

Jdeveloper 下 Svn 的使用

--author 周晨
--date 2011/11/09
--company hand

Environment

Tools	Version	Source
Jdeveloper	Studio Edition Version 11.1.1.5.0	http://www.oracle.com/technetwork/cn/developer-tools/jdev/downloads/index.html
Svn server	Versioning support for subversion 1.1.1.5.37.60.13	Studio Edition Version 11.1.1.5.0 自身已经集成
Svn client	1.7.1 64bit	http://tortoissvn.net/downloads.html

Introduce

Subversion 是一个开放源代码的版本控制系统。它功能强大，可以替代常用的并发版本控制系统 (CVS)。它对 CVS 进行了多项改进，如版本控制日志和元数据，并且其体系结构可实现更简单、更灵活的网络访问。Oracle 为 JDeveloper IDE 提供了一个扩展功能，该扩展功能可将与 Subversion 交互所涉及的大部分手动工作自动化。有关 Subversion 的详细介绍，请查阅 <http://svnbook.red-bean.com/> 上的“Version Control with Subversion”（用 Subversion 进行版本控制）。

JDeveloper Subversion 集成的重要优势

Oracle 在 JDeveloper IDE 内提供了一个 Subversion 集成，可自动化在与 Subversion 交互时涉及到的大量手动工作。

- 无需访问应用程序的根目录就可以自动提交/更新工作副本。
- 基于 IDE 的全面操作
- 审核和操作暂停更改
- 分支支持

- 导入 JDeveloper 应用程序
- 审核历史更改
- 合并和解决冲突
- 重构支持
- 查看和编辑 Subversion 属性

安装 visual svn server 并设置 repository

您需要首先设置信息库并提供对它的访问，然后才能使用 Subversion 存储应用程序修订。Subversion 提供三种主要的信息库访问方式：

- 通过本地文件系统（文件：协议）
- 通过专用的 svn：网络协议
- 通过配置有相应模块（`mod_dav` 和 `mod_dav_svn`）的 Apache HTTP 服务器

尽管 Oracle JDeveloper 提供对创建本地信息库并（通过文件：协议）访问它们的支持，但此方法只有在您是项目的唯一开发人员时才很有用。最常用的访问 Subversion 信息库的方法是通过 HTTP/HTTPS 协议（使用 Apache）。要创建信息库并通过 Apache 提供访问，您必须安装并配置一些软件 或 连接到一个已有 Subversion 信息库。对于我的一些 Oracle ADF 项目，我使用了提供 Web 界面来创建和管理信息库的商业 Subversion 托管服务。这可能是一个非常方便的选择，因为托管公司将会为您管理备份并且还会经常提供与问题跟踪软件的集成。

如果您希望创建您自己的信息库，您将需要安装并配置 Subversion 软件，以及 Apache 和所需模块。许多 Linux 发布版本带有已配置的 Subversion 和 Apache。针对 Microsoft Windows，一个名叫 VisualSVN 的公司创建了一个用于安装 Subversion 软件的 Windows 安装程序、一个预配置的 Apache HTTP 服务器，以及一个易于使用的管理控制台。当然，您可以单独安装 Subversion 和 Apache HTTP 服务器并使用命令行信息库管理工具来创建信息库，但使用 VisualSVN Server (www.visualsvn.com/server) 是一种更快捷的方法。

VisualSVN Server 的安装非常简单：只需下载安装程序并运行。我提供了该安装中重要部分的屏幕截图，包括指定信息库的存储路径，Apache HTTP 服务器的端口以及 Subversion 服务器应该在哪儿获取其用户和组的信息。我指定了一些适用于独立安装的选项，使用 HTTP（不是 HTTPS），并且选择管理我自己的用户和组而不是使用 Windows 用户和组：



(图一)



(图二)

安装很快完成，并且会创建一个名为 Visual SVN Server 的 Windows 服务。在安装结束时，您将会看到启动 VisualSVN Server Manager 的选项：



(图三)

在 VisualSVN Server Manager 中，您将会看到 Repositories、Users 和 Groups 节点。在本文中，只需右键单击 Repositories 节点并在上下文菜单中选择 Create New Repository... 即可创建信息库：



(图四)

创建 user



(图五)



(图六)

Add your applications into repository

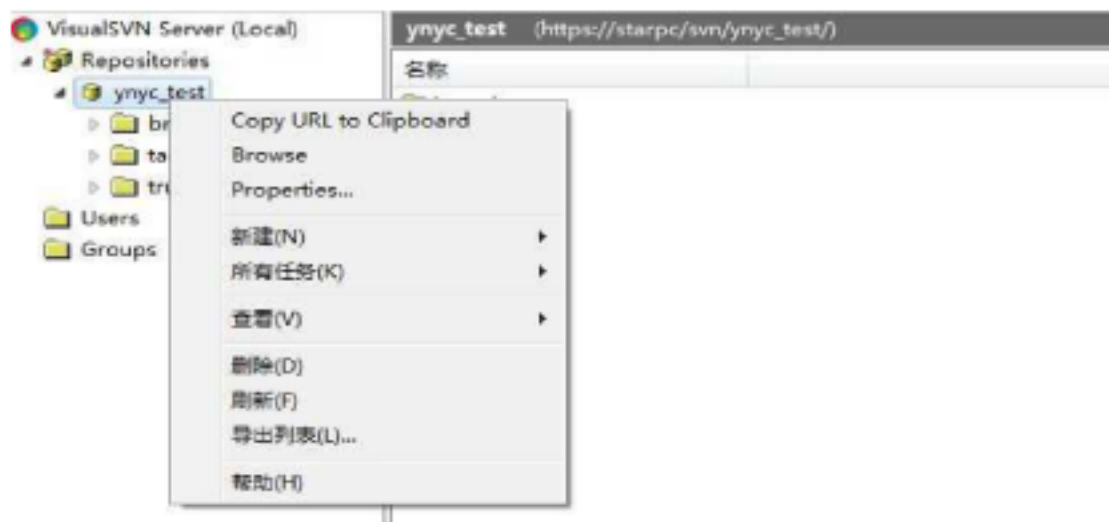
在这一部分，将在 `jdeveloper` 中使用 `svn`：

这里包括几步：

- 1: 建立与 `svn repository` 的链接
- 2: 建立链接后，进行 `repository` 的初始化
- 3: 初始化后，将旧的文件备份，然后将已经上传到 `repository` 的文件 `check out`（如果不出错，这一步会由 `svn` 自动完成，如果失败，就需要手动进行。）

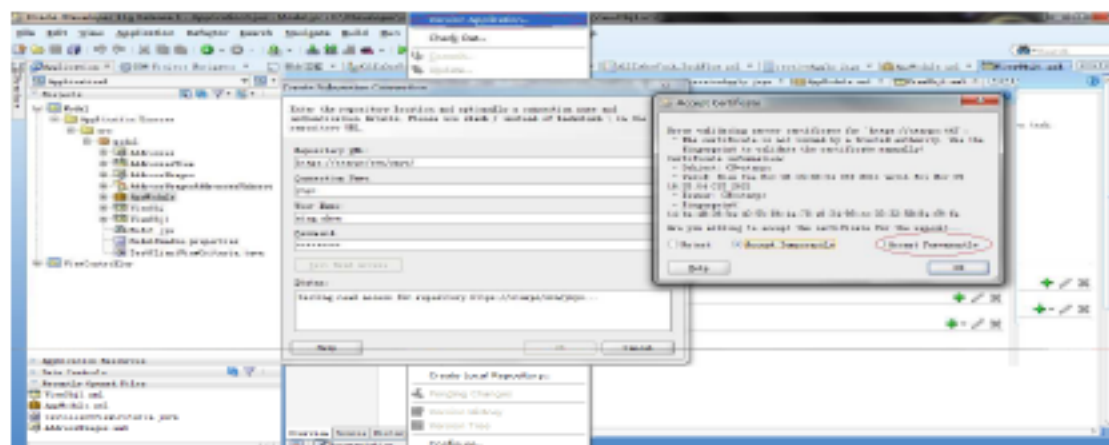


(图七)

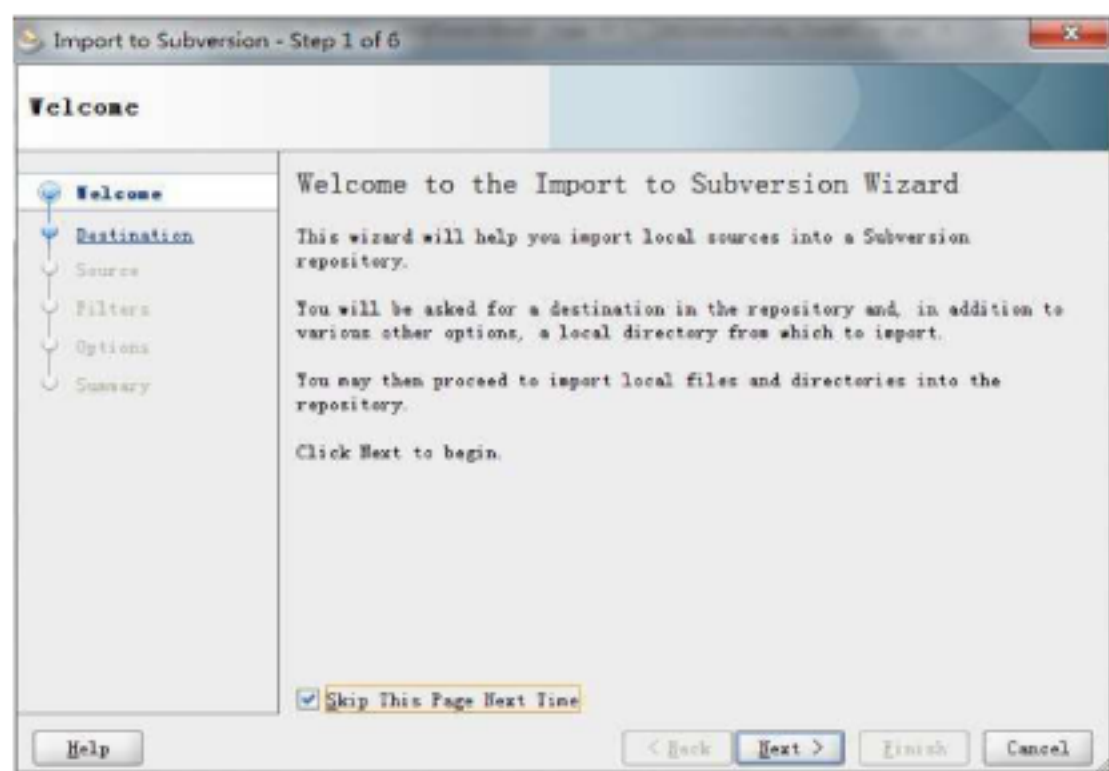


(图八)

图九中 url 可有图八的方法获得

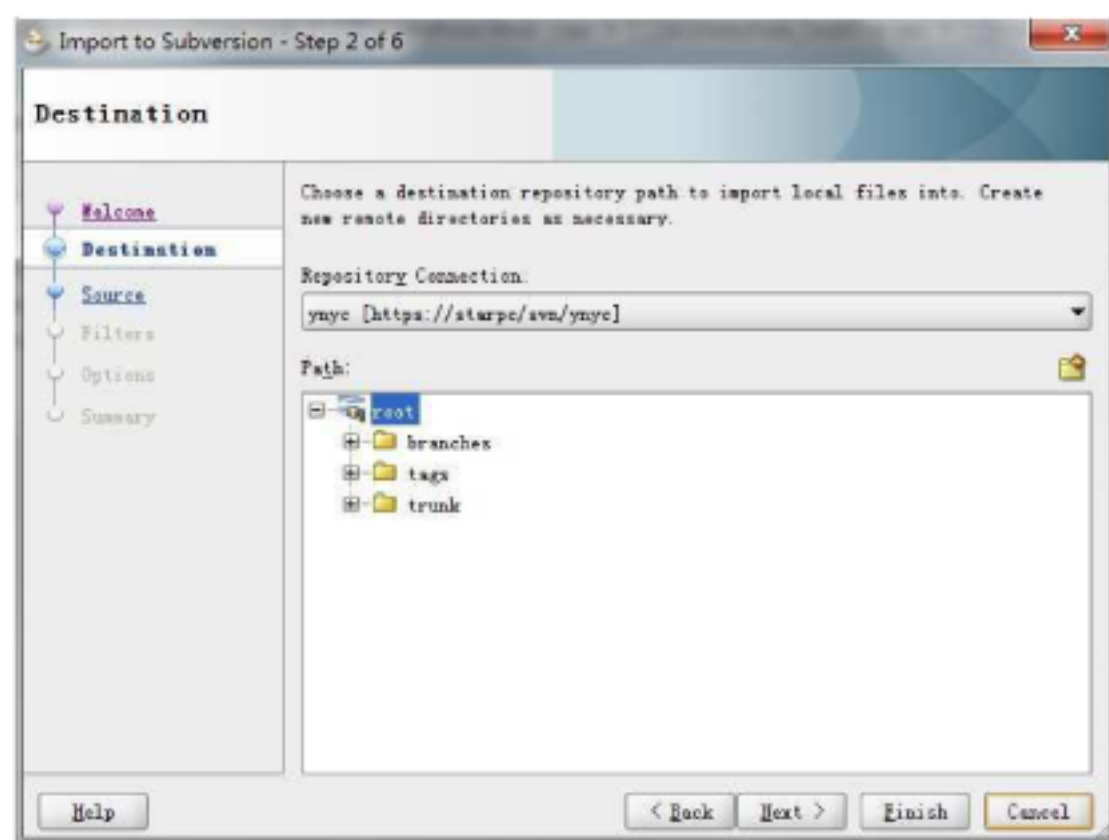


(图九)



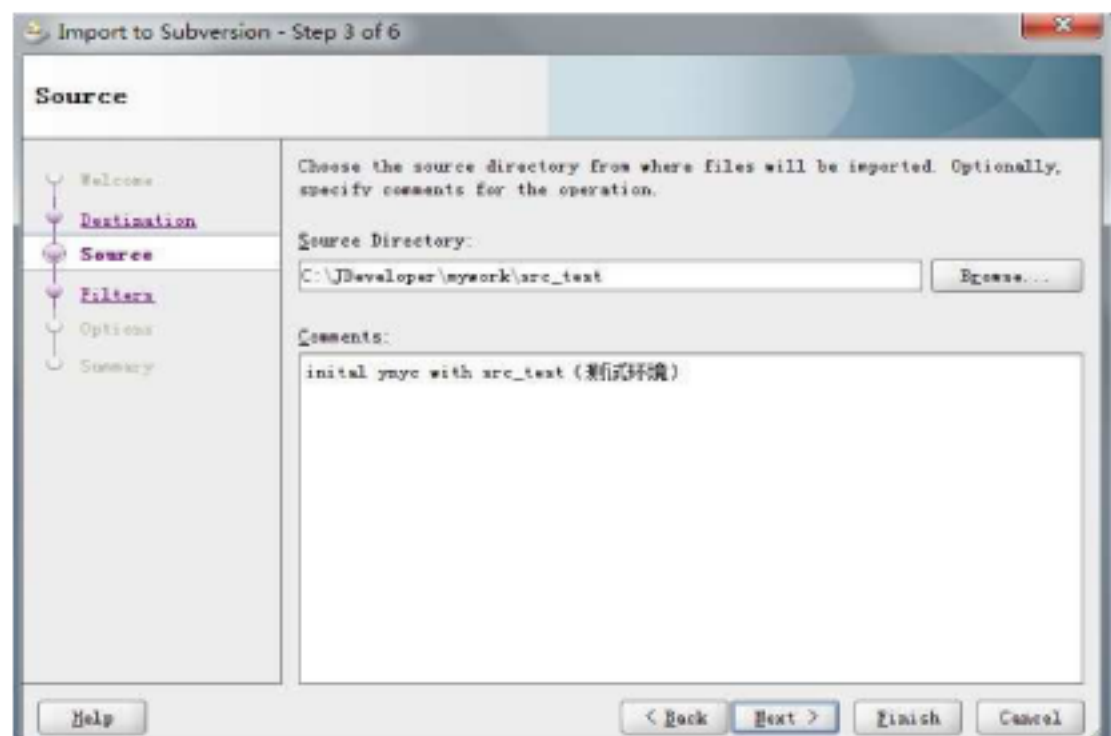
(图十)

选择一个 repository 的链接，并选择一个目标文件夹，这里 trunk 是开发的主目录。



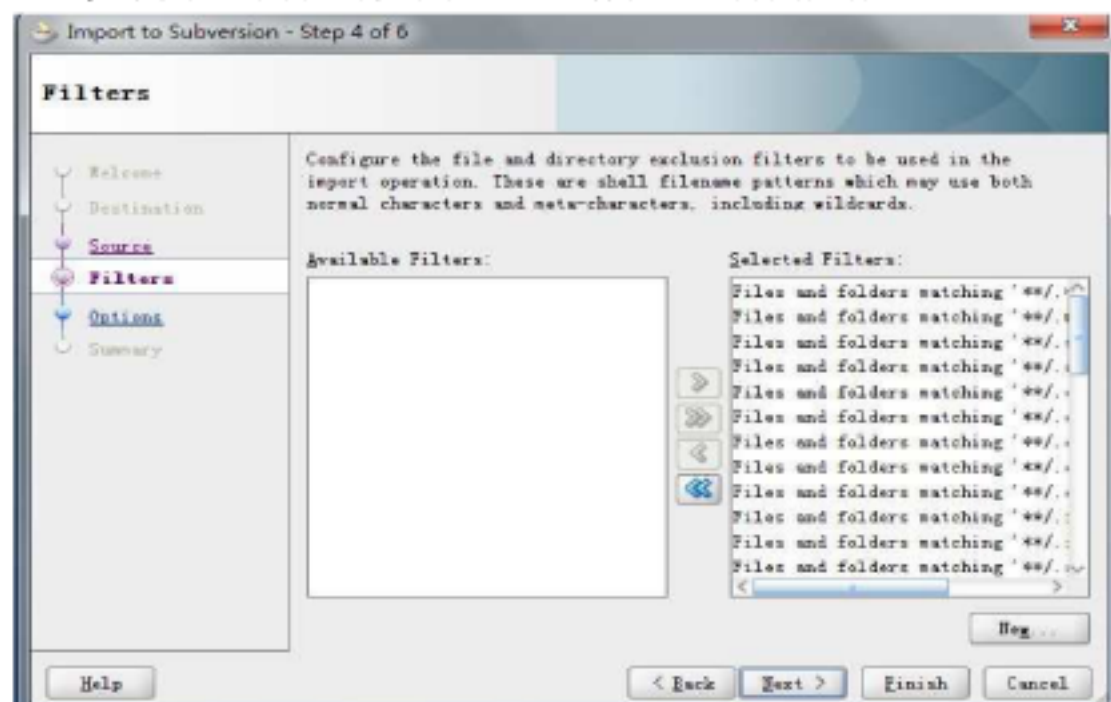
(图十一)

选择需要导入的文件路径



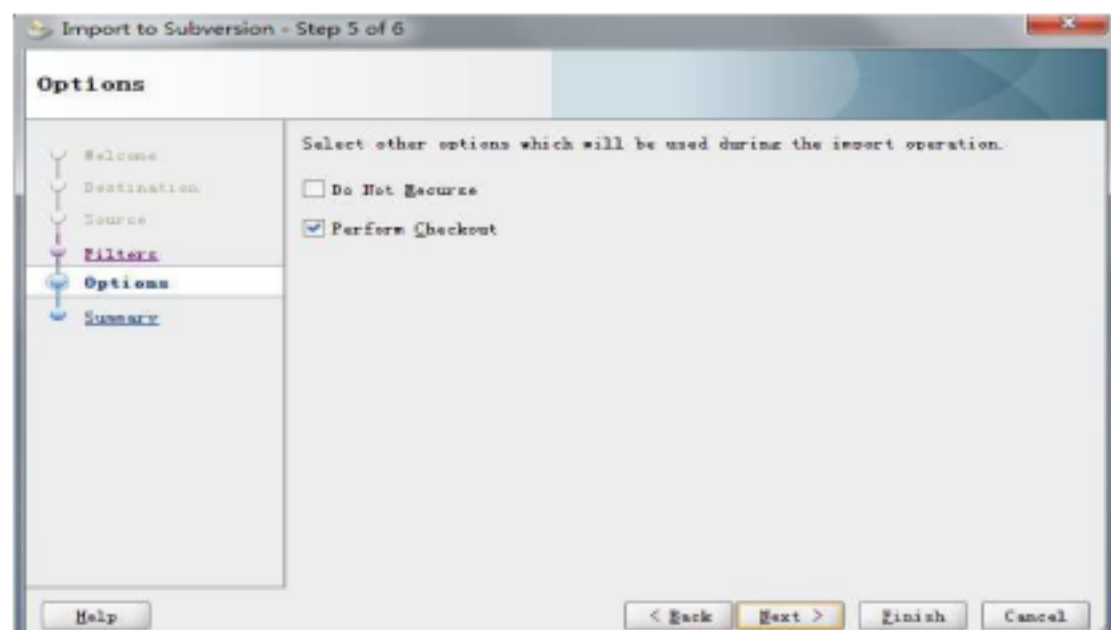
(图十一)

导入的下一步是告诉 Subversion 我们希望它忽略的文件和目录。我们通常不希望生成的文件（例如类文件）、临时文件等存储在 Subversion 信息库中，所以我们需要告诉 Subversion 忽略它们。关于机制我们不需要了解太多，只需知道 Subversion 使用一个特殊的属性来识别应该忽略的对象就够了。Oracle JDeveloper 提供了一整套默认忽略模式，然而您也可以使用 Filters 对话框自己添加：



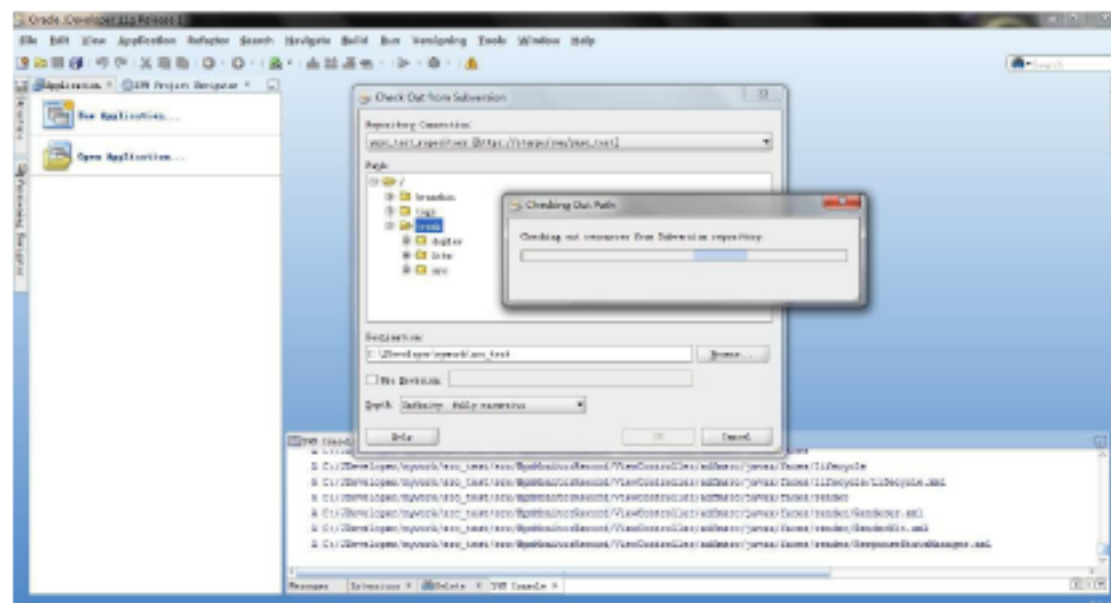
(图十二)

当您您的应用程序导入 Subversion 时，您本地磁盘上的原始文件仍然是一组本地文件；本地副本根本不具有到信息库的连接（它仅仅是一个本地副本而不是一个 Subversion“工作副本”）。通常导入您应用程序后的下一步是删除您的本地副本并签出信息库中的工作副本。Oracle JDeveloper 在下一个对话框 Options 中会向您提供一个快捷选项：

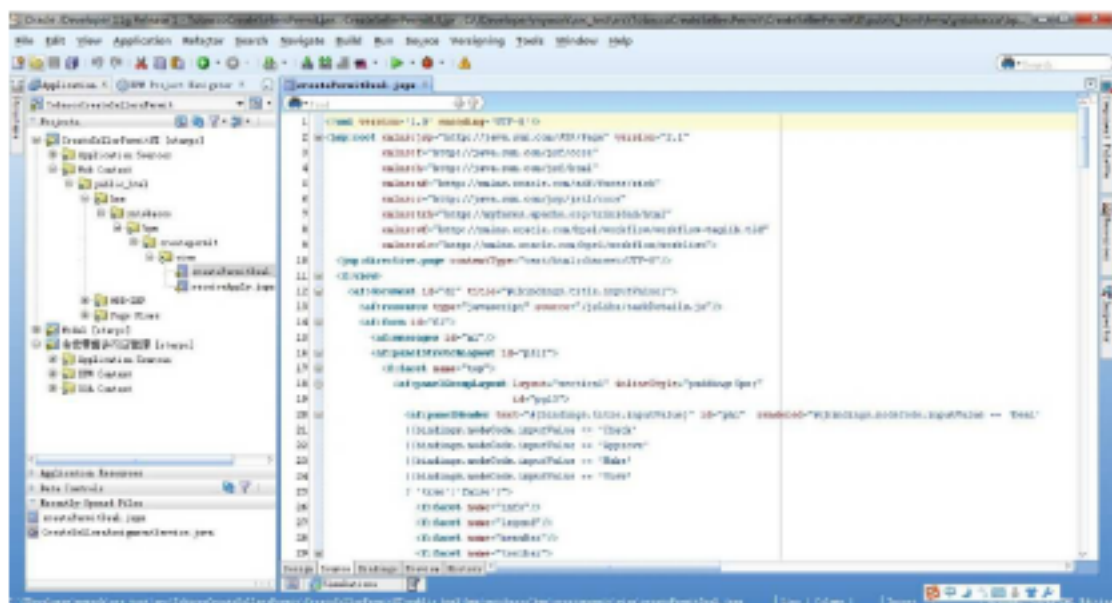


(图十三)

点击 finish 等待一段时间，完成后，即可看到：



(图十四)



(图十五)

安装 Subversion 客户端软件(非必须)

除了 JDeveloper Subversion VCS 扩展以外，您不必安装任何 Subversion 软件，但以下情况除外：

- 您希望使用 JDeveloper Subversion VCS 扩展创建本地 Subversion 信息库。
- 您希望使用 Java 绑定（帮助类库）而不是该扩展提供的 JavaSVN。
- 您希望通过代理服务器连接到 Subversion 信息库（请参阅[通过代理服务器连接到 Subversion 信息库](#)）。

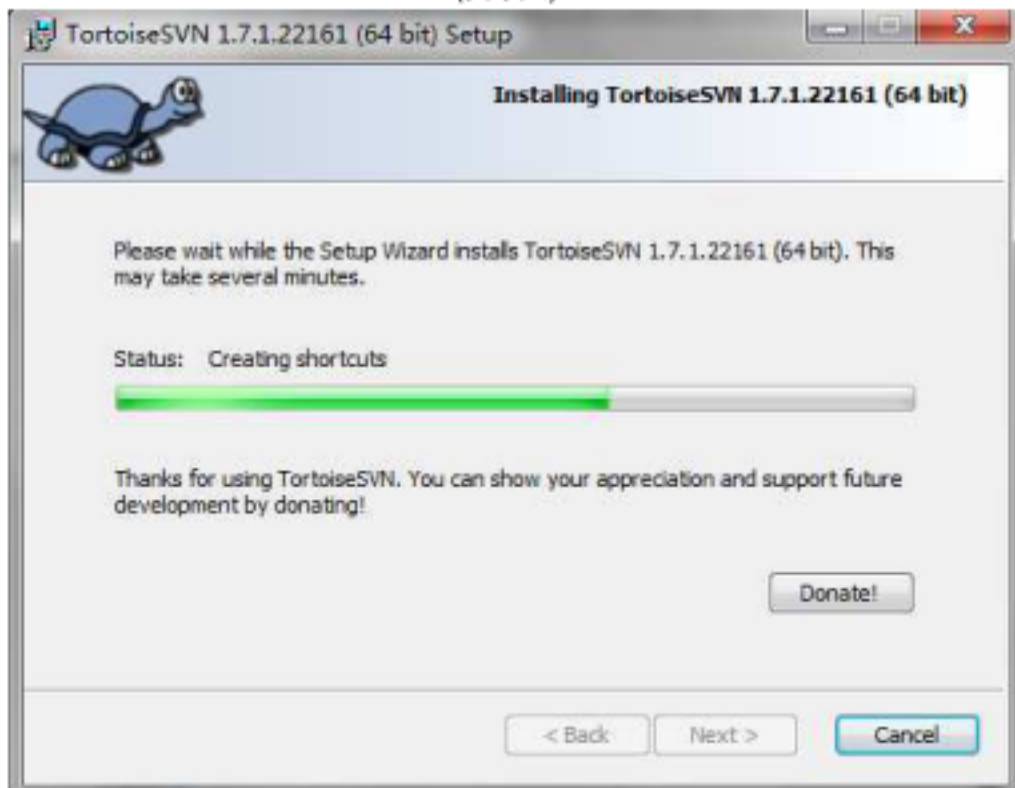
对于 Microsoft Windows，请从 <http://subversion.tigris.org> 下载 Subversion 安装程序 `svn-1.3.2-setup.exe`，然后，运行该安装程序，并将 Subversion 客户端放在一个方便的位置，如 `c:\subversion`。要检查安装，请打开一个命令提示符，然后键入 `svn help`。如果您没有看到子命令列表，请检查系统路径是否包含了 Subversion 的安装位置(例如，`echo %PATH%` 应该包含 `c:\subversion\bin`)。

客户端 `svn` 一般使用：

<http://tortoisesevn.net/downloads.html>

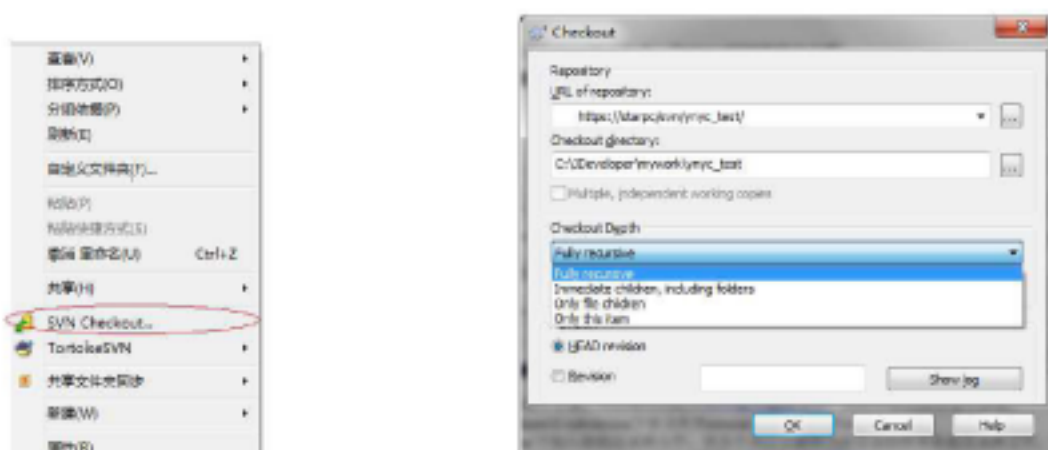


(图十六)



(图十七)

默认检出最新版本，可以指定检出特定版本。TortoiseSVN操作如下图：



如果只需要读写项目的某个子集，可以细化 URL 到你所感兴趣的节点，同时对检出深度进行控制。例如，https://starpc/svn/ynyc_test/下有文件夹 tutorial, binary, 文件 clearSVN.bat, 此时你打算在 subversion/下加入你的目录和文件，并且不关心上级和当前目录的所有其他目录和文件。

那么 url 可以填写为 https://starpc/svn/ynyc_test/，并且选择检出深度为“仅此项”；(--depth=empty)

如果有需要读写当前目录已有的文件，选择检出深度为“仅文件子节点”；(--depth=files)

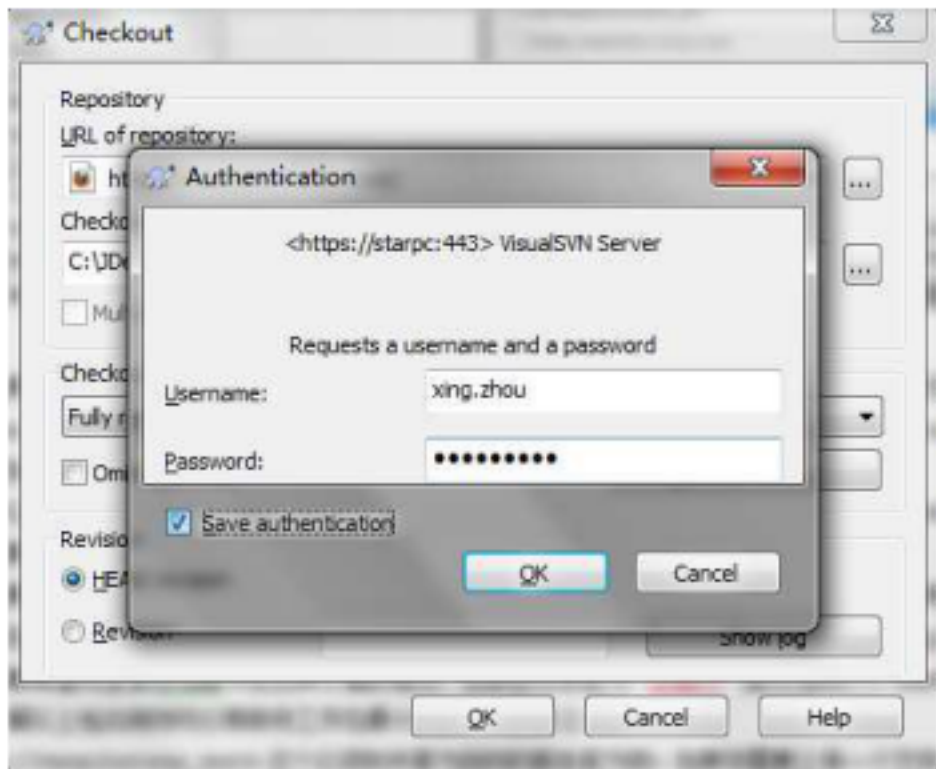
如果有需要了解当前目录已包含的子目录的名字但不需要读写其内容(比如为了避免命名冲突需要了解当前已有的文件夹)，选择检出深度为“直接子节点，包含文件夹”；(--depth=immediates)

如果希望完全掌控当前节点及其子集的情况，选择检出深度为“全递归”(默认选项)。(--depth=infinity)

了解以上检出规则可以帮助你工作在最小化的工作副本上。

以 https://starpc/svn/ynyc_test/e 这个以资料共享为目的的版本库为例。如果你需要上传一个文件夹及里面的资料，选定一个合适的目录节点，仅检出该项，然后在你自己的工作副本里组织目录结构并提交。

(图十八)



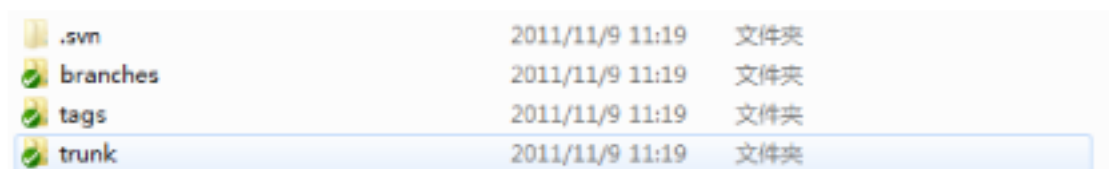
输入用户名密码

(图十九)



开始 checkout

(图二十)



此时，checkout 的工程图标上已经有了绿色的勾，这正是 svn client 的标志，现在就对工程进行操作了。

(图二十一)

JDeveloper TortoiseSVN 版本问题

Developer 10.1.3.4 只能与 TortoiseSVN 1.4 版本的客户端配合使用，不能与更高版本的配合。

<http://forums.oracle.com/forums/thread.jspa?threadID=841377>

* if you are using JDeveloper 10g, use SVN 1.3.x or 1.4.x tools if you want to work with them outside of JDev. Do not touch the working copy with the 1.5.x tools.

* if you are using the original JDeveloper 10g, use SVN 1.4.x tools if you want to work with them outside of JDev. Do not touch the working copy with the 1.5.x tools.

* if you are using JDeveloper 11g update 1, use SVN 1.5.x tools if you want to work with them outside of JDev. Earlier versions of the SVN tools will not be able to deal with the working copy.

如果不小心升级到了高版本的 SVN，JDEV 就不能识别 SVN 库了。下载 `change-svn-wc-format.py`，配合 Python2.6 降级 SVN。

<http://sducxh.javaeye.com/blog/589136>

重新启动 JDEV 后可以看到已经恢复正常了，jdev 和小乌龟可以共管项目的 SVN 库了。

通过代理服务器连接 Subversion 信息库

如果您希望通过代理服务器连接 Subversion 信息库，则必须先安装单独的 Subversion 客户端软件。请参阅[安装 Subversion 客户端软件](#)。

一旦您安装了 Subversion 客户端软件，在 Windows Application Data 目录下就会有一个 Subversion 子目录。要查找 Application Data 目录，请在 `c:/` 提示符下键入 `cd %APPDATA%`。然后，打开 Subversion 子目录。（在 Linux 上，等价的子目录将位于 `~/subversion`，其中 `~` 是主目录。）

在 Subversion 中，子目录将是一个名为 `servers` 的文件。使用文本编辑器打开该文件，然后找到 `[global]` 区段。从 `http-proxy-host` 行移除注释标记（`#`），然后用您使用的代理服务器的详细信息覆盖占位符代理信息。从 `http-proxy-port` 行移除注释标记（`#`），然后用代理服务器的端口号覆盖占位符端口信息。如果您希望某些 URL 不使用代理服务器，请从 `http-proxy-exceptions` 行移除注释标记（`#`），然后用这些 URL 覆盖占位符 URL。

用您使用的任何其他代理服务器的详细信息添加额外的 `http-proxy-host` 和 `http-proxy-port` 行。

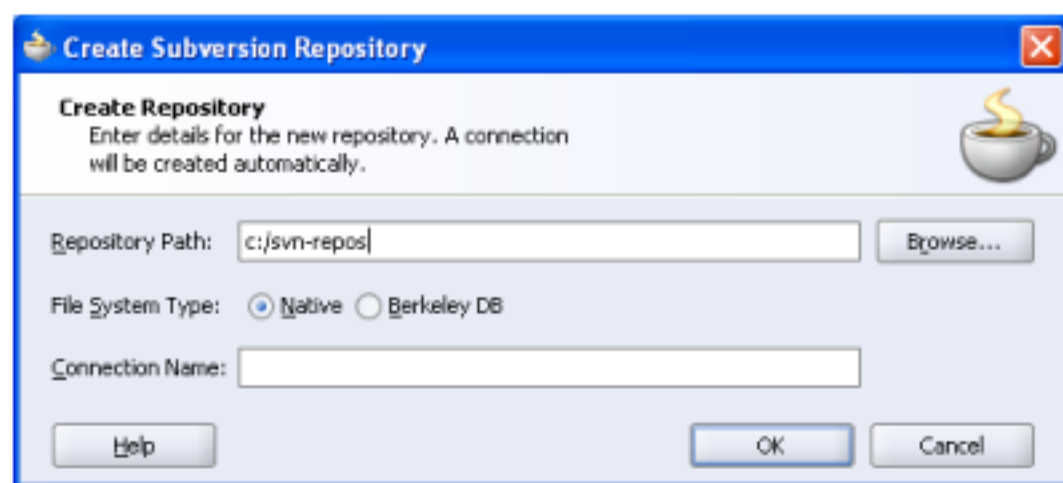
重要的是，代理服务器支持 Subversion 使用的所有 http 方法。默认情况下，某些代理服务器不支持以下方法：PROPFIND、REPORT、MERGE、MREACTIVITY 和 CHECKOUT。如果您在使用代理服务器访问 Subversion 信息库时遇到问题，可以要求服务器的系统管理员更改配置，以支持这些 http 方法。

创建本地 Subversion 信息库

JDeveloper Subversion VCS 扩展包含一个创建本地 Subversion 信息库的功能。由于该扩展目前提供的支持软件的限制，您必须先安装单独的 Subversion 客户端软件，才能使用 JDeveloper 创建本地信息库。安装 Subversion 客户端软件的说明请见[安装 Subversion 客户端软件](#)。

要创建本地 Subversion 信息库，请选择 **Versioning > Create Repository**。

在 **Create Subversion Repository** 对话框中，会为您的新信息库推荐一个位置和文件系统类型。为了提供在本指南中使用的示例，我们将信息库路径改写为 `c:/svn-repos`。您可以将文件系统类型保留为默认设置 `Native`，不必提供连接名称。



当信息库创建之后，还将自动创建指向它的连接。您可以在 **Subversion Navigator** 中看到这个连接。



如果您使用 JavaSVN，则需要将访问方法协议从 `file:///` 更改为 `http://`、`https://`、`svn://` 或 `svn+ssh://`。这些协议将在 Subversion 文档中描述。

提示：对于不需要着重考虑安全性的小组或单个开发人员来说，`svn://` 协议特别有用，并且易于设置。在命令提示符下输入：

```
svnserve -d -r [信息库位置]
```

保持命令窗口打开。当您创建信息库连接时，请使用：

```
svn://[计算机名]/
```

作为信息库位置。

对于本指南中使用的示例，在命令提示符下输入 `svnserve -d -r c:/svn-repos/`，并保持命令窗口打开。在 Subversion Navigator 中，双击连接名称 `file:///c:/svn-repos`。在 Edit Subversion Connection 对话框中，用 `svn://localhost/` 改写 Repository URL。测试连接，然后单击 **OK** 关闭对话框。

参考文档

1: 在 JDeveloper 中使用 Subversion

JDeveloper Subversion VCS 扩展开发人员指南 10.1.3.0，2006 年 7 月
<http://www.oracle.com/technetwork/cn/testcontent/subversion-086916-zhs.html#installdevextn>

2: Oracle ADF 开发必读：第 1 部分

<http://www.oracle.com/technetwork/tutorials/part1-088591-zhs.html>

3: Oracle 的 JDeveloper Subversion 集成

<http://www.collab.net/collabXchange/JDeveloper/>

4: TortoiseSVN 教程

5: TortoiseSVN-1.6-zh_CN.pdf