

使用 Subversion 构建版本控制环境

(Windows 2000 Server)

起草	Tim
时间	2005-6-14
版本	V1.0

摘要:

在软件开发中, *CVS* 在很长的一段时间内一直都是版本控制的首选。但是现在用户有了另一个或许是更好的选择 *SVN* (*subversion*)。 *SVN* 是下一代版本控制系统, 完全可以替代 *CVS*, *CVS* 下的项目, 也可以非常容易的转到 *SVN* 的控制之下 (*SVN* 项目主页是 <http://subversion.tigris.org>, *Tigris* 本身就是一些在 *SVN* 控制下的开源项目, 可以说是 *sourceforge* 的 *SVN* 版本)。

SVN 是一个自由、开放源码、跨平台的版本控制系统。它是一个通用系统, 可用来管理任何类型的文件, 其中包括程序源码。

SVN 的设计目标很明确, 实现绝大部分 *CVS* 的已有功能, 充分考虑现有的 *CVS* 用户, 在使用方式上模仿 *CVS*, 同时开发了一系列工具, 使得基于 *CVS* 的项目能够顺利迁移到 *SVN* 上。作为新一代的版本控制系统, 和 *CVS* 相比, 它有很多优点, 例如目录版本控制、不可分割的提交、一致的数据处理方式和更有效率的分支与标记等。

现在, 好多开源项目已经开始转移到了 *SVN* 控制之下。作为软件开发者, 使用 *Subversion* 控制所开发的项目是必然趋势, 本文即以 **Apache+SVN** 为例, 介绍了 *SVN* 环境的搭建以及简单使用方法。

1 准备相关软件

1.1 Subversion 服务器 [svn-win32-1.3.2.zip](#)

建议直接到 <http://subversion.tigris.org/> 下载最新版本

1.2 Subversion 客户端 [TortoiseSVN-1.3.5.6804-svn-1.3.2.msi](#)

TortoiseSVN 是一个非常不错的 SVN 客户端, 可以实现跟资源管理器的紧密结合。建议到 <http://tortoisesevn.tigris.org/> 下载最新版 (相信很多使用过 CVS 的朋友应该用过 TortoiseCVS 吧)。

1.1 VS.NET 使用 SVN 的插件 [AnkhSetup-0.5.3.1636.msi](#)

AnkhSVN 是一个相当不错的 VS.NET 的 SVN 插件, 可以到 AnkhSVN 的官方网站下载最新版本 <http://ankhsvn.tigris.org/>。有关 AnkhSVN 的更详细的信息, 请访问 <http://ankhsvn.tigris.org/screenshots.html>。

1.2 Apache Web 服务器 ([svn-1.3.2](#) 建议使用 [apache-2.0.54](#))。

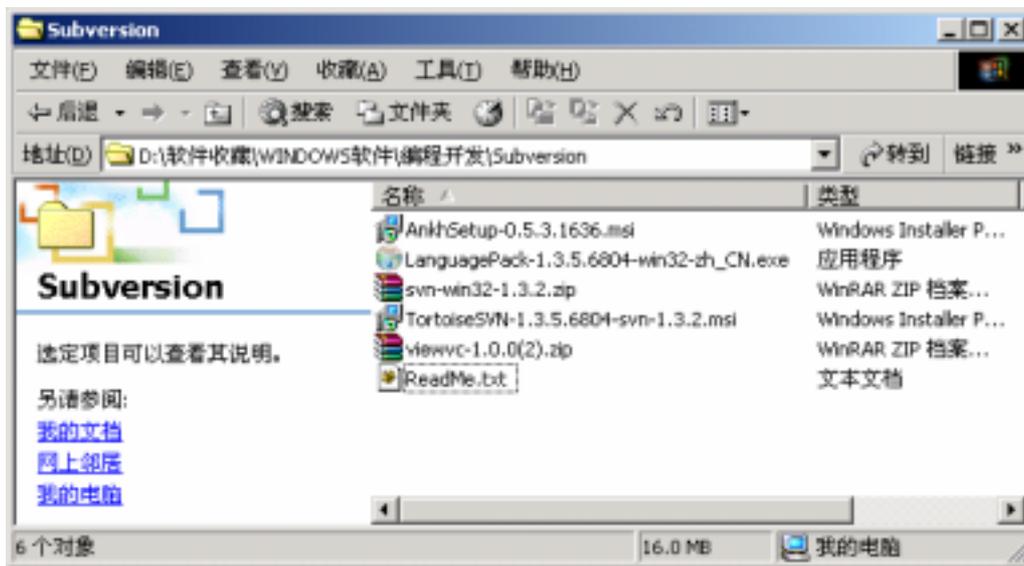
可以到 <http://www.apache.org/dist/> 下载合适的版本, 最好跟 subversion 软件中使用的版本一致, 若要求的 apache 版本比较旧, 可以到 apache 历史版本页面下载 <http://archive.apache.org/dist/>。

1.3 TortoiseSVN 简体中文语言包 ([LanguagePack-1.3.5.6804-win32-zh_CN.exe](#))

请到 <http://tortoisesevn.tigris.org/download.html> 选择下载相应的版本。

小技巧:

以上软件下载完了后, 不妨放到同一文件夹下, 并附加一包含软件简单说明的 *readme.txt* 文件, 以备日后使用(如图一)。



图一、文件分类存放

2 软件安装

2.1 安装 apache

Apache 的安装相当简单、值得注意的一点是，若已经安装了 IIS，80 端口将被占用，请先停止，可以使用以下命令：

```
Net stop w3svc
```



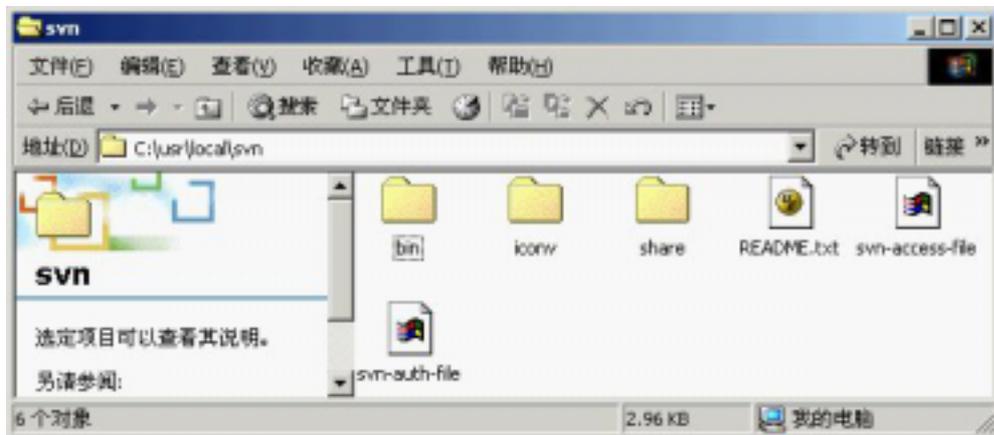
图二、停止 IIS

2.2 安装 TortoiseSVN & TortoiseSVN 简体中文语言包

安装没什么特别之处，只要先安装 TortoiseSVN 在安装语言包即可。

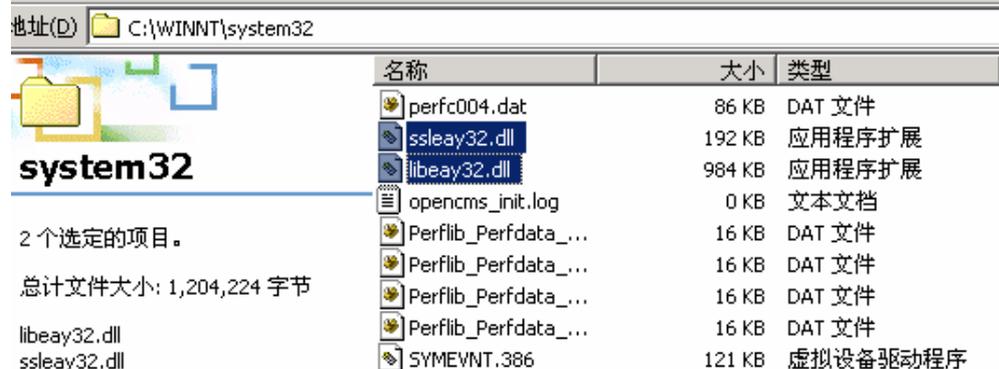
2.3 安装 Subversion.

2.3.1 解压下载得到的 [svn-win32-1.3.2.zip](#) 到 c:/usr/local/ 目录。并重命名为 svn，即可 /usr/local/svn/;



图三、解压 svn-win32-1.3.2.zip

2.3.2 拷贝 libeay32.dll、ssleay32.dll 两个 DLL 件到 c:/winnt/system32 目录；



图四、拷贝 OPEN SSL 加密库

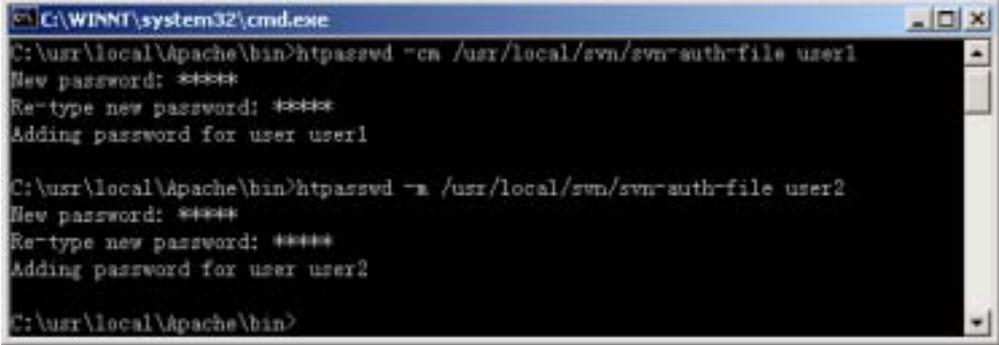
- 2.3.3 拷贝 libapriconv.dll、libdb43.dll、intl3_svn.dll 到 apache/bin 目录；
- 2.3.4 拷贝 mod_dav_svn.so、mod_authz_svn.so 两个文件到 apache 的 Modules/ 目录；
- 2.3.5 在 apache 的 conf/ 目录下建立一个文本文件 mod_svn.conf 输入以下内容

```
#加载相应模块
LoadModule dav_svn_module /usr/local/svn/bin/mod_dav_svn.so
LoadModule authz_svn_module /usr/local/svn/bin/mod_authz_svn.so
#LoadModule dav_svn_module modules/mod_dav_svn.so
#LoadModule authz_svn_module modules/mod_authz_svn.so

#配置仓库
<Location /svn>
    DAV svn
    SVNParentPath /usr/local/repository#仓库父目录位于
    c:/usr/local/repository
    # SVNPath /usr/local/respository/svn1 #仓库目录
</Location>
```

- 2.3.6 若需要使用 Apache 验证则还需要做以下调整：

```
进入 apache/bin 目录，执行以下命令产生密码验证文件
htpasswd -cm /usr/local/svn/svn-auth-file user1
提示输入密码时输入该用户的密码
若还需要增加用户 user2，则执行以下命令
htpasswd -m /usr/local/svn/svn-auth-file user2
删除某个用户（user2），可以使用下列指令：
htpasswd -D /usr/local/svn/svn-auth-file user2
```



```
C:\WINNT\system32\cmd.exe
C:\usr\local\apache\bin>htpasswd -cm /usr/local/svn/svn-auth-file user1
New password: ****
Re-type new password: ****
Adding password for user user1

C:\usr\local\apache\bin>htpasswd -m /usr/local/svn/svn-auth-file user2
New password: ****
Re-type new password: ****
Adding password for user user2

C:\usr\local\apache\bin>
```

图五、使用 htpasswd 产生密码文件

- 2.3.7 启用验证后，SVN 相关配置如下

```
<Location /svn>
    DAV svn
    SVNParentPath /usr/local/repository#仓库父目录位于
    c:/usr/local/repository
    AuthType Basic
    AuthName "Subversion repository"
    AuthUserFile /usr/local/svn/svn-auth-file
```

```

Require valid-user
AuthzSVNAccessFile /usr/local/svn/svn-access-file
</Location>

```

- 2.3.8 配置文件中使用到了 `svn-access-file`，这个文件是用来控制用户访问 SVN 仓库权限的，下面就讲述以下 `access` 文件的配置。

```

[svn1: /]           //这表示，仓库 svn1 的根目录下的访问权限
user1 = rw         // svn1 仓库 user1 用户具有读和写权限
user2 = r          // svn1 仓库 user2 用户具有读权限
[svn2: /]           //svn2 仓库根目录下的访问权限
user1 = r          // user1 用户在 svn2 仓库根目录下只有读权限
user2 =            // user2 用户在 svn2 仓库根目录下无任何权限
#[/]               // 这个表示在所有仓库的根目录下
#* = r             // 这个表示对所有的用户都具有读权限
#[groups]          // 这个表示群组设置
#svn1-devel opers = user1,user2 // 这个表示某群组里的成员
#svn2-devel opers = user1
#[svn1: /]
#@svn1-devel opers = rw// 如果在前面加上@符号，则表示这是个组权限设置

```

2.3.9 重启 APACHE

注意：`DocumentRoot` 目录与 `svn` 的 `SVNParentPath` 值不可设置为一样，否则可能会出现 `301 Moved Permanently...` 错误

3 软件使用

3.1 、使用命令行 (CMD 窗口模式)

创建新的版本库

```
svnadmin create /path/to/repository
```

svnadmin 是有用的 subversion 系统管理命令, 使用 svnadmin help 查看在线帮助信息。

添加新的项目 (import new project)

```
svn import project file:///repository_name/project -m "Comment"
```

```
svn import project http://yourhost/svn_dir/repository_name/project -m "Comment"
```

检出 (checkout) 项目

```
svn checkout file:///repository_name/project/trunk project
```

```
svn checkout http://yourhost/svn_dir/repository_name/project/trunk project
```

列出仓库中的项目 (list)

```
svn list --verbose file:///repository_name/project
```

```
svn list --verbose http://host/svn_dir/repository_name/project
```

状态查询 (status)

```
svn status
```

给出新文件, 已经改变的文件和被删除的文件列表;

添加文件或目录 (add)

```
svn add
```

删除文件或目录 (delete)

```
svn delete
```

```
svn delete http://yourhost/svn_dir/repository/project_dir
```

提交 (commit)

```
svn commit http://yourhost/svn_dir/repository/project_dir
```

更新 (update)

```
svn update
```

更新仓库中的文件到本地。

标签和特殊版本项目

使用 copy 命令

```
svn copy http://yourhost/repos/project/trunk
```

```
http://yourhost/repos/project/tags/1.0.0
```

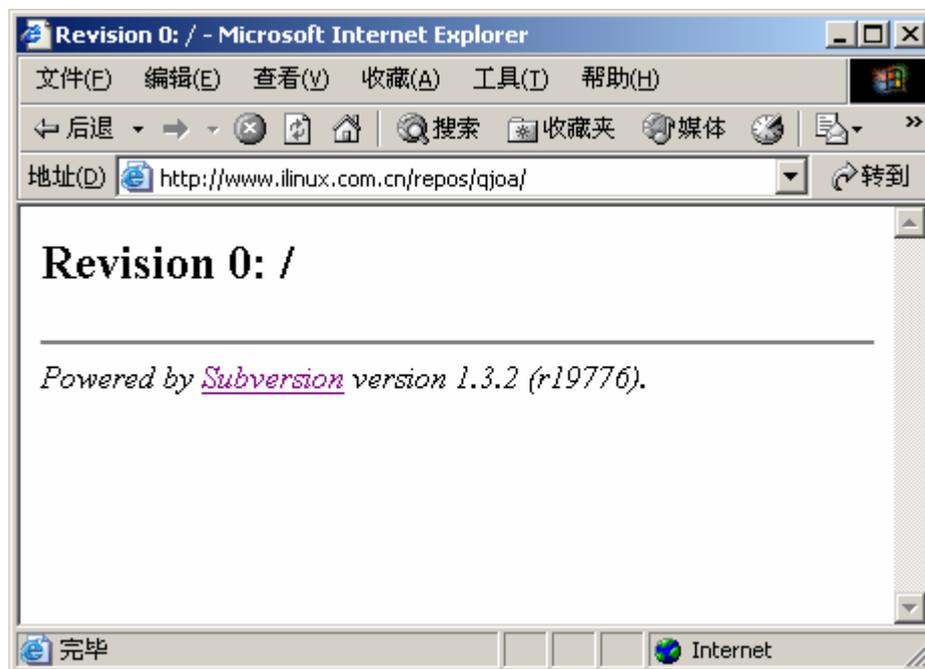
用于创建某个特定版本的快照 (snapshot) ;

```
svn list http://yourhost/repos/project/tags/1.0.0
```

3.2 、下面我们就用以上介绍的命令创建一个测试仓库

3.2.1 打开命令提示符, 进入 E:/repository 目录, 执行以下命令

3.2.2 重启 apache, 浏览 <http://localhost/repos/qj oa>, 若看到以下画面, 则表示 SVN 已经正常工作



图六、SVN 开始工作了

3.2.3 刚才建立的仓库只是一个空的仓库, 里面没有任何的项目/文件, 下面, 我将演示一下怎么向该仓库添加项目, 使用以下命令

```
svn import D:\WorkSpace\OTH_007_QJTech-0A
http://localhost/repos/qj oa -m "SVN init project"
```

注意: 以上命令必须单行输入, import 项目时, 最好先去掉 apache 验证, 否则将导致 import 项目失败!

3.2.4 OK, 重新浏览 <http://localhost/repos/qj oa> 试试, 若出现跟下图类似的画面, 则表示项目导入成功;

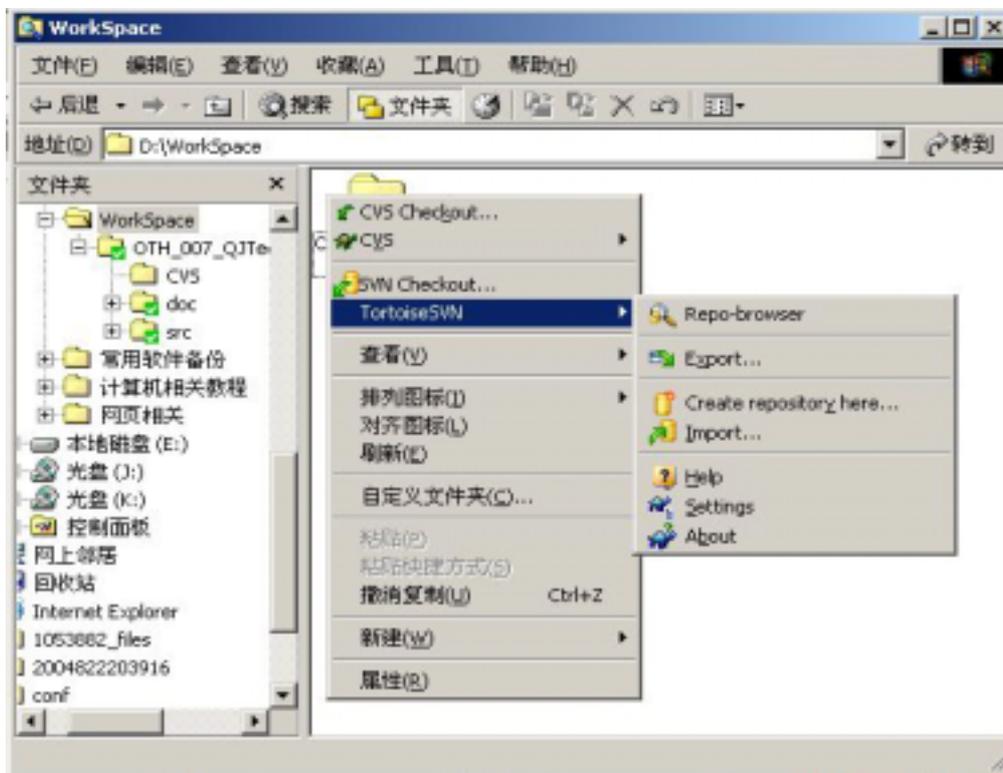


图七、导入项目后 Revision 版本变为 1

3.3 TortoiseSVN 操作简介

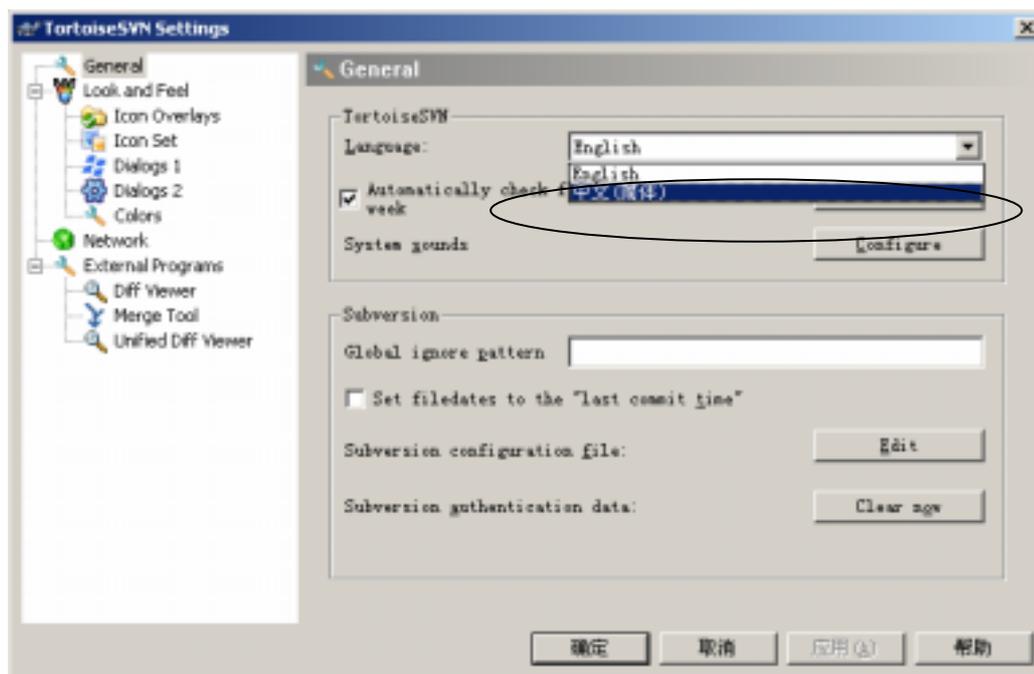
3.3.1 选择语言

TortoiseSVN 安装后，默认使用的是英文，为了使用中文需要执行以下操作，在 windows 资源管理器中右击 TortoiseSVN->Settings.



图八、安装 TortoiseSVN 增强的右键菜单

弹出以下对话框，选择“中文（简体）”后点击应用，即更改默认显示为中文如下图所示：



图九、更改默认语言

3.3.2 添加项目(import)

进入需要导入的文件夹，右击“TortoiseSVN”->“导入”，则可出现如下图所示对话框，在文件库路径中填写刚新建的仓库

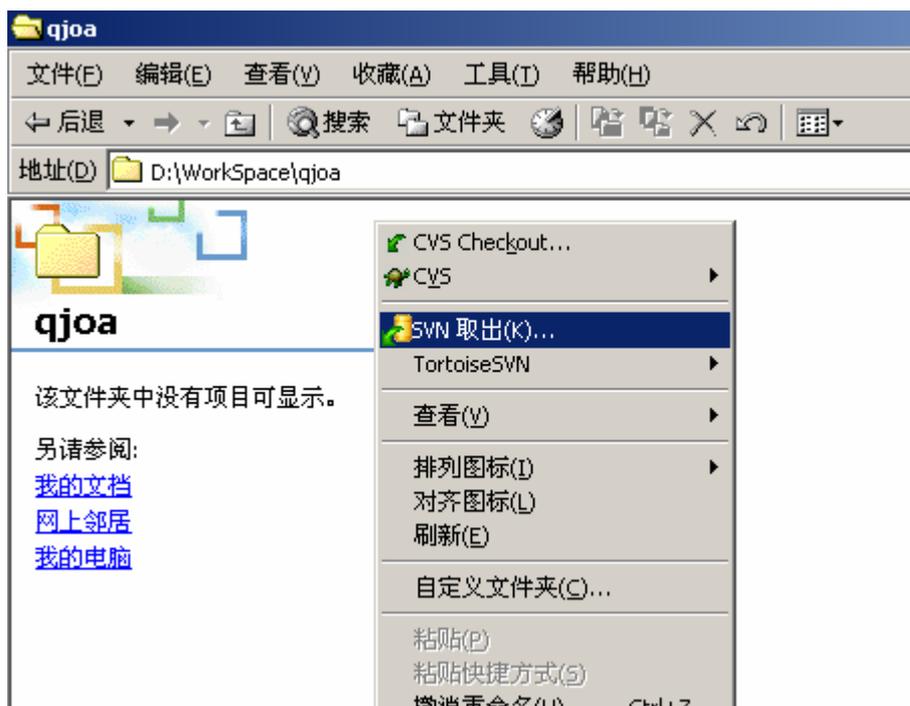
<http://www.ilinux.com.cn/repos/qj oa>，点击确定，即可完成项目的导入



图十、导入项目

3.3.3 检出项目(checkout)

在工作目录新建一文件夹 QJOA（我使用的是 D:/workspace/qj oa），在有多个项目时，使用工作目录绝对是一个不错的习惯^_^，进入 qj oa 目录，右击，即出现如下图所示菜单



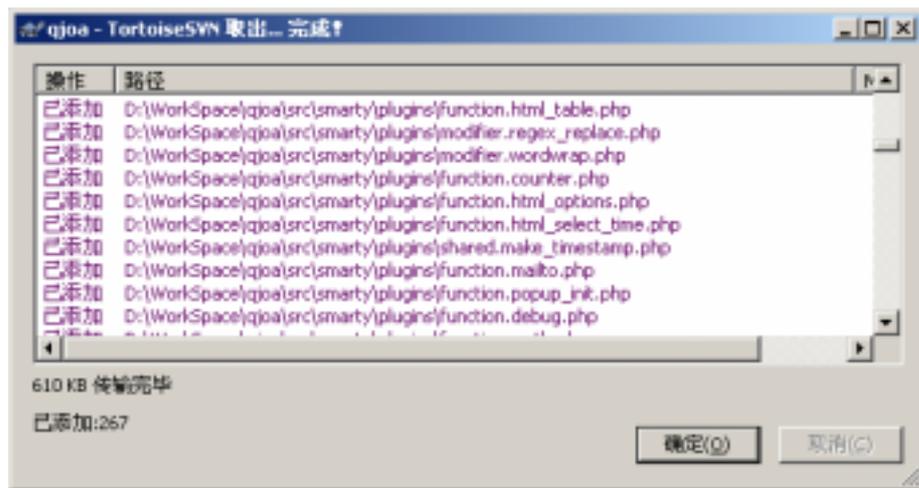
图十一、导出项目(step1)

单击“SVN 取出”，则出现如下图所示的画面，



图十二、导出项目(step2)

填写适当的信息，单击确定，则 TortoiseSVN 将开始检出项目，如下图所示



图十二、导出项目(step3)

3.3.4 列出项目

资源管理器中右击 TortoiseSVN->文件库浏览器:



图十三、文件浏览器

在 URL 栏中填入仓库路径，就可以浏览仓库内容，如下图所示



图十四、仓库文件浏览

3.3.5 状态查询

3.3.6 添加文件或目录

选中要添加的文件夹或文件右击 “TortoiseSVN” -> “添加”

3.3.7 删除文件或目录

选中要添加的文件夹或文件右击 “TortoiseSVN” -> “删除”

3.3.8 提交修改

在工作目录，右击 -> “SVN 提交”

3.3.9 更新

在工作目录，右击 -> “SVN 更新”

注意：添加或删除后需要进行提交操作。更多的有关 Subversion 的详细用法，请查阅 [Subversion 官方文档中文版本](#)。

4 相关资源

- 4.1 、 [Subversion 中文站](http://www.subversion.org.cn/) (<http://www.subversion.org.cn/>);
- 4.2 、 [Subversion 官方站](http://subversion.tigris.org/) (<http://subversion.tigris.org/>);
- 4.3 、 [Subversion 官方文档中文版本](http://www.easyea.com/doc/CH/svnbook/) (<http://www.easyea.com/doc/CH/svnbook/>);

5 参考

- 5.1 、 [Installing Subversion](http://svn.collab.net/repos/svn/trunk/INSTALL) (<http://svn.collab.net/repos/svn/trunk/INSTALL>);
- 5.2 、 [CVS to SVN Crossover Guide](http://svn.collab.net/repos/svn/trunk/doc/user/cvs-crossover-guide.html)
(<http://svn.collab.net/repos/svn/trunk/doc/user/cvs-crossover-guide.html>);
- 5.3 、 [apache 文档 2.0](http://httpd.apache.org/docs/2.0/) (<http://httpd.apache.org/docs/2.0/>);