

第 3 章 学习环境搭建

一个实际的网络管理系统的环境不仅提供对于网络管理的感性认识,而且提供与理论知识的对照,有利于加深对于理论知识的理解,锻炼动手实践的能力和解决实际问题的能力。选用 Windows[®] 作为网络管理的操作系统平台,选用基于 Java 语言实现的开源软件包 SNMP4J 作为 SNMP 的参考实现,使得只要具有一定 Java 语言编程基础的学生就能够了解到 SNMP 实现的细节。完成本书的学习以后,不但掌握了网络管理方面的理论知识,而且可以了解 SNMP4J 的实现原理和编程方法。

本章主要介绍 Windows[®] 平台提供的网络管理功能、SNMP4J 简介和建立 SNMP4J 的 Java 工程 3 个部分。

3.1 Windows[®] 平台提供的网络管理功能

为提供基于 SNMP 的 Windows[®] 域服务器、Web 服务器、数据库服务器以及电子邮件服务器的管理,Windows[®] 在推出 TCP/IP-32 For Windows[®] 协议族时包含了一个 SNMP 服务软件,可以安装在 Windows[®] 服务器或者工作站上,接收或者发送 SNMP 请求、响应或者通知。SNMP 在 Windows[®] 中以系统服务组件的形式存在,可以使用“服务”管理工具来进行控制。

SNMP 服务在 Windows[®] 的默认安装时并不会自动安装,需要手动安装。以 Windows XP 系统为例,SNMP 服务的安装有以下几个步骤。

(1) 右击“网上邻居”图标,在弹出的快捷菜单中选择“属性”选项,弹出“网络连接”窗口,如图 3-1 所示。



图 3-1 “网络连接”窗口

(2) 选择“高级”→“可选网络组件”命令,弹出“Windows® 可选的网络组件向导”对话框,如图 3-2 所示。在窗口中选中“管理和监视工具”复选框。

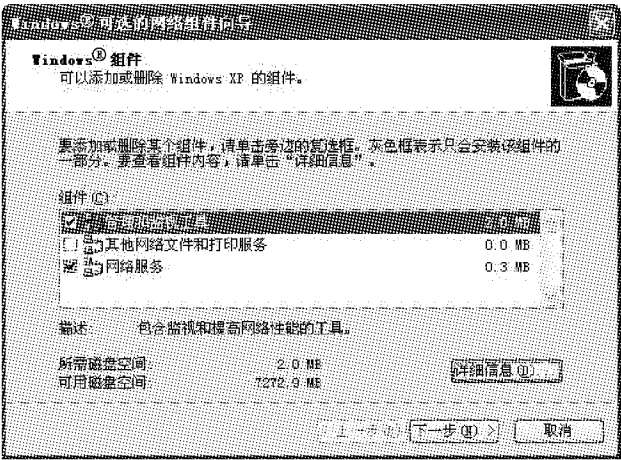


图 3-2 “Windows® 可选的网络组件向导”对话框

(3) 单击对话框中的“详细信息”按钮,弹出“管理和监视工具”对话框,其中包含“WMI SNMP 提供程序”和“简单网络管理协议”两个复选框,如图 3-3 所示。

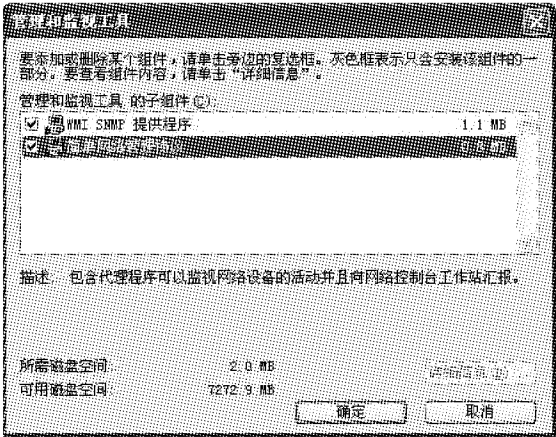


图 3-3 “管理和监视工具”对话框

(4) 将 Windows® XP 安装盘插入光驱中,单击“管理和监视工具”对话框中的“确定”按钮,然后单击“Windows® 可选的网络组件向导”窗口中的“下一步”按钮,开始安装 SNMP 服务。

(5) SNMP 服务安装完毕以后,可以选择“控制面板”→“管理工具”→“服务”命令,打开系统服务窗口,查看已经安装的 SNMP 服务,发现相关的两个服务 SNMP Service 和 SNMP Trap Service 都已经启动,如图 3-4 所示。

(6) 可以右击“SNMP Service”选项,在弹出的快捷菜单中选择“属性”命令,在弹出的“SNMP Service 的属性”对话框中设置 SNMP 服务的属性或控制 SNMP 服务的启动,如图 3-5 所示。



图 3-4 系统服务窗口

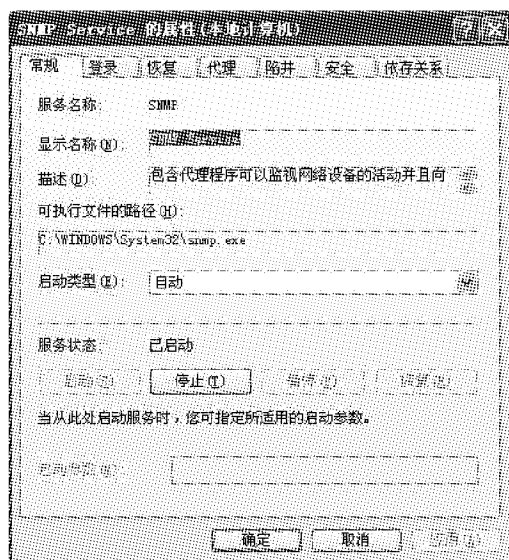


图 3-5 “SNMP Service 的属性”对话框

在安装完 SNMP 服务以后,安装 SNMP 服务的主机就可以作为被管设备对外提供基于 SNMP 的网络管理功能,网络内的网络管理者可以利用 SNMP 管理该主机。

除提供 SNMP 服务外,还提供了应用编程接口 SNMP API,主要由扩展代理编程接口、管理编程接口、实用程序编程接口和 WinSNMP 编程接口 4 部分组成。

① 扩展代理编程接口: 定义了 SNMP 服务和第三方 SNMP 扩展代理 DLL 之间的接口,用来解析由引入的 SNMP PDU 指定的变量绑定;

② 管理编程接口: 定义了第三方 SNMP 管理端应用程序与管理函数动态连接库之间的接口,第三方管理应用程序可以调用管理 API 实现发送 SNMP 请求报文、接收响应等管理操作;

③ 实用程序编程接口：简化 SNMP 数据结构的操作，提供在 SNMP 应用程序开发中有用的函数集；

④ WinSNMP 编程接口：为在 Windows® 平台下开发基于 SNMP 的网络管理程序提供了解决方案，提供了必须遵循的接口规范，规定了过程调用、数据结构和相关语法等。

编程接口所包含的具体函数以及具体函数的调用方法本书不详细介绍，有兴趣的读者请查阅相关资料。

3.2 SNMP4J 简介

SNMP4J 是 Apache 基于 Java 语言的开源项目，为 SNMP 编程提供了很好的框架，支持 SNMPv1、SNMPv2c 和 SNMPv3 版本。SNMP4J 既支持编写客户端程序，也支持编写服务器端程序；既支持命令的产生（网络管理者），也支持命令的响应（网络管理代理）。SNMP4J 的源码包可以到 www.snmp4j.org 下载。

SNMP4J 具有以下特征。

① SNMPv3 支持基于 MD5 和 SHA 的认证和基于 DES、AES 128、AES 192 和 AES 256 的私密性支持；

② 具有可插拔的消息处理模型 (Message Processing Model)，支持 MPv1 (Message Processing version 1)、MPv2c (Message Processing version 2 with community) 和 MPv3 (Message Processing version 3)；

③ 具有可插拔的传输映射 (Transport Mapping)，支持到传输协议 UDP 和 TCP 的映射；

④ 具有可插拔的超时模型 (Timeout Model)；

⑤ 支持同步或者异步的请求；

⑥ 既支持命令产生器，也支持命令响应器；

⑦ 支持基于 Log4J 的日志处理；

⑧ 支持多线程。

SNMP4J 的类能够创建、发送和接收 SNMPv1/v2c/v3 消息（包括消息头和 PDU 载荷）。在 SNMP4J 的包中主要包含 3 组类和接口。

① SNMP 消息和目标 (target) 创建的类；

② SNMP 消息发送 (命令的产生) 的类；

③ SNMP 消息分发 (命令的响应) 的类。

如图 3-6 所示的包图显示了 SNMP4J 核心编程接口的包之间的依赖关系。通常编程人员只需要使用 org.snmp4j 包和 org.snmp4j.smi 包中的类。图中包含以下 6 个包。

① org.snmp4j.smi 包：提供了表示 SMIV1/v2 数据类型的类，也包含一些基本的 ASN.1 的原始数据类型，这些类型能够以 BER (Basic Encoding Rules, 基本编码规则) 中解码或者编码为 BER；

② org.snmp4j.asn1 包：提供了 ASN.1 格式的值和 BER 编码的传输语法之间的映射；

③ org.snmp4j.event 包：SNMP4J 的事件处理包，允许内部或者外部的模块侦听 SNMP4J 内部事件，如处理 SNMP 响应消息；

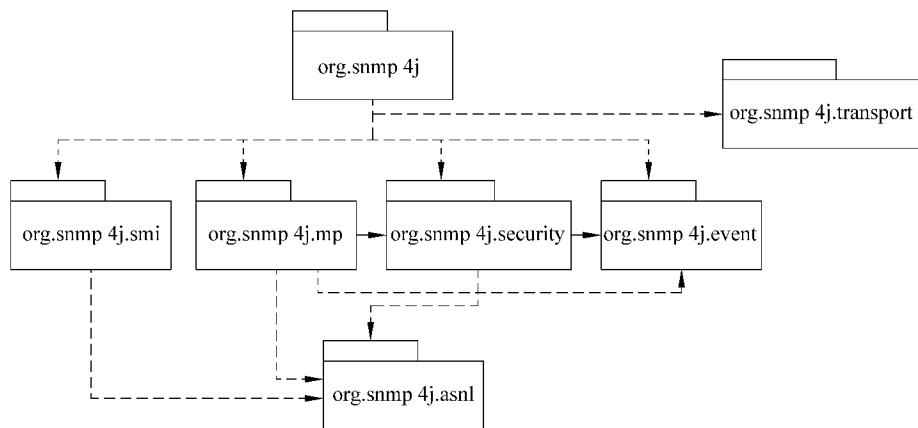


图 3-6 SNMP4J 包图

④ org.snmp4j.mp 包：消息处理包，负责 SNMP 消息的发送和分发；

⑤ org.snmp4j.security 包：实现 SNMP 消息认证和私密性的包；

⑥ org.snmp4j.transport 包：提供 SNMP 传输协议映射的包，使用传输映射指定的传输协议来发送和接收字节消息。

如图 3-7 所示的类图显示了 org.snmp4j 包中最主要的类及其之间的关系。

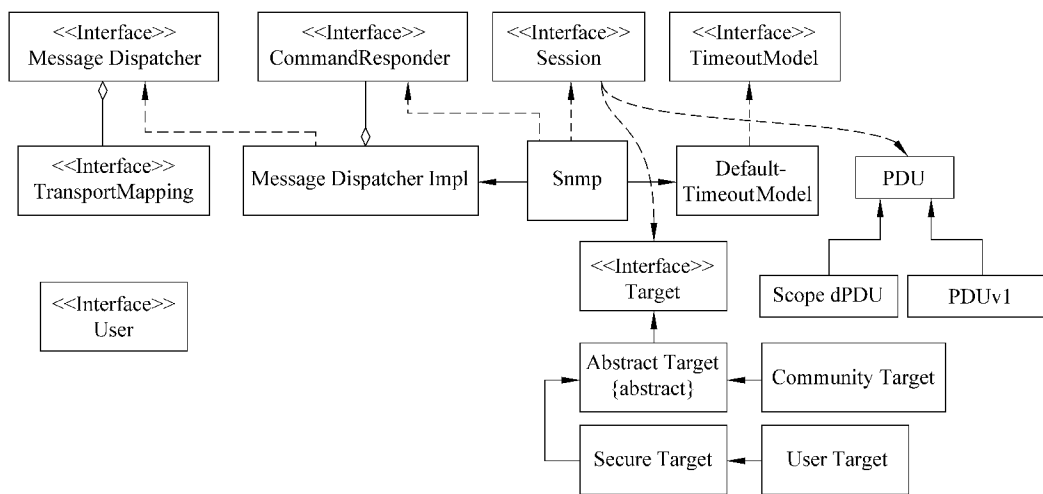


图 3-7 SNMP4J 中主要的类及其之间的关系

通过上面的介绍，可以对 SNMP4J 有一个概貌性的了解，其实现细节在后续章节中还会详细介绍。

3.3 建立 SNMP4J 的 Java 工程

SNMP4J 是基于 Java 语言的 SNMP 实现，可以在 Eclipse 等 Java 集成开发环境中建立一个 SNMP4J 的工程，从而建立开发、测试和跟踪运行的 Java 环境，便于后续章节中 SNMP 协议知识的学习。

在 Eclipse 中建立 SNMP4J 工程有以下几个步骤。

- (1) 安装 Java 环境 JDK,JDK 可以到 <http://java.sun.com/javase/downloads/index.jsp> 下载。
- (2) 下载并解压缩 Eclipse,Eclipse 可以到 www.eclipse.org 下载。
- (3) 双击 Eclipse 图标,运行 Eclipse,选择 File→New→Project 命令,弹出 New Project 对话框,如图 3-8 所示,选择 Java Project 选项,然后单击 Next 按钮,进行下一步操作。

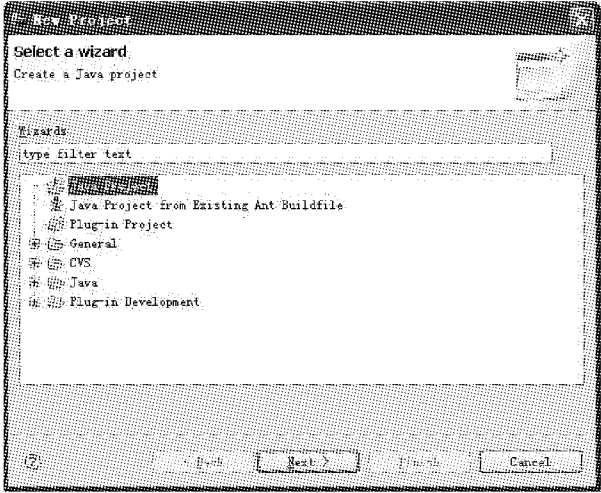


图 3-8 New Project 对话框

- (4) 在转换后的 New Java Project 对话框中,输入新建 Java 工程的名称,如“SNMP”,其他采用默认选项,然后单击 Next 按钮,进行下一步操作,如图 3-9 所示。



图 3-9 New Java Project 对话框 1

(5) 在转换后的 New Java Project 对话框中选择默认设置,然后单击 Finish 按钮,完成新的 Java 工程的创建,如图 3-10 所示。

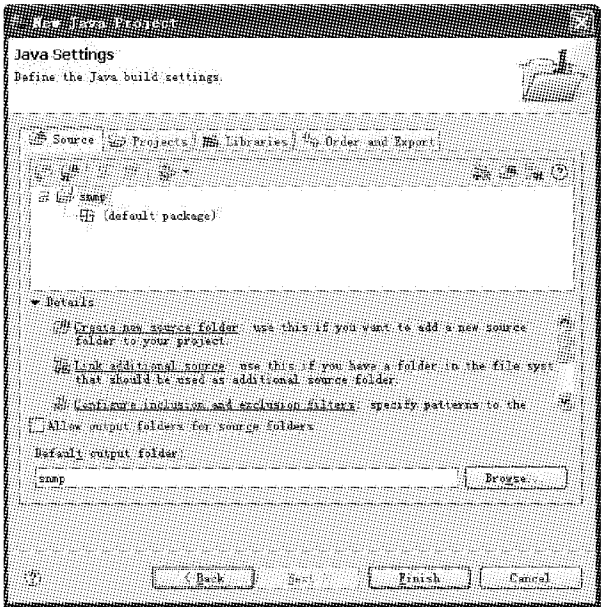


图 3-10 New Java Project 对话框 2

(6) 在新建的 Java 工程中,选择 File→New→Folder 命令,建立一个新的文件夹,用来存放 Eclipse 自动编译后生成的二进制类文件,例如名字为“build”,如图 3-11 所示。然后单击 Finish 按钮,完成文件夹的创建。

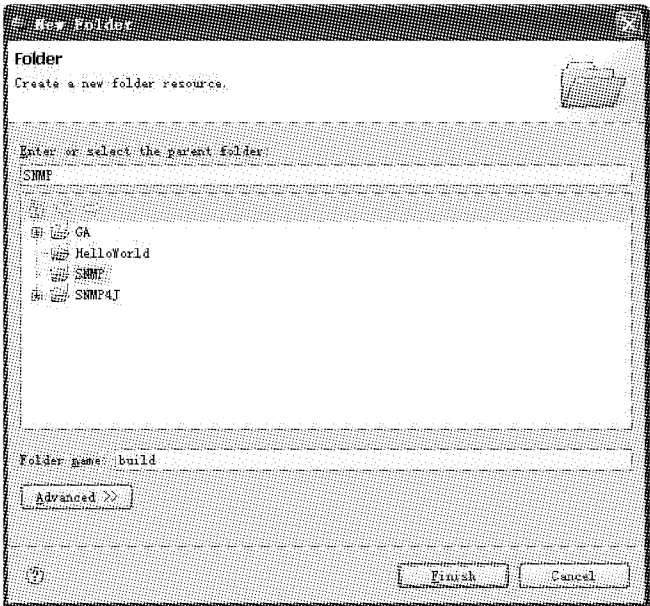


图 3-11 New Folder 对话框

(7) 解压缩 SNMP4J 软件包,将源码目录和库文件(jar 文件)通过资源管理器复制到新建的 Java 工程目录下,如图 3-12 所示。

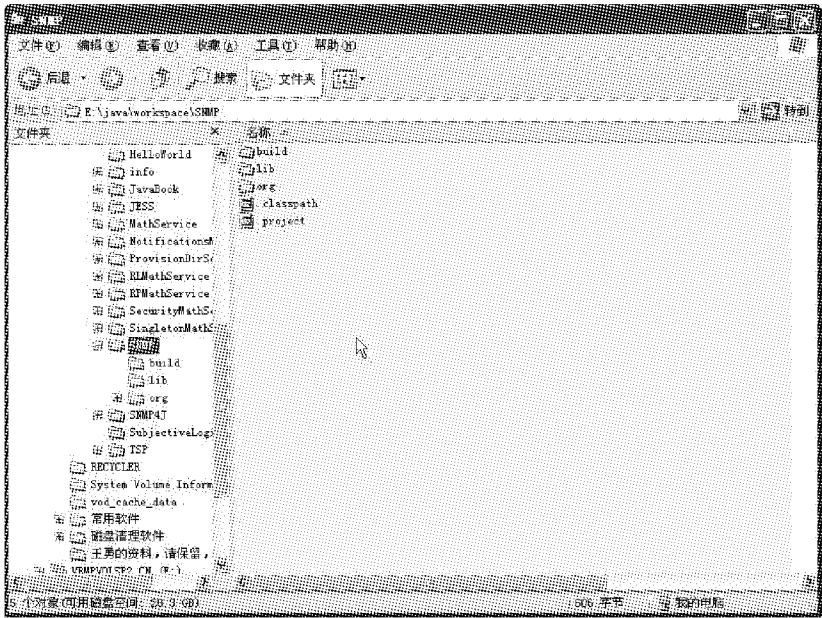


图 3-12 复制源码目录和库文件目录

(8) 选中新建的 Java 工程,选择 Project→Properties 命令,在弹出的 Properties for SNMP 对话框中选择 Java Build Path 选项,在 Source 选项卡中的 Default output folder 文本框中输入“SNMP\build”,注意,“SNMP”为新建 Java 工程的名字,而“build”为新建文件夹的名字,如图 3-13 所示。

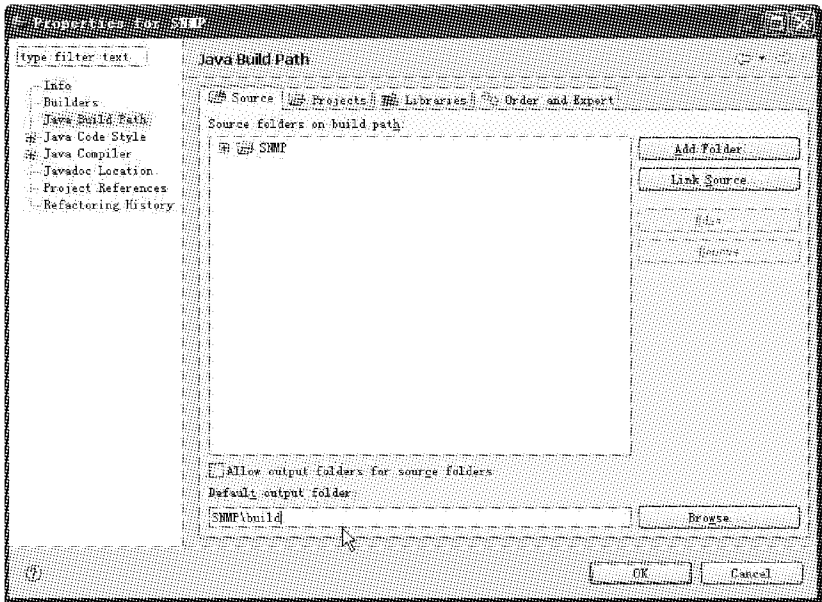


图 3-13 设置输出文件夹

(9) 在 Properties for SNMP 对话框中,打开 Libraries 选项卡,看到需要的库只有 Java 运行环境的基本库,没有 SNMP4J 需要的 log4j. jar,单击 Add Library 按钮,如图 3-14 所示。

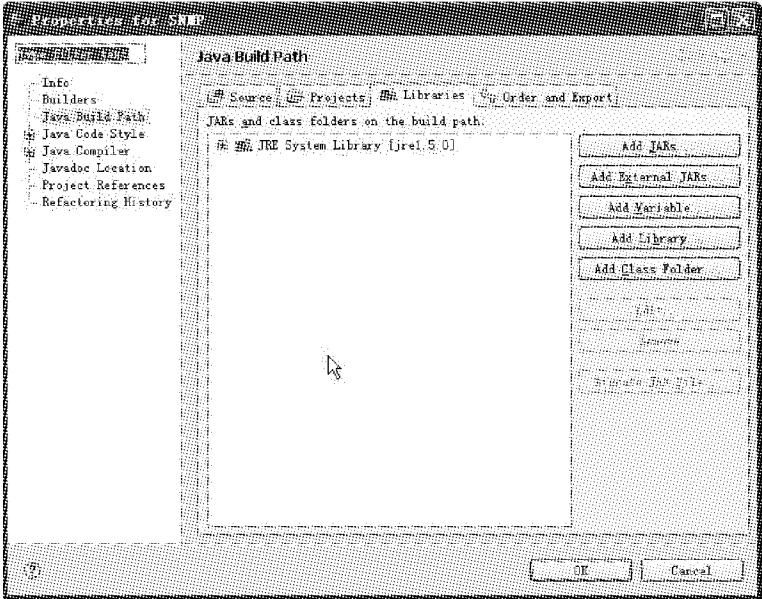


图 3-14 添加库文件

(10) 在弹出的 Add Library 对话框中选择 User Library 选项,然后单击 Next 按钮,进行下一步操作,如图 3-15 所示。

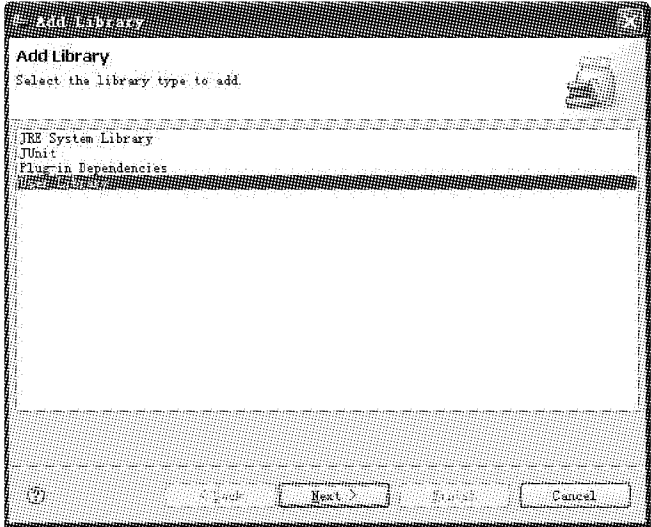


图 3-15 Add Library 对话框

(11) 在转换后的 Add Library 对话框中,单击 User Libraries 按钮,如图 3-16 所示。

(12) 在弹出的对话框中,单击 New 按钮,如图 3-17 所示。

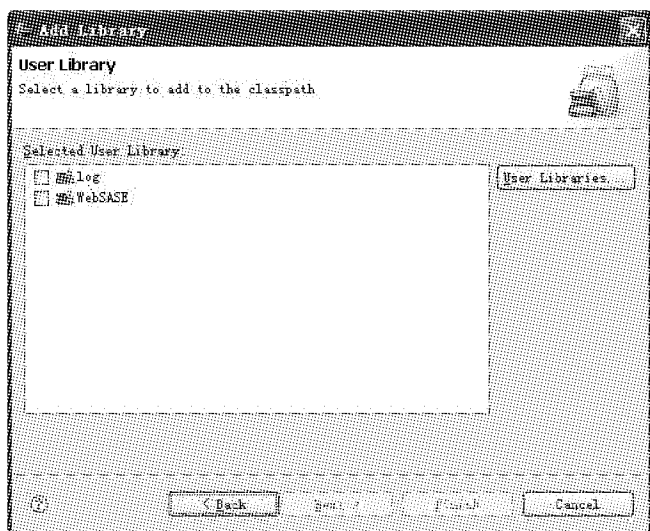


图 3-16 选择用户库对话框

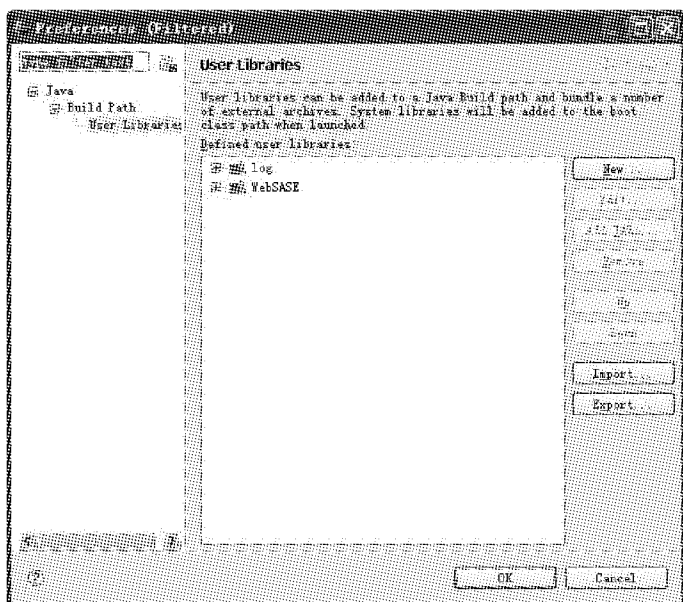


图 3-17 新建用户库对话框

(13) 在弹出的 New User Library 对话框中, 输入用户库的名称, 如“log4j”, 然后单击 OK 按钮, 完成用户库的建立, 如图 3-18 所示。

(14) 选中新建的用户库 log4j, 然后单击 Add JARs 按钮, 选择新建 Java 工程所在文件夹下库文件所在目录中的库文件 log4j.jar, 如图 3-19 所示。

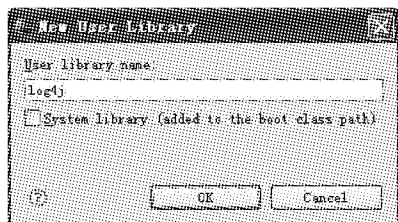


图 3-18 New User Library 对话框

(15) 选中新建的用户库 log4j 前面的复选框, 然