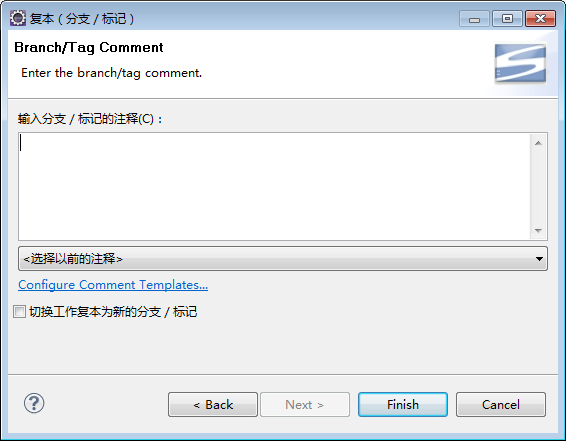
1. **建立分支与合并分支**

## 3.1建立分支

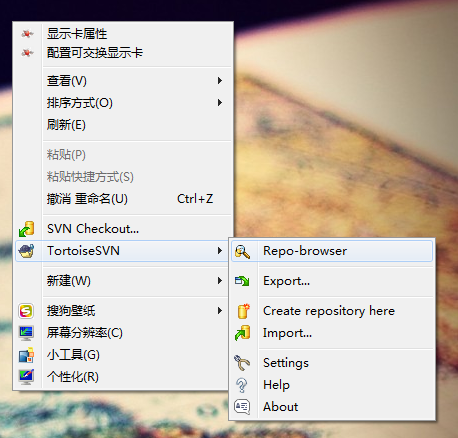
右击项目》》team》》选择分支/标记

创建目录名及位置

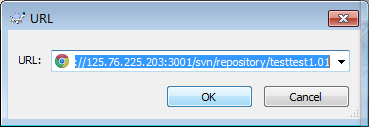
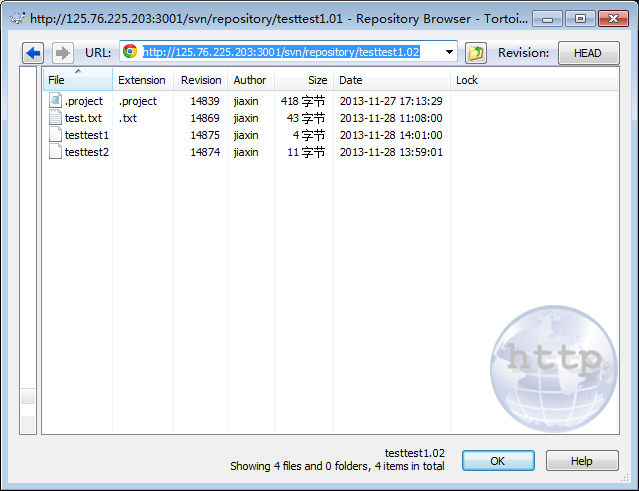


 选择要标记/分支的版本（HEAD是最新版，可以指定版本） 为创建分支添加注释

2、查看资源库中是否分支完成。右击桌面可以，》》TOrtoiseSVN》》Repo-browser



填写连接地址

如上图，testtest1.02已经存在资源库中。

## 3.2、分支合并

### 1.1对svn分支合并类型的理解：

合并的工作是把主干或者分支上合并范围内的所有改动列出，并对比当前工作副本的内容，由合并者手工修改冲突，然后提交到服务器的相应目录里。如果当前工作副本是主干，则合并的范围是分支上的改动，如果工作副本是分支的，则合并范围是主干上的改动，并且一定要注意，合并的起始位置URL一定要和当前的工作副本的URL是相同的。

**1)合并一个范围的版本**

此类型应用最为广泛，主要是把分支中的修改合并到主干上来。在主干上点击右键选择合并，然后选择合并类型：合并一个范围的版本。合并的源URL填写的是要合并的分支的URL，待合并的版本范围如果为空，则指的是合并分支上所有的版本，即自从分支创建以来到分支当前最新版本的所有演变。如果只是选择其中一个版本，或者几个版本，那么就表示只是将制定的n个版本的变化合并到主干上。如果只是选择其中一个版本，那么表示只是选择那个版本的修改，之前或之后的修改将不被采纳。

**2)复兴合并**

复兴合并可以理解为是第一种合并类型的一种特例，在复兴合并中，主干可以理解为是自从开创分支之后没有任何修改，而分支是经过修改的，而且合并中分支是没有版本选择的。经过复兴合并，分支中所有的修改都会合并到主干中，合并的结果将使得分支和主干一模一样，从而可以删除分支。

**3)合并两个不同的树**

此类型与前两种类型不同，第一种类型可以选择分支合并的版本，主干不能选择版本；第二种类型是主干和分支都不能选择合并的版本；而这种类型则是无论是主干还是分支都可以选择合并的版本，即可以选择过去的一个主干版本与分支的某个版本进行合并。合并的时候以选择的分支版本为主，如果选择的主干版本与分支版本有不同的地方，合并时主干部分将被放弃。

>起始URL：选择主干目录的URL（应当和当前工作副本的URL一致，这个是所谓的合并点）

>结束URL：选择要合并的分支的URL。

>起始和结束的版本：一般起始版本应当找到最后一次同步时的版本，如果从没有同步过（第一次合并），则选择创建分支时的版本，结束版本一般是最新版本，如果你不想将某些内容合并进主干的话，也可以选择一个合并点。

### 1.2 SVN分支深度合并的理解：

1）工作副本：即你当前的工作目录，一般默认为这个选项；

2）全递归：即你选择的目录的版本库，包括了其下面的子文件，子文件夹，包括子文件夹里面的内容；

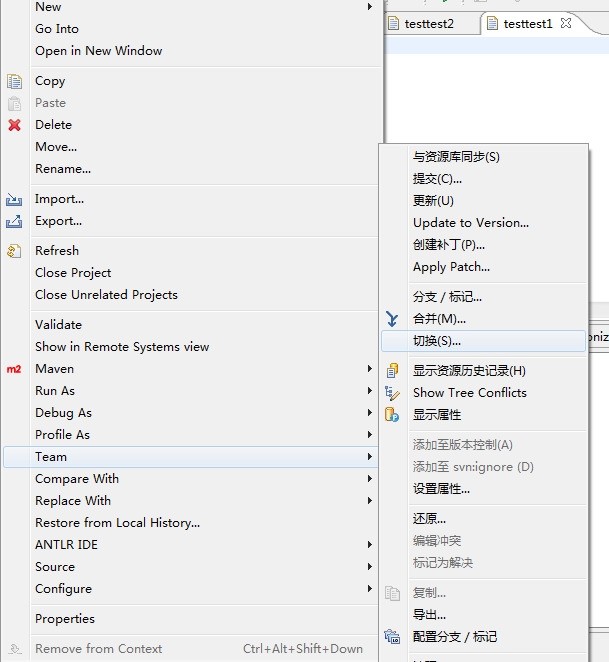
3）直接子节点，包括文件夹：即你选择的目录下面的文件，文件夹，但是不包括文件夹里面的子文件，子文件夹；

4）仅文件子节点：即你选择的目录下面的文件，但不包括文件夹，当然不包括的文件夹下面的所有内容也都不纳入合并范围；

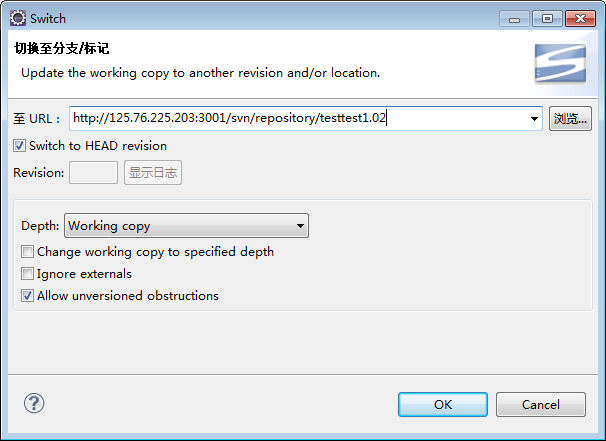
5）仅此项：没有任何合并内容。

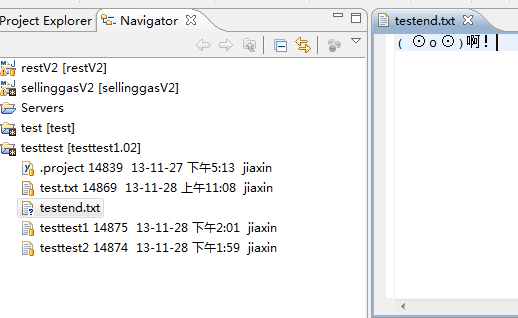
## 3.3分支与合并示列

首先切换到testtest1.02.右击项目》》team》》切换

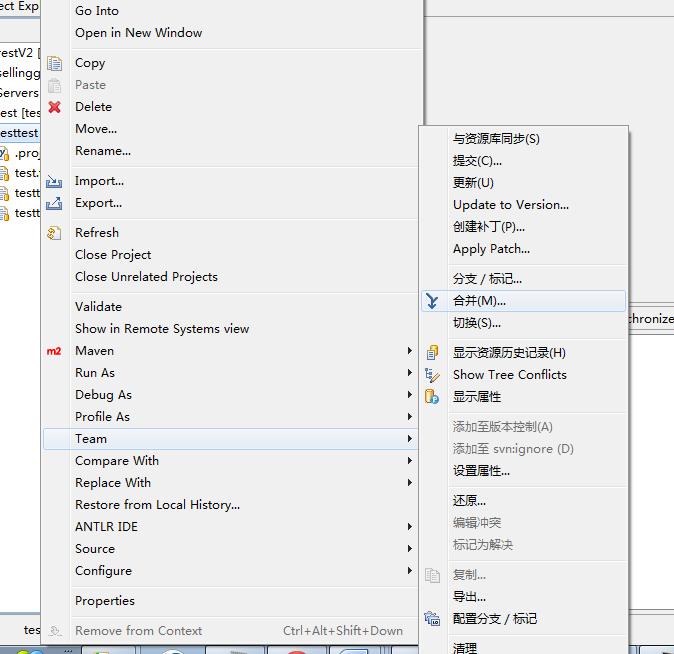


选择testtest1.02

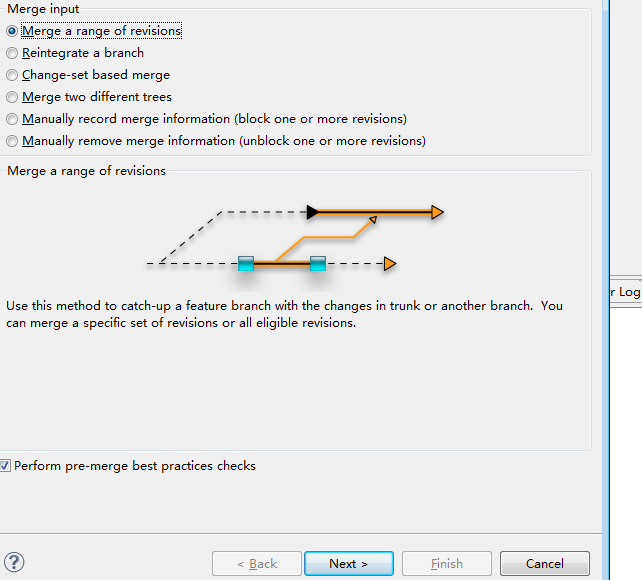
新增文件testend.txt 并文件内容

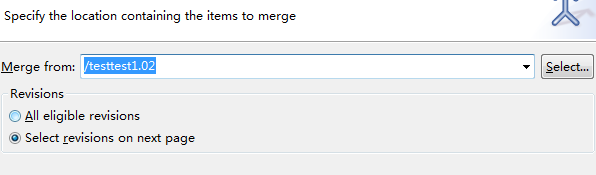


切换回testtest》》右击》》team》》合并

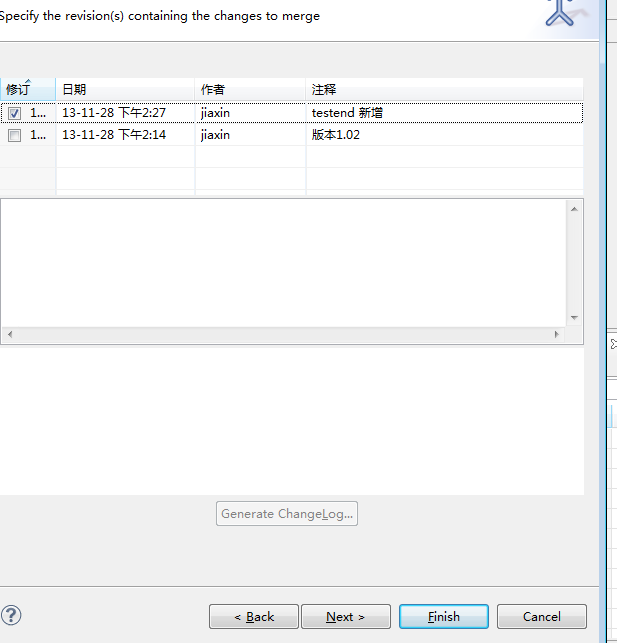


选择合并类型

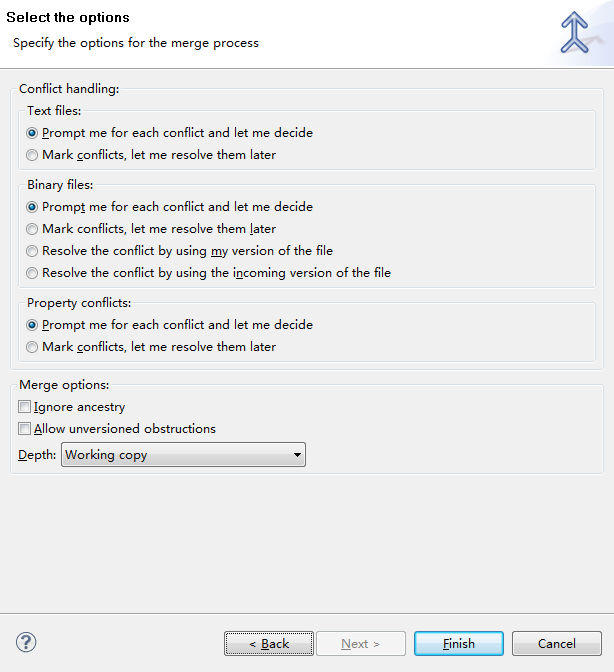


选择要合并的分支如下（testtest1.02） 

选择合并版本



选择冲突处理办法以及合并深度



如下图合并成功，testtest副本出现了新增的testend.txt文件。提交后便同步到主干资源库。

