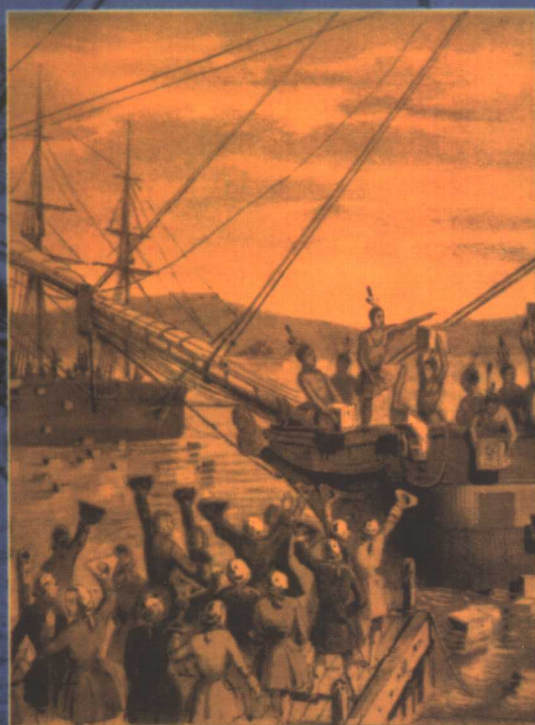


Java Application Development on Linux

PRENTICE
HALL
PTR

Linux 应用程序开发 (Java 版)



学会在Java开发和部署平台Linux上编写专家级代码！

- 在 Linux 环境下如何创建、管理和部署 Java 应用程序
- 在Linux环境下如何充分利用Java开发工具来最大化效率，这些工具包括 gcj、cvs、Ant、JUnit，以及 NetBeans IDE 和 Eclipse IDE
- 结合 Sun 的 Swing 和 IBM 的 SWT 组件介绍了 GUI 开发

(美) Carl Albing
Michael Schwarz
侯普秀 宋美娜

著
译



清华大学出版社

Linux 应用程序开发

(Java 版)

(美)	Carl Albing	著
	Michael Schwarz	
	侯普秀 宋美娜	译

清华大学出版社

北 京

Simplified Chinese edition copyright © 2006 by PEARSON EDUCATION ASIA LIMITED and
TSINGHUA UNIVERSITY PRESS.

Original English language title from Proprietor's edition of the Work.

Original English language title: Java Application Development on Linux, by Carl Albing, Michael
Schwarz, Copyright © 2005

EISBN: 0-13-143697-X

All Rights Reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison
Wesley.

This edition is authorized for sale only in the People's Republic of China (excluding the Special
Administrative Region of Hong Kong and Macao).

本书中文简体翻译版由 Pearson Education(培生教育出版集团)授权给清华大学出版社在中国境内(不包括
中国香港、澳门特别行政区和中国台湾地区)出版发行。

北京市版权局著作权合同登记号 图字: 01-2005-3430

版权所有, 翻印必究。举报电话: 010-62782989 13501256678 13801310933

本书封面贴有 Pearson Education(培生教育出版集团)激光防伪标签, 无标签者不得销售。

图书在版编目(CIP)数据

Linux 应用程序开发(Java 版)/(美)阿尔宾(Albing, C.)(美)施瓦兹(Schwarz, M.)著; 侯普秀, 宋美娜 译.

—北京: 清华大学出版社, 2006. 12

书名原名: Java Application Development on Linux

ISBN 7-302-13750-1

I. L… II. ①阿…②施…③侯…④宋… III. ①Linux 操作系统②JAVA 语言—程序设计

IV. ①TP316.89②TP312

中国版本图书馆 CIP 数据核字(2006)第 105281 号

责任编辑: 王 军 李 阳

封面设计: 康 博

版式设计: 康 博

责任印制: 何 芊

出版发行: 清华大学出版社 地 址: 北京清华大学学研大厦

<http://www.tup.com.cn> 邮 编: 100084

c-service@tup.tsinghua.edu.cn

社 总 机: 010-62770175 邮购热线: 010-62786544

投稿咨询: 010-62772015 客户服务: 010-62776969

印 刷 者: 北京市清华园胶印厂

装 订 者: 北京市密云县京文制本装订厂

经 销: 全国新华书店

开 本: 185×260 印 张: 24.25 字 数: 590 千字

版 次: 2006 年 12 月第 1 版 印 次: 2006 年 12 月第 1 次印刷

印 数: 1~4000

定 价: 45.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调
换。联系电话: (010)62770177 转 3103 产品编号: 019257-01

绪 言

Java 与 Linux

为什么要编写另一本关于 Java 的书呢？为什么是关于 Java 以及 Linux 的一本书呢？Java 不是独立于系统平台的吗？关于 Java 的书还不够多吗？我可以从网络上学到所需要的一切吗？

毫无疑问，市面上已有许多关于 Java 的书。我们不是在某个早晨醒来，然后说：“您知道世上缺什么吗？另一本关于 Java 的书”。我们已意识到市面上的 Java 书存在许多的“漏洞”。

首先，作为一种开发平台，一种 Java 应用程序的开发平台，Linux 在很大程度上被忽略了。尽管一直以来，*nix 平台(包括所有的 UNIX 以及类 UNIX 系统，Linux 也在其中)被认为是现有的对程序员最友好的平台。Linux 下的 Java 资源很少会强调工具中排斥 Java 语言以及 API。

其次，关于 Java 以及 API 的书大都关注用来说明语言以及语言库的细节的教学示例，但是示例本身很少具有实际价值，它们只是处理编程问题，根本不考虑部署以及程序的维护问题。参加过重要软件项目(特别是以商业形式开发并部署的商业软件项目)的任何人都知道设计以及编码仅仅完成了一半的工作。是的，编写 Java 代码很少受开发部署平台的影响，但是在不同的平台上发布以及维护这种应用程序有明显的不同。

为了弥补这个缺陷，我们决定讲述 Java 应用程序的开发以及部署，这些程序具有在 Linux 平台下的命令行、GUI、servlet 以及企业组件。我们在开始编写并部署 Linux 下的 Java 应用程序时希望有一本指南书，本书就是这样一本书。我们将展示一些简单的企业应用程序，在此过程中将讨论设计过程、产品环境、安装、管理以及维护，其他书籍很少涉及到这些问题¹。

如果您正在考虑买这本书，并且想知道这本书的全部信息是否都可以在网络上免费获得，答案是“不可以”。事实上，Java 以及 Linux 书籍的内容几乎全部可以从网络上找到。不过在每一章我们都会告诉您在那个网站可以找到我们这里所讲述的全部内容。然而我们还是向您推荐这本书。原因是网络上的信息是分散的、无条理的，并且有很大的变数。我们试图将所有的相关信息以清晰而条理的方式(并且我们相信是以相对较高的质量)组合在本书中。我们认为这样做是有价值的。

1. 这并不是说本书没有纯粹的教学示例。特别是在第一部分，我们使用了“一次性”示例以及单独的类来说明基本内容，使用完整的应用程序会使得将要说明的问题变得晦涩混乱。

另外,本书是 Bruce Peren 开放源代码的一部分,也是网络文献的一部分。您可以在网络上自由地阅读并使用它。我们希望您购买本书,并在网络上使用本书。我们对您不了解,但我们喜欢使用网络资料作为参考,然而在阅读时还是喜欢书本。我们至少拥有三本可以在网络上免费获得的书: *Thinking in C++*、*Think in Java* 以及 O'Reilly 的 *Docbook: The Definitive Guide*。

自由软件以及 Java

GNU/Linux 是自由软件²。它是开放源代码的。我不想讨论每个术语的含义以及哪一个术语是“正确的”。本书的作者一个是自由软件的拥护者,另一个则对这个问题持纯粹的放任态度(我们不会告诉您是那一个,请您自己猜)。然而即使有这样精心安排的停火协定,问题仍然存在: Java 是开放源代码的还是自由软件?

答案是混乱的。Sun 以及 IBM 的 Java 实现都不是开放源代码的,也不是自由软件。您可以免费下载并使用它们,但是没有它们的源代码,也没有权利去修改它们³。本书将讲述 GNU 的 Java 编译器,它可以将 Java 源代码编译为本地机器代码。GUN 的 Java 编译器是开放源代码的,也是自由软件。然而,它支持不同层次的 Java API(某些包是当前的,某些在 1.1.x 层次之后),并且不完全支持 AWT 或者 Swing GUI。

然而,这并不意味着在自由软件或者开放源代码许可下,您不能编写自己的程序并将其发布。因此您可以使用 Java 开发自由软件。坚定的自由软件派别(如 Richard 以及自由软件基金)可能会怀疑这样做是否明智。他们的观点是,依赖于非自由工具的自由软件产品不是真正的自由软件,因为在编译、使用或者修改它的时候,需要使用具有所有权的工具。

许多人都致力于开发自由软件的 Java 运行库实现。在我们看来, Sun 将 SDK 以及 Java 虚拟机作为自由软件发布是非常好的一件事。但是到目前为止,他们仍然坚决反对这样做。

事实上,这两个截然不同的供应商(Sun 以及 IBM)提供了可以充分互换的开发平台以及运行库环境,从而降低了您选择只使用一个供应商(不提供源代码)的开发平台的风险。

因此要坚信: Java 是可以自由使用的,但是它不是 GUN Manifesto⁴或者 GUN 通用公共许可证⁵定义的自由软件。对于前面提过的自由软件派别而言,这是一个有趣的政治和哲学问题。对于其他人而言,这并不影响 Java 的技术或者商业价值。很明显,我们十分喜欢这门语言,否则不会编写关于它的书籍。

2. 这是唯一一次我们称它为“GNU/Linux”。参考 7.3 节来了解为什么 GNU/Linux 是众多名称中的首选。我们理解 Stallman 以及 FSF 的立场,但是“Linux”看起来比“GNU/Linux”顺眼得多。这就是名称以及单词进入语言的方式(并非准则)。无论如何,“Linux”是这个操作系统的名称。

3. 在我们写这本书的时候, Sun、IBM 以及开放源代码的奠基人 Eric Raymond 之间关于在某种开放源代码许可下开放 Java 开展了一场广泛的讨论。此时,没有人知道结果如何,但是在将来 Java 可能会成为自由软件。

4. <http://www.gnu.org/gnu/manifesto.html>。

5. <http://www.gnu.org/copyleft/gpl.html>。

您可以参与

本书是 Bruce Peren 开放源代码系列的一部分。当本书以固定形式出版后，它很快就会出现在网络上⁶，它遵循公开出版物许可证的规定⁷，可以被自由使用、重新发布、修改，不会有任何问题。许可证还允许您建立自己的版本。

很自然，我们计划保留在线书籍的“官方”版本，希望您向我们提供建议、修正、扩充、评论以及思想。请将之发送到 Javalinux@multitool.net。我们将维护这与时俱进的小册子，来满足 Java 以及 Linux 开发团体的需要。

6. <http://www.javalinuxbook.com/>。

7. <http://www.opencontent.org/openpub/>。

前言

本书针对两类不同的读者。我们意识到，本书可能会吸引不熟悉 Linux 而有 Java 经验的程序员，以及不熟悉 Java 而有 Linux 经验的程序员，或者二者之间的其他可能。

除了平衡这两个极端之外，我们还试图在书的篇幅以及话题的范围之间获取平衡。幸运的是，现在有许多关于 Java 和 Linux 的书籍以及网络资料，因此我们可以在篇幅有限的情况下尽力而为，我们将求助于许多外部参考资料，您可以用这些资料来补充本书的内容。

本书读者对象

如果您是一位有经验的 Java 程序员，但是不熟悉 Linux，而且您已经花费了很长时间来寻找在 Linux 下开发并部署 Java 应用程序工具的信息，那么本书将为您提供大量有价值的信息。

如果您是一位经验丰富的 Linux 用户或者开发人员，并且有兴趣在这个平台上使用 Java，本书将为您讨论一些 Java 开发的高级话题，并提供常见的 Linux 以及 GNU 工具的一些新颖用法。

如果您刚开始接触 Linux 以及 Java，这本书仍然有价值，但是我们建议将其与更多的入门书一起使用。对于 Java 以及面向对象程序设计的介绍，我们推荐 Bruce Eckel 的名著 *Thinking in Java*(ISBN 0-13-100287-2)。对于介绍 Linux 及其工具的书，我们推荐 David Elboth 的 *The Linux Book*(ISBN-0-13-032765-4)¹。本书的“资源”部分列出了一些其他书籍。我们推荐的许多书籍并不局限于 Linux。因为 Linux 与 UNIX 平台相似(在很大程度上)，我们有时候也会推荐对所有的 *nix 系统都适用的书籍。

如果您是一位开发人员、承包人或者项目多出预算的 MIS 开发管理人员，本书将向您介绍许多可靠的工具，这些工具具有免费许可证，可以自由地使用它来开发并部署 Java 应用程序产品。我们总是被要求“多快好省”。在许多(当然不是所有)情况下，开放源代码的自由软件是做到这点的优秀方法。

如何使用本书

使用一本书可以有多种方式。有些人喜欢从最后一章开始，来观察它最后的结论。另一些人喜欢从前面开始，在深入之前掌握所有的内容。一些人喜欢快速地通读，然后详细阅读，另一些人喜欢忽略一些内容，选择感兴趣的话题来阅读。无论您采用哪种方式，我们都希望本书对您有所帮助。

1. 注意，我们倾向于推荐 Pearson Education 的书籍，但是这并不意味着我们会局限于这个出版社。

本书的每一章都不是真正独立的，也不是杂乱地与前面的章节联系在一起。如果我们就一个话题深入阐述，或许可以一章接一章地编写。但我们却将众多的内容浓缩在一起，给出一个好的总结，从而让您了解这些可能性，并鼓励您寻找更多的资料。一些章节与前面章节联系得非常紧密，其他一些章节不需要前面章节的知识就可以阅读——这取决于所讨论的话题。

阅读本书的方法

下面介绍一些阅读本书的方法，这取决于您的经验以及技巧、耐性以及持久性。我们试图将许多有价值的实用信息压缩在少数章节，对每个课题都提炼出最重要的话题。我们希望即使对于非常有经验的读者，也能提供一些新的内容，至少提供了一些新的解释，可以让您熟悉的话题有新的认识。

Linux 新手学习本书的方法

如果您不熟悉 Linux，那么应该从前两章开始学起。如果已经熟悉了 Java，则可以略过第 3 章，但是至少应该浏览第 4、5 章。绝对有必要阅读第 7 章，我们确信在这一章您能学到不少新知识。

第 8 章讨论的是您在 Linux 之外的环境中没有遇到过的一个话题，尽管 CVS 并不局限于 Linux 环境。除此之外，这一章的学习也取决于您已经掌握的内容。

Java 新手学习本书的方法

如果您不熟悉 Java，那么一定要阅读第 3、4 章，但是如果您还不是一个有经验的程序员，或许应该阅读更多相关的入门书籍。

第 5、10 章将会介绍一些背景知识，以便您选择自己的 Java 开发工具。现在，许多的 Java 项目都与 Ant 有关，如果不知道这个工具，还应该阅读第 9 章。

如果具有其他语言的编程经验，您可能已经做过许多单元测试，在第 13 章介绍了一些 Java 开发人员最常用的方法。

客户端的方法

您或许想将本书一半的内容浓缩为几段话，这取决于您所需要的 Java 开发类型。对前端或者客户端感兴趣的读者应该关注中间章节。您最感兴趣的可能是第 16、17 章。客户端的重点应该包括第 18、19 章。

服务器端的方法

对于那些重点在中间层以及第三层的读者，或者重点在通用服务器的读者，第 IV 部分的所有内容都是有价值的。除了第 I、II 部分的前面章节之外，这是另外一些基础知识。

企业的方法

第 V 部分将讨论企业级的软件。这种软件通常包括 JSP 以及 Servlet 软件，在第 18~20 章将对此进行讨论。

对于这个层次上的工作，项目通常很大，需要各类不同的人员。即使您的角色不包括部署软件，我们也鼓励您阅读第 20 章和第 24 章，从而理解企业软件需要什么，如何将其组合在一起。

现在我们开始介绍这些内容，并揭示当您将计算机历史上功能最强大的两个软件(Java 以及 Linux)结合在一起的时候，您可以使用的一些奇妙功能。

目 录

第 I 部分 基础知识

第 1 章 Linux 环境	3
1.1 本章内容	3
1.2 命令行	3
1.3 Linux 的基本概念和命令	4
1.3.1 重定向 I/O	5
1.3.2 ls 命令	7
1.3.3 文件名	7
1.3.4 权限	8
1.3.5 文件复制	8
1.3.6 通配符	10
1.3.7 文件内容	11
1.3.8 grep 命令	13
1.3.9 find 命令	14
1.3.10 再论 shell	15
1.3.11 tar 和 zip 命令	19
1.3.12 man 命令	20
1.4 本章小结	21
1.5 内容扩展	21
1.6 参考文献	21
第 2 章 编辑器	22
2.1 概述	22
2.2 了解 vi	22
2.2.1 退出	26
2.2.2 查找和替换	27
2.2.3 正则表达式	28
2.2.4 向右缩进: .exrc	29
2.3 品种繁多的编辑器	30
2.3.1 sed 或 stream editor	31
2.3.2 简单的图形编辑器	31
2.4 本章小结	32
2.5 有关正则表达式的补充说明	33
2.6 参考文献	33

第 3 章 Java 介绍	34
3.1 本章内容	34
3.2 基本的语言元素	34
3.2.1 数值类型	35
3.2.2 对象类型	38
3.2.3 语句	43
3.2.4 Java 风格的错误处理	49
3.2.5 print()、println()和 printf()	52
3.3 使用 Java API	54
3.3.1 package 语句	55
3.3.2 import 语句	56
3.4 封装、继承及多态性	57
3.4.1 封装	57
3.4.2 继承	57
3.4.3 多态性	60
3.5 模板	62
3.6 virtually 和 final 关键字	63
3.7 一个简单的应用程序	63
3.8 本章小结	65
3.9 内容扩展	65
3.10 参考文献	65
3.11 练习	66

第 4 章 执行语境	67
4.1 本章内容	67
4.2 简单的开始	67
4.2.1 命令行参数	67
4.2.2 单元测试	69
4.3 System 类	69
4.3.1 Java 和标准 I/O	69
4.3.2 环境变量	70
4.3.3 Java 和环境变量	71
4.4 Properties 类	73

4.5	Runtime 类	75
4.5.1	exec()	75
4.5.2	可移植性	76
4.6	本章小结	76
4.7	内容扩展	77
4.8	参考文献	77
第 5 章	Sun Microsystem Java 软件开发包	78
5.1	本章内容	78
5.2	建议	78
5.3	Java 编译器	79
5.3.1	编译器行为、默认值和 环境变量	79
5.3.2	javac 选项	81
5.4	Java 运行库引擎	82
5.4.1	基础	82
5.4.2	java 选项	83
5.5	程序文档	83
5.5.1	运行 javadoc	83
5.5.2	javadoc 命令行选项	85
5.5.3	javadoc 注释	87
5.6	applet	89
5.7	本地化	89
5.8	RMI	93
5.8.1	RMI 简介	93
5.8.2	rmic 工具	97
5.8.3	rmiregistry 工具	98
5.8.4	设置服务程序和客户程序	98
5.8.5	RMI 概要	100
5.9	Java 调试器	100
5.10	Java 反编译器	107
5.11	绑定 Java 程序	110
5.11.1	部署应用程序	110
5.11.2	基本 jar 操作	112
5.12	其他工具软件	113
5.13	本章小结	114
5.14	内容扩展	114
5.15	参考文献	114

第 6 章	Linux 下的 IBM 开发工具包, Java 2 技术版	115
6.1	本章内容	115
6.2	多个 Java SDK 共存	115
6.2.1	链接	115
6.2.2	利用符号链接切换 Java 版本	119
6.3	IBM JDK 与 Sun JDK 的 不同之处	120
6.3.1	性能	120
6.3.2	命令上的不同	120
6.3.3	IBM 类	121
6.4	“_g”的用途	121
6.5	本章小结	121
6.6	内容扩展	121
6.7	参考文献	121
第 7 章	GNU 编译器	122
7.1	本章内容	122
7.2	GNU 的发展	122
7.3	GNU 编译器集合	122
7.4	用 gcj 编译简单应用程序	123
7.4.1	用 gcj 编译 FetchURL	124
7.4.2	编译多类程序	125
7.5	选项和开关	125
7.6	使用 gcj 的理由	126
7.7	不使用 gcj 的理由	127
7.8	本章小结	127
7.9	内容扩展	127
7.10	参考文献	127
第 8 章	CVS	129
8.1	本章内容	129
8.2	源代码控制的原因和方法	130
8.2.1	安装	132
8.2.2	导入	132
8.2.3	一般用法	134
8.2.4	更新	137
8.2.5	合并	137
8.2.6	日志	138

8.2.7 CVS 状态	140	10.7 参考文献	175
8.2.8 cvs tag	140	第 II 部分 开发业务逻辑	
8.2.9 分支标签	141	第 11 章 平衡的掌握	
8.2.10 cvs export 命令	143	179	
8.2.11 效率	143	11.1 本章内容	179
8.3 jCVS	143	11.2 需求	179
8.4 本章小结	145	11.3 如何开发软件	180
8.5 内容扩展	145	11.4 如何产生好的需求	181
8.6 参考文献	146	11.5 需求询问的对象	182
第 9 章 ant 简介		11.6 预算应用程序的需求	182
147		11.6.1 周一早晨 10 点	182
9.1 本章内容	147	11.6.2 回到写字台	184
9.2 不同创建工具的需求	147	11.7 文档制作、原型设计和项目	
9.3 下载并安装 ant	148	干系人的支持认同	184
9.4 ant buildfile 示例	150	11.7.1 文档制作	184
9.4.1 XML	150	11.7.2 项目干系人的支持认同	185
9.4.2 buildfile 标签	151	11.7.3 原型制作	186
9.4.3 实际的 buildfile	156	11.8 本章小结	186
9.5 本章小结	158	11.9 内容扩展	187
9.6 内容扩展	159	11.10 参考文献	187
9.7 参考文献	159	11.11 练习	187
第 10 章 集成开发环境		第 12 章 分析与设计：寻找对象	
160		188	
10.1 本章内容	160	12.1 本章内容	188
10.2 NetBeans	161	12.2 起始工作	188
10.2.1 NetBeans 的发展史	161	12.3 使用 CRC 卡	188
10.2.2 安装 NetBeans	161	12.4 确定对象	189
10.2.3 熟悉 NetBeans	163	12.5 寻找方法和属性	190
10.2.4 与 CVS 集成	166	12.6 本质与非本质	191
10.2.5 集成 ant	168	12.7 分析停顿	193
10.2.6 其他附加工具	169	12.8 真实的软件工程	193
10.3 SunONE Studio 公共版	170	12.9 核心类	193
10.4 Eclipse	170	12.10 本章小结	197
10.4.1 选择和安装 Eclipse	170	12.11 内容扩展	197
10.4.2 在新项目里使用 Eclipse	172	12.12 参考文献	197
10.4.3 在已有的项目上使用		12.13 练习	197
Eclipse	174		
10.4.4 使用 Eclipse	174		
10.5 本章小结	174	第 13 章 JUnit：自动单元测试	
10.6 内容扩展	175	198	
		13.1 本章内容	198

13.2	JUnit	198	第 15 章	数据访问: JDBC	219
13.3	设计, 再测试, 再编码	198	15.1	本章内容	219
13.4	安装并运行 JUnit	200	15.2	JDBC 介绍	219
13.4.1	下载和解压缩	200	15.3	连接	222
13.4.2	使用 JUnit	202	15.4	查询数据	223
13.5	编写测试案例	203	15.5	查询结果	224
13.5.1	JUnit 论断	206	15.6	更新、插入和删除数据	225
13.5.2	运行测试案例	207	15.7	本章小结	225
13.6	运行一组测试程序	208	15.8	内容扩展	225
13.7	本章小结	209	15.9	参考文献	226
13.8	内容扩展	210	15.10	练习	226
13.9	参考文献	210	第III部分 开发 GUI		
13.10	练习	210	第 16 章	GUI 设计	229
第 14 章	数据存储	211	16.1	本章内容	229
14.1	本章内容	211	16.2	一个简单的 Swing 程序	229
14.2	数据库设计	211	16.3	Swing 示例	230
14.3	持久性	211	16.4	基本的 Swing 对象	231
14.4	独立性	211	16.5	布局管理器	233
14.5	Oracle、PostgreSQL 和 MySQL	212	16.6	行为、监听器与事件	233
14.5.1	MySQL	212	16.7	设计 GUI	234
14.5.2	PostgreSQL	212	16.7.1	概况	235
14.5.3	Oracle	212	16.7.2	创建块	236
14.5.4	选择标准	213	16.8	本章小结	252
14.6	在代码中设置数据库	213	16.9	内容扩展	252
14.7	持久性	214	16.10	参考文献	253
14.8	持久性并非设计数据库表 的唯一考虑	215	16.11	练习	253
14.9	设置 PostgreSQL	215	第 17 章	SWT	254
14.9.1	安装 PostgreSQL	215	17.1	本章内容	254
14.9.2	创建 postgres 用户	216	17.2	IBM SWT 工具包	254
14.9.3	创建数据库	216	17.2.1	另一种 GUI 工具包	254
14.9.4	JDBC	217	17.2.2	介绍 SWT 的原因?	255
14.10	本章小结	218	17.2.3	移植性	255
14.11	内容扩展	218	17.2.4	本章其他内容	255
14.12	参考文献	218	17.2.5	SWT: 贴近硬件	256
14.13	练习	218	17.2.6	“Hello, world” SWT 类型	256
			17.3	移植 BudgetPro 到 SWT	258

17.3.1	第 1 步: 转换类成员	260
17.3.2	第 2 步: 转换 main()方法	262
17.3.3	第 3 步: 转换 GUI build() 和 init()方法	262
17.3.4	完成 BudgetPro 类 的转换	265
17.3.5	完成应用程序的转换	265
17.3.6	SWT 和 Swing 的比较	265
17.4	SWT 和 gcj	266
17.5	本章小结	267
17.6	内容扩展	267
17.7	参考文献	267
17.8	练习	268

第IV部分 开发 Web 接口

第 18 章	Servlet	271
18.1	本章内容	271
18.2	Servlet	271
18.3	深入理解与 Servlet 的交互	272
18.4	如何编写 Servlet	274
18.5	输入与输出	277
18.6	状态问题: cookie、隐藏 变量以及“后退”按钮	278
18.7	设计 BudgetPro Servlet	280
18.7.1	原型	280
18.7.2	设计	280
18.8	本章小结	283
18.9	内容扩展	283
18.10	参考文献	283
18.11	练习	283

第 19 章	JSP	285
--------	-----	-----

19.1	本章内容	285
19.2	JSP	285
19.3	如何编写 JSP 应用程序	287
19.3.1	scriptlet	287
19.3.2	声明	289
19.3.3	表达式	289

19.3.4	指令	291
19.3.5	新语法	292
19.3.6	JSP 中的 JavaBeans	292
19.3.7	标记库	294
19.4	在 BudgetPro 应用程序 中使用 JSP	295
19.5	本章小结	296
19.6	内容扩展	296
19.7	参考文献	296
19.8	练习	297

第 20 章 开源代码的 Web 应用

程序服务器

20.1	本章内容	298
20.2	下载 JBoss	298
20.3	软件系统的管理	299
20.3.1	无需 root 权限安装软件	299
20.3.2	出色的粒度访问控制	302
20.4	安装 JBoss	302
20.5	开源应用程序服务器与系统 的集成	303
20.5.1	System V 的 init 系统	303
20.5.2	RedHat/Fedora 的 chkconfig 程序	308
20.5.3	其他发行版	312
20.5.4	IDE 集成	312
20.6	部署	312
20.7	Apache Geronimo	314
20.8	安装 Geronimo	314
20.9	运行 Geronimo 服务器	315
20.10	本章小结	315
20.11	内容扩展	316
20.12	参考文献	316

第 V 部分 开发企业级可 扩展软件

第 21 章 Enterprise JavaBeans

介绍	319
21.1 本章内容	319

21.2	EJB 扩展	319	第 23 章	部署 EJB	339
21.2.1	EJB 概念	319	23.1	本章内容	339
21.2.2	bean 类型	321	23.2	Enterprise 打包与部署	339
21.2.3	EJB 的内幕	324	23.2.1	EJB-JAR 文件的内容	341
21.3	JNDI 介绍	324	23.2.2	使用 bean	343
21.3.1	命名以及目录系统 的概念	324	23.2.3	打包 servlet	344
21.3.2	通用目录服务	325	23.3	部署 EAR	346
21.3.3	JNDI	327	23.3.1	JBoss	346
21.3.4	通过 JBoss 使用 JNDI	330	23.3.2	Geronimo	346
21.4	本章小结	330	23.4	维护分布式应用程序	346
21.5	内容扩展	330	23.4.1	And 与 CVS	347
21.6	参考文献	330	23.4.2	XDoclet	348
第 22 章	建立 EJB	331	23.5	抽象传统的应用程序	348
22.1	本章内容	331	23.6	本章小结	348
22.2	了解 bean	331	23.7	内容扩展	348
22.2.1	SessionBean	332	23.8	参考文献	349
22.2.2	EJBObject	334	第 24 章	临别赠言	350
22.2.3	EJBHome	335	24.1	未来如此光明, 但我有些 迷惑	350
22.2.4	会话 bean、远程接口 以及 home 接口小结	335	24.2	欢迎您的参与	350
22.2.5	EJBLocalHome 与 EJBLocalObject	335	24.3	到来的革命	350
22.2.6	编译 bean	336	24.4	内容扩展	351
22.3	本章小结	337	24.5	参考文献	351
22.4	内容扩展	338	附录 A	ASCII 图表	353
22.5	参考文献	338	附录 B	BudgetPro 的 Java Swing GUI	355
			附录 C	GNU 通用公共许可证	365

第 I 部分

基础知识

- Linux 环境
- 编辑器
- Java 介绍
- 执行语境
- Sun Microsystems Java 软件开发包
- Linux 下的 IBM 开发工具包, Java 2 技术版
- GNU 编译器
- CVS
- ant 简介
- 集成开发环境

