

Java

编程案例精解

孙印杰

刘 斌 等编著

孙玉强



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

Java 编程案例精解

孙印杰 刘 斌 孙玉强 等编著

電子工業出版社·

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Java 语言是一种新型的网络编程语言,其卓越的功能和特性为无数开发人员所推崇,越来越多的应用开发采用了基于 Java 技术的解决方案。Java 语言也是一种真正面向对象的编程语言,它提升了应用程序的编程概念和开发思维。Java 语言提供了丰富的开发类库,为用户编程提供了极大的支持和方便。

本书以实例详解的方式,选取 Java 应用的典型实例循序渐进地介绍了使用 Java 语言开发各种应用程序的方法和技巧。全书共 10 章,第 1 章介绍了 Java 语言的基本概念和常用工具,第 2 章~第 10 章通过 9 个具体的案例,详细介绍了数据库连接池的实现、Taglib 的实现、文件操作技术、Socket 编程、Applet 开发、图像处理技术、多线程的实现和 XML 的应用。

本书内容丰富,结构清晰,实例的选择紧贴实际应用,具有很强的实用性和针对性,是一本易懂、实用、极具参考价值的实例教程。本书主要面向有一定 Java 语言编程基础、准备提高 Java 各种编程技术的读者。对于使用 Java 语言进行开发的工程人员,也是一本不可多得的参考资料。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

Java 编程案例精解 / 孙印杰等编著. —北京: 电子工业出版社, 2005.1

ISBN 7-121-00894-7

I .J... II.孙... III.Java 语言—程序设计 IV.TP312

中国版本图书馆 CIP 数据核字 (2005) 第 005955 号

责任编辑: 祁玉芹

印 刷: 北京市天竺颖华印刷厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 27.5 字数: 672 千字

印 次: 2005 年 1 月第 1 次印刷

印 数: 6000 册 定价: 39.00 元

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。
联系电话: (010)68279077。质量投诉请发邮件至 zltts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

前 言

近年来,随着网络应用的迅速普及,Java 语言作为主流语言在网络编程中的应用越来越广泛,优势越来越明显。Java 是一种程序设计语言,它具有简单的、完全面向对象以及
与平台无关性的结构,也具有可移植性、高性能和安全性,并提供了多线程的功能,因而
受到广大程序员的好评和喜爱。同时,随着 Web 应用程序越来越复杂,处理的数据越来越
多,Web 应用程序的使用也相应地广泛起来。Web 应用程序维护量很大,如何有效地编写
Web 应用程序因此显得非常重要。

本书以实例的形式介绍了 Java 的经典编程范例,力图使已具有 Java 编程基础的读者能够
从书中得到启发,提高自己的应用开发水平。对于每个实例,先分析其需求和相关的技术
背景,再介绍编程思路和设计过程,最后给出实现实例程序的关键代码。整个过程条理
清晰、循序渐进,符合程序设计的自然思路,读者读完一个章节,也就相应地掌握了相关的
Java 的编程思想和技术。

本书共分为 10 章。第 1 章讲述了 Java 编程语言的发展历史,语言的实现机制,常用的
Web 应用服务器以及常用的开发工具。

第 2 章给出了一个基本的连接管理框架,在其中使用了一些广泛使用的设计模式(如资源
池和引用记数等),使得高效、安全地使用数据库连接成为可能。

第 3 章讲述了如何通过标记库实现下拉列表值的变动,通过 Sourceforge.net 上的项目
Providers,让读者很容易实现下拉列表的联动,彻底解决传统的直接在网页中写 Javascript
而造成的不易维护、浪费时间和金钱等问题。

第 4 章是以日常发生的财务状况为实例,讲解了基于 Web 的文件操作的实现,提供了
创建目录、删除文件、删除目录、获得目录大小、获得硬盘上目录的大小、获得字节数、
获得文件名称等功能的文件操作实现。

第 5 章以一个完整的网络聊天程序的实现,介绍了 Socket 编程的相关知识。

第 6 章通过一个基于 Applet 的网页游戏的开发,讲解了 Applet 小程序的基本开发技巧。

第 7 章通过一个基于 Java 的图像处理程序的开发方法,主要演示了如何利用 Java 提
供的 API 进行图像变换、文件的读写等操作。

第 8 章通过一个简单的 HTTP 代理服务器的实现,对多线程的知识进行应用,并且通
过共享数据缓冲池实现了多线程的同步与协作。

第 9 章实现了一个功能完整的邮件客户端。该邮件客户端具有发送与接收简单邮件及
带附件的邮件的功能,全面展示了 Java Mail 开发包的高级应用。

第 10 章给出了一个基于 XML, JSP 和 JavaBean 的在线相册系统,其中使用 JDOM 接
口访问 XML 文件,使用 JSP 页面同用户交互和访问 JavaBean。在 JavaBean 中封装了应用
层逻辑,实现主要的系统功能。

本书内容丰富，结构清晰，实例的选择紧贴实际应用，具有很强的实用性和针对性。· 所选择的 9 个实例，涵盖了数据库连接池的实现、Taglib 的实现、文件操作技术、Socket 编程、Applet 开发、图像处理技术、多线程的实现、XML 的应用等诸多高级应用领域，是一本易懂、实用、极具参考价值的实例教程。

本书由孙印杰、刘斌和孙玉强主持编写。其中孙印杰编写本书的第 1 章和第 2 章，刘斌编写第 3 章，孙玉强编写第 4 章和第 5 章。此外，参加编写的人员还有蓝荣香、王昊亮、喻波和邵蕴秋等。由于编者水平有限，书中难免有疏漏和不足之处，恳请广大读者批评指正。我们的 E-mail 地址：qiyuqin@phei.com.cn。

为了最大限度地降低读者的购书成本，方便读者学习，本书中的素材可免费从网上下载。

网址：<http://www.firstarcicl.com.cn>

编 者
2004 年 12 月

目 录

第 1 章	Java 编程概述.....	1
1.1	Java 语言简介.....	1
1.1.1	Java 语言的特点	1
1.1.2	Java 语言实现机制	1
1.2	Web 应用服务器	3
1.2.1	Resin.....	3
1.2.2	Tomcat	5
1.2.3	WebLogic.....	7
1.3	Java 开发工具.....	9
1.3.1	Together	9
1.3.2	JBuilder	10
1.3.3	Scioworks Camino Pro	11
1.4	本章小结	12
第 2 章	数据库连接池的实现	13
2.1	Java 访问数据库.....	13
2.1.1	JDBC 与数据库的连接.....	13
2.1.2	访问数据库的组件.....	14
2.2	设计思路	17
2.2.1	实例演示.....	17
2.2.2	数据库连接池功能.....	18
2.2.3	UML 图分析	19
2.3	实例代码解析	20
2.3.1	ResourceManager.java 的实现.....	20
2.3.2	UsersDaoImpl.java 的实现.....	30
2.4	本章小结	40
第 3 章	Taglib 的实现——providers	41
3.1	技术介绍	41

3.1.1	JSTL 简介	41
3.1.2	Taglib 原理和实现	42
3.2	设计方案	44
3.2.1	实例演示	44
3.2.2	实例功能	45
3.2.3	UML 图	47
3.3	实例代码解析	48
3.3.1	设计和实现 index.jsp	48
3.3.2	设计和实现 providers.jsp	49
3.3.3	设计和实现 comboselect.jsp	51
3.3.4	设计和实现 comboselect-splitCollections.jsp	53
3.3.5	设计和实现 ComboSelectAction.java	54
3.3.6	设计和实现 web.xml	59
3.3.7	设计和实现 struts-config.xml	61
3.3.8	设计和实现 server.xml	62
3.4	本章小结	64

第 4 章 个人理财系统——FileUtil 的使用65

4.1	文件操作	65
4.1.1	文件对象——File 对象	65
4.1.2	FileWriter 类的应用	67
4.1.3	FileReader 类的应用	67
4.2	设计方案	67
4.2.1	实例演示	67
4.2.2	Model-View-Control	68
4.2.3	UML 图	68
4.3	实例代码解析	70
4.3.1	控制层代码解析	70
4.3.2	模型层代码的实现和分析-Mxz.java	82
4.3.3	数据访问对象的实现(DAO)	83
4.3.4	工具类的实现-Util	93
4.3.5	视图层的实现	96
4.4	本章小结	114

第 5 章 网络通信程序的实现115

5.1	Socket 编程基础	115
5.1.1	TCP/IP 简介	115
5.1.2	TCP 与 UDP 协议的介绍与比较	116

5.1.3	TCP 协议通信的响应流程	117
5.1.4	使用面向连接的流 Socket 通信的实现	117
5.1.5	使用面向无连接的数据包通信的实现	120
5.2	设计思路——聊天室程序的实现	121
5.2.1	实例功能演示	121
5.2.2	实例功能分析	125
5.3	实例代码解析	126
5.3.1	聊天室服务器的实现	126
5.3.2	聊天室客户端的实现	145
5.4	本章小结	153

第 6 章 Applet 网页游戏开发 155

6.1	Applet 开发	155
6.1.1	Applet 的生命周期	155
6.1.2	JApplet 的主要方法	156
6.1.3	Applet 的显示和访问	158
6.2	设计思路	160
6.2.1	实例演示	161
6.2.2	实例功能	161
6.2.3	UML 图分析	162
6.3	实例代码解析	163
6.3.1	TetrisApplet.java 的实现	163
6.3.2	Brick3D.java 的实现	170
6.3.3	Board.java 的实现	179
6.3.4	TetrisApplet.html 文件的实现	196
6.4	本章小结	196

第 7 章 图片编辑器实现 197

7.1	Java 图形图像处理	197
7.1.1	图形图像处理相关类	197
7.1.2	Swing 组件包	201
7.2	设计思路	201
7.2.1	实例演示	202
7.2.2	实例功能	202
7.2.3	UML 图分析	205
7.3	实例代码解析	206
7.3.1	PictureDrawer.java 的实现	207
7.3.2	ImageFrame.java 的实现	221

7.3.3 JAdvLabel.java 的实现	240
7.3.4 生成 JavaDoc 文件	244
7.4 本章小结	246

第 8 章 多线程的实现247

8.1 Java 多线程基础.....	247
8.1.1 Java 多线程概述.....	247
8.1.2 Java Thread 类和 Runnable 接口.....	248
8.1.3 线程的通信.....	249
8.1.4 线程的同步与 synchronized 关键字	249
8.1.5 线程的等待与唤醒.....	250
8.2 设计思路——HTTP 代理服务器.....	250
8.2.1 实例功能演示.....	251
8.2.2 实例功能分析.....	253
8.3 实例代码解析.....	253
8.3.1 服务器主框架类的实现.....	253
8.3.2 信息显示类.....	257
8.3.3 服务器侦听线程类.....	258
8.3.4 用户请求转发线程类.....	260
8.3.5 数据缓冲类.....	265
8.3.6 数据接收线程类.....	267
8.3.7 数据转发线程类.....	269
8.4 本章小结.....	270

第 9 章 Java Mail 高级编程271

9.1 Java Mail 基础	271
9.1.1 POP3 协议、SMTP 协议与 E-Mail	271
9.1.2 Java Mail 开发包简介	272
9.2 设计思路——邮件客户端的实现.....	276
9.2.1 实例功能演示.....	276
9.2.2 实例功能分析.....	277
9.3 实例代码解析.....	278
9.3.1 应用程序类.....	278
9.3.2 主框架类.....	279
9.3.3 属性结构类.....	282
9.3.4 邮件结构类.....	284
9.3.5 邮件处理类.....	286
9.3.6 自定义标签类.....	290

9.3.7 属性配置类.....	291
9.3.8 邮件发送类.....	298
9.3.9 邮件接收类.....	309
9.4 本章小结.....	322

第 10 章 在线相册系统——Java 与 XML 323

10.1 XML 基础.....	323
10.1.1 XML 简介.....	324
10.1.2 支持 XML 访问的几个类.....	325
10.1.3 使用 Java 访问 XML.....	326
10.1.4 JDOM 接口.....	330
10.2 设计思路.....	332
10.2.1 实例演示.....	332
10.2.2 实例功能.....	333
10.2.3 UML 图分析.....	339
10.3 实例代码解析.....	343
10.3.1 程序主页面.....	343
10.3.2 config.jsp.....	347
10.3.3 style.css 式样文件.....	348
10.3.4 用户登录 long.jsp.....	349
10.3.5 DoLogin.jsp.....	352
10.3.6 UserAdmin.java 文件.....	354
10.3.7 regedit.jsp 文件.....	365
10.3.8 Register.jsp 文件.....	373
10.3.9 modify.jsp 文件.....	374
10.3.10 DoModify.jsp 文件.....	375
10.3.11 CheckPermission.jsp.....	376
10.3.12 CheckAdminPermission.jsp 文件.....	376
10.3.13 logout.jsp 文件.....	377
10.3.14 msg.jsp 文件.....	378
10.3.15 addmessage.jsp.....	380
10.3.16 removemsg.jsp 文件.....	381
10.3.17 Doremovemsg.jsp 文件.....	382
10.3.18 Message.java.....	383
10.3.19 MessageAmin.java 文件.....	388
10.3.20 upload.jsp 文件.....	394
10.3.21 upload_post.jsp 文件.....	395
10.3.22 browsephoto.jsp 文件.....	396
10.3.23 photodetails.jsp 文件.....	399
10.3.24 addcomment.jsp 文件.....	401

10.3.25	Picture.java 文件	402
10.3.26	PictureAmin.java 文件	410
10.3.27	FileUpload.java 文件	421
10.3.28	Comment.java 文件	427
10.3.29	CommentAdmin.java 文件	427
10.4	本章小结	430

第 1 章 Java 编程概述

Java 语言是一种新型的网络编程语言，其卓越的特性为广大开发人员所推崇，越来越多的应用开发采用了基于 Java 技术的解决方案。Java 语言也是一种真正面向对象的编程语言，它创造了应用程序的编程新概念和开发新思维。

本章主要介绍 Java 语言的发展历史、语言特点、Web 应用服务器和 Java 主要的开发工具。

1.1 Java 语言简介

Java 语言是一种网络编程语言，它最大限度地利用了网络资源。Java Applet 小应用程序可以跨平台、跨系统和跨网络运行。Java 语言提供了丰富的开发类库，为用户编程提供了极大的支持和方便。

1.1.1 Java 语言的特点

在 Java 诞生时，世界上已有上千种不同的编程语言，而当时最为流行的是 C++。由于 C++是从 C 语言发展而来，它背负着沉重的包袱，不能做到完全面向对象。Java 吸收了大量 C++的优秀特性，除去了那些模糊、复杂、容易出错的特性，如指针、内存申请和释放等，并引入了很多独特的高级特性。

作为一种程序设计语言，Java 是分布式的、面向对象的、不依赖于机器的，具有可移植性、鲁棒性、安全性，并提供了多线程并发的机制，以及动态下载程序代码的机制，还提供代码检验机制以保证安全性。它能最大限度地利用网络，Java Applet 可在网络上传输而不受 CPU 和环境的限制。另外，Java 代码清晰合理、简明流畅，还提供了丰富的类库，使程序设计者可以很方便地建立自己的系统。

总之，Java 是一种编程语言、一种开发环境、一种应用环境、一种部署环境、一种广泛使用的网络编程语言，它是一种新的计算概念。在面向对象的程序设计(OOP)中，使用 Java 语言的继承性、封装性、多态性等面向对象的特性可以较好地实现信息隐藏、对象的封装，从而降低程序的复杂性，实现代码的重用，提高开发效率。

1.1.2 Java 语言实现机制

1. Java 虚拟机(Java Virtual Machine)

Java 虚拟机(JVM)是在一台计算机上由软件模拟或用硬件来实现的假想的计算机。它定义了指令集(相当于中央处理器 CPU)、寄存器集、类文件结构栈、垃圾收集堆、内存区域。

首先，Java 编译器在获取 Java 应用程序的源代码后，把它编译成符合 Java 虚拟机规范的字节码(byte code)的 class 文件，class 文件是 JVM 中可执行文件的格式。Java 编译器针对 Java 虚拟机产生 class 文件，Java 虚拟机规范为不同的硬件平台提供了不同的编译代码规范，

该规范使 Java 软件独立于平台。然后, Java 解释器负责将 Java 字节码文件解释运行。但是,它是边解释边运行的,这样,运行速度会受到一定影响。为了提高运行速度,Java 提供了另一种解释运行方法 JIT(Just In Time),可以一次解释完,再运行特定平台上的机器码,这样就实现了跨平台、可移植的功能。

在 Java 的运行环境中,每个 Java 解释器,不管它是 Java 开发工具,还是可运行的 Web 浏览器,都可执行 Java 虚拟机。字节码的运行要经过下面 3 个步骤。

- (1) 加载代码: 由 Class loader 完成。
- (2) 校验代码: 由 Bytecode Verifier 完成。
- (3) 执行代码: 由 Runtime Interpreter 完成。

2. 垃圾回收机制(Garbage Collection)

Java 语言中的任何事务都是封装在类中的,每个类都会创建一个或多个实例对象,每个对象都有生命周期,我们需要时就去创建、调用它,不用时就应销毁它。这种动态的实例对象是被存放在内存堆(Memory Heap)中的。随着科技的发展,我们可以不断更新和扩展硬件设施,但任何存储介质都是有极限的,内存也不例外,对于不再使用的对象,我们应该将其销毁,从而释放资源。

许多编程语言都允许在程序运行时动态分配存储器。分配存储器的过程由于语言句法的不同而不同,当不再需要分配存储器或存储器指针溢出范围时,程序或运行环境应停止继续分配存储器,进行内存回收。但如何进行内存回收是一件很困难、很复杂的事情,在 C、C++ 或其他语言中,程序员负责回收已分配的内存。由于存储器是动态分配的,通常我们无法准确判断存储器应在何时被释放。这就为程序运行留下隐患,当系统运行中没有能够被分配的存储器时,就会导致程序瘫痪。

Java 解除了程序员释放已分配存储器的责任,通过提供一个系统级线程对内存使用进行跟踪。由于 Java 是单根结构,任何一个类都直接或间接地继承于 Java.lang.Object 类,所以系统级线程可以跟踪每一次存储器的分配情况,并且可逐级回溯,定期检测出不再使用的内存,在系统空闲时自动进行回收。垃圾回收是在 Java 程序的生命周期中自动进行的,我们无法判断垃圾回收线程何时启动,要执行多长时间,这使得程序在运行期间出现一种不连贯的状态,并且在一定程度降低了程序的运行效率,但这个代价还是值得的。

3. 代码安全性检测(Code Security)

对 Java 代码的安全性的考虑最初来源于 Java Applet,由于它可以被轻易地下载到 Web 浏览器上运行,改变了以往传统模式下桌面计算机系统的应用软件的更新方式。这种移动代码可以经 Internet 的 Web 浏览器自动下载和更新,但是在享受便捷快速的技术更新的同时也带来了安全隐患。Java 1.0 采用了沙箱(sandbox)安全模型,它的主要思想是像 Applet 这样通过远程下载的代码只能受限地访问系统资源,如不能访问本地文件、不能建立新的网络连接等,它们的行为受限于沙箱之中。由于这个模型约束太多,大大削弱了移动代码的优势。在 Java 1.1 中采用了信任安全模型,其主要思想是用户可以有选择地授权给远程代码,如果远程某地的资源可以被信任,那么带有本地签名的代码将被授权可以访问本地的系统资源,其他不被信任的代码仍受限在沙箱内。Java 2 平台则采用了域管理方式的安全模型,无论是本地代码还是远程代码都可以通过配置的策略,设定可访问的资源域,这种策略更好地支持了企业级应用,同时也消除了区分本地代码和远程代码带来的困难。

Java 体系结构中不仅定义了 Java 的开发编译环境, 也定义了 Java 的运行环境。为运行 Java 应用程序和 Applet, 计算机上应安装 JVM 和 Java 运行时解释器, 这两个部分构成 Java 的运行环境, 其主要任务包括加载代码(由类加载器执行)、校验代码(由字节码校验器执行)和执行代码(由运行时解释器执行)。

1.2 Web 应用服务器

Web 应用服务器(或简称为应用服务器)是为创建、部署、运行、集成和管理事务性 Web 应用提供一系列运行时服务(如消息、事务、安全、应用集成等)的分布式系统。Web 应用服务器是分布式中间件软件实施产品, 其作用相当于一个接口, 连接一端的各种类型的客户机, 以及另一端的各类后台系统资源, 具有一整套集成分布式计算能力的软件服务器产品。它管理客户请求, 为业务逻辑提供宿主环境, 连接数据、事务处理和目录等后端计算资源。

Web 应用服务器有如下特点。

(1) 可伸缩性

可伸缩性指应用服务器应付动态站点流量, 并为不断增长的客户请求提供服务的能力。Web 流量的不确定性是服务器管理程序的潜在问题, 如果处理不当, 很容易打破服务器处理的能力底线, 导致系统崩溃。

(2) 负载均衡

负载均衡指向同一集群中不同服务器均衡分配客户请求的能力。简单的、具有负载均衡能力的硬件解决方案早见于智能化程度不高的请求路由安排中。但应用服务器与路由安排不同, 它的负载均衡能力要高得多, 其均衡分配的是应用组件而非客户请求。

(3) 容错

容错指当出现组件故障、网络故障或服务器崩溃时, 能够维持或重新启动系统, 从而使它们能继续正常运转的能力。通常, 多台服务器被配置成一个具有协同工作能力的集群, 当集群中一台服务器发生故障时, 其他服务器可以立即接替承担附加负载。

(4) 事务处理监测

应用服务器应具备高效执行跨服务器及跨网络事务处理的能力。两段提交模式对特定的企业事务处理至关重要。监测软件设计的好坏以及能否确保事务处理的成功完成应该作为衡量应用服务器性能的一个重要指标。

(5) 资源的可用性

应用服务器的资源——诸如内存、计算能力、组件、数据库和其他后端系统——应该在任何时候都能为客户请求提供服务。确保客户请求不会因为服务器计算能力不够或缺少所需服务组件而导致请求超时失败是应用服务器的职责所在。

Web 应用服务器除了以上特点外, 还有价格、安全等方面的特点, 这里就不做讲述。

1.2.1 Resin

Resin 是目前应用比较广泛的免费的应用服务器, 用户可以从 <http://www.caucho.com> 下载, 使用和配置起来都比较简单, 适用于一般应用, 支持 JSP、Servlet 的一般应用。下载完毕后, 解压软件至系统盘中, Resin 的软件结构如图 1-1 所示。

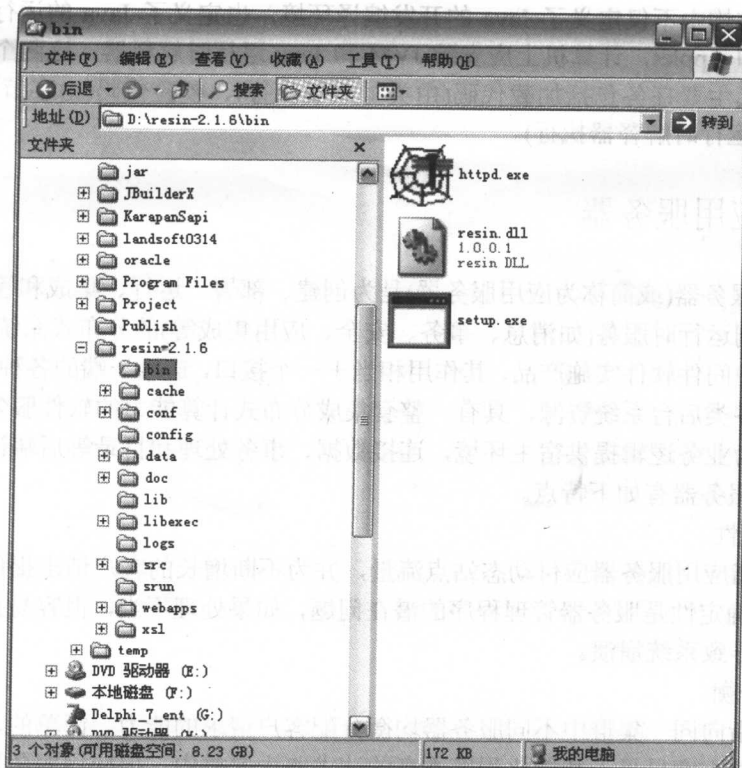


图 1-1 Resin 的软件结构图

用户安装完毕后，双击 Resin 的 bin 目录下的 httpd.exe 启动服务，显示如图 1-2 所示的窗口。

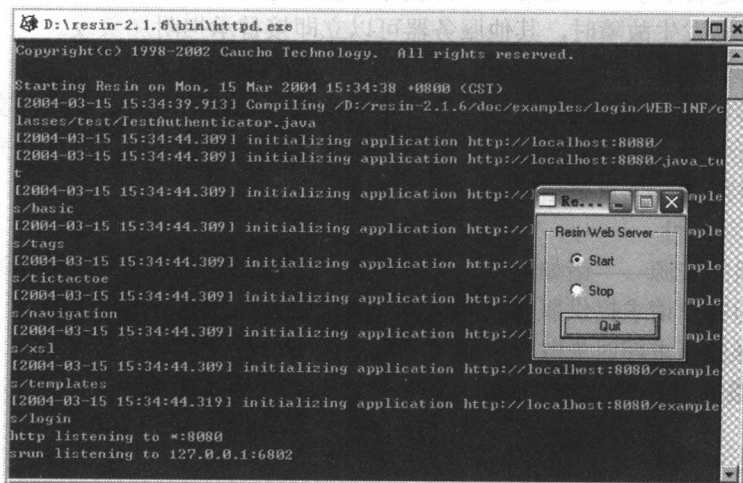


图 1-2 Resin 的后台服务

显示图 1-2 的界面以后，Resin 的服务已经正常启动。Resin 的默认端口为 8080，用户在浏览器的地址栏中输入 <http://localhost:8080/>，显示如图 1-3 所示的界面，即 Resin 提供的首页。

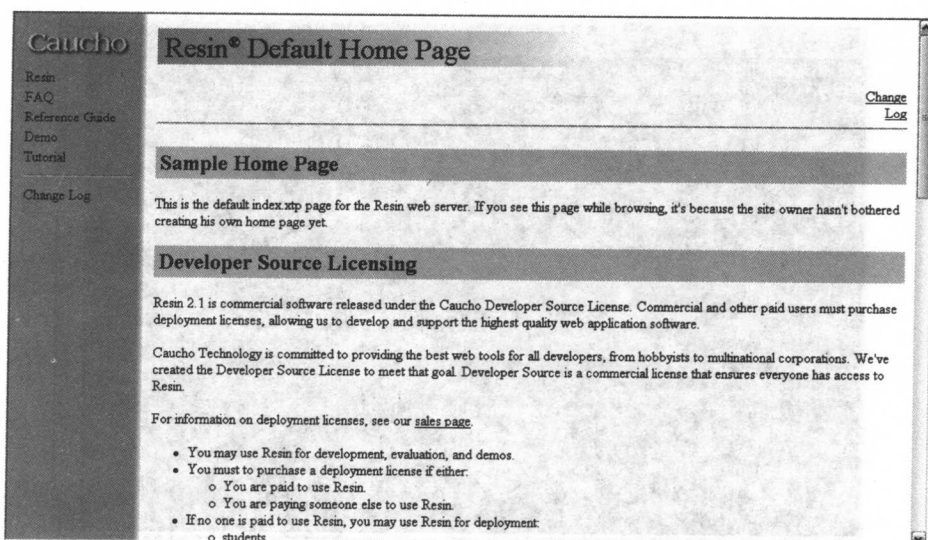


图 1-3 Resin 的首页

Resin 首页的显示，证明用户已经成功配置 Resin。

1.2.2 Tomcat

Tomcat 和 Struts 一样都是 Jakarta 组织提供的，属于开源软件的一种。Tomcat 可以从 <http://jakarta.apache.org> 直接下载，用户可以下载安装版本，也可以下载解压版本直接使用。用户安装完毕以后，Tomcat 的程序目录如图 1-4 所示。

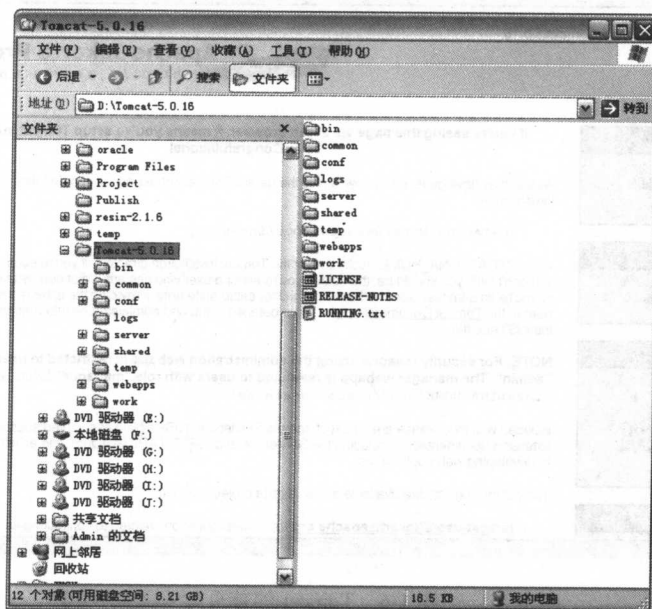


图 1-4 Tomcat 的程序目录

双击图 1-4 中 bin 目录下的 startup.bat，弹出如图 1-5 所示的窗口。

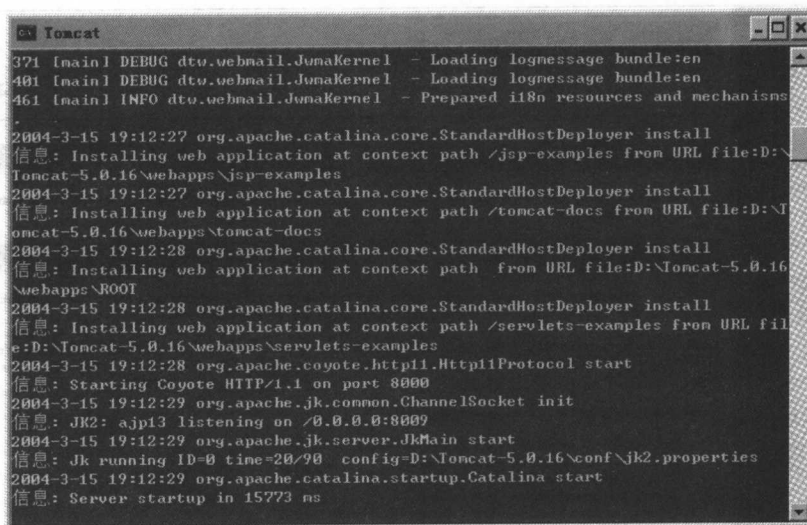


图 1-5 Tomcat 应用程序服务器启动界面

显示图 1-5 的界面以后，Tomcat 的服务已经正常启动。Tomcat 的默认端口为 8000，用户在浏览器的地址栏中输入 <http://localhost:8000/>，显示如图 1-6 所示的界面，即 Tomcat 的首页。

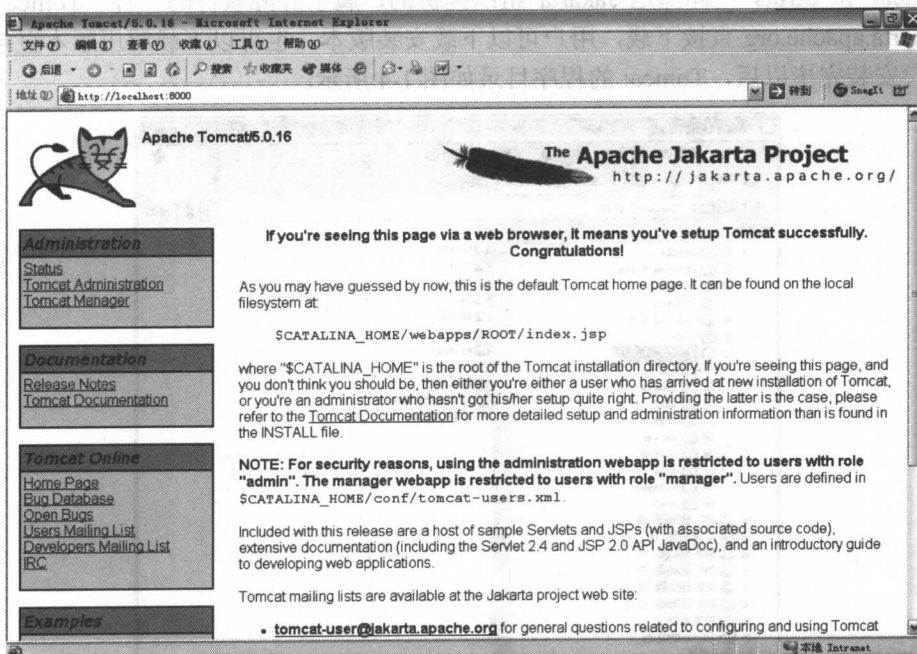


图 1-6 Tomcat 的首页

用户单击 Tomcat Administration 超链接，弹出如图 1-7 所示的网页，这是 Tomcat 的管理程序登录网页。