

开发专家
之 Sun ONE

JBoss

管理与开发核心技术 (第三版)

JBoss Administration
and Development
Third Edition



原厂技术手册



(美) Scott Stark, JBoss, Inc.
罗时飞
飞思科技产品研发中心

著
译
监制



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

随书光盘内容为
书中范例源代码



开发专家
之 Sun ONE

JBoss

管理与开发核心技术 (第三版)

JBoss Administration
and Development

Third Edition

(美) Scott Stark, JBoss, Inc. 著
罗时飞 译
飞思科技产品研发中心 监制

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书的主要目的是从配置和架构这两个角度弄清楚标准 JBoss 3.2.x 的架构组件。使用 JBoss 的开发者将理解如何配置标准的组件。其中,需要开发者注意的是,本书不是介绍 J2EE 和如何应用 J2EE 的,而是关注 JBoss 服务器架构的内部细节及具体 J2EE 容器实现的配置和扩展。

通过阅读本书,JBoss 开发者可以深入理解标准组件的架构、集成,以及扩展或替代标准组件,从而满足其基础信息框架需求。同时,本书也将指导开发者如何获得 JBoss 源码,并在此基础上构建和调试 JBoss 服务器。

本书最后提供了索引,以方便读者更加方便快速地找到相关技术点。随书光盘内容为书中范例源代码,希望对读者有所帮助。

Original English language edition copyright ©2004 by Scott Stark and JBoss,Inc.

Chinese language edition copyright ©2004 by Publishing House of Electronics Industry.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission in writing from the Proprietor.

本书原版英文版权归 Scott Stark 和 JBoss 公司所有,中文版权归电子工业出版社所有。未经版权所有者和出版者许可,任何单位和个人不得以任何方式复制或抄袭本书的部分或全部内容。

版权贸易合同登记号 图字:01-2004-5116

图书在版编目(CIP)数据

JBoss 管理与开发核心技术:第3版/(美)斯塔克(Stark,S.)著;罗时飞译. —北京:电子工业出版社,2004.11

(开发专家之 Sun ONE)

书名原文:JBoss Administration and Development Third Edition

ISBN 7-121-00393-7

I.J... II.①斯...②罗... III.①JAVA 语言—程序设计②因特网—网络服务器 IV.TP312

中国版本图书馆 CIP 数据核字(2004)第 097269 号

责任编辑:郭 晶

印 刷:北京东光印刷厂

出版发行:电子工业出版社

北京海淀区万寿路 173 信箱 邮编:100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:33.75 字数:864 千字

印 次:2004 年 11 月第 1 次印刷

印 数:5 000 册 定价:59.00 元(含光盘 1 张)

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 zlt@phei.com.cn,盗版侵权举报请发邮件至 dbqq@phei.com.cn。

出版说明

“开发专家”是电子工业出版社计算机研发部长期以来精心培育的计算机科学技术类本版品牌。这个品牌是由多个专题系列组成的横向大系列，涵盖了计算机技术的各个方面，特别是一直受到极大关注的程序开发类系列，例如“开发专家之数据库”、“开发专家之网络编程”、“开发专家之 Delphi”、“开发专家之 Sun ONE”、“开发专家之 Oracle”和“嵌入式开发专家”等。这些专题系列基于各自的角度，从纵向上包含了该专题的所有内容。因此，整个“开发专家”的品牌架构纵横交错，囊括了所有的计算机技术和所有的技术层面，海纳百川而又极具可扩展性。

“开发专家”的作者队伍主要依托于“飞思科技产品研发中心”。“飞思科技产品研发中心”由专业的策划人员、权威的技术专家和资深的作者队伍共同组成。在图书的出版上，形成了以研发为基础、以出版为中心、以服务为支持的专业化出版框架和流程。通过深入的市场调查和技术跟踪，在综合了技术需求和读者焦点等因素的基础上，形成各系列丛书的写作重点和大纲，然后聘请业界的最前沿学者进行写作。同时，策划工作全程介入写作进程，严格控制写作质量，用最专业的技术背景、最深刻的理论基础、最具代表性的案例、最能为专业读者接受的形式，为读者提供品质最佳的图书产品，体现了出版者和著作者的完美结合。

多年来，我们始终把创造社会效益摆在首位，秉承一切为国内计算机技术专业读者服务的精神，为推动国内信息技术的发展、为体现国内技术的原创水平，穷尽所有的创意与努力，将出版者的命运与读者的支持紧紧地连在了一起。

在此，我们临出版之残酷竞争而不惧，旌旗猎猎而异军突起，这与广大读者的支持是分不开的。为使我们的脚步更坚实，使我们的队伍永远保持活力和创造力，我们期待着您能为我们的前进贡献出您的意见和建议。同时，我们也在等待着您的加入。

电子工业出版社计算机研发部

飞思人理念

我们经常感谢生活的慷慨，让我们这些原本并不同源的人得以同本，为了同一个梦想走到一起。

因为身处科技教育前沿，我们深感任重道远；因为伴随知识更新节奏，我们一刻不敢停歇。虽然我们年轻，但我们拥有：

“严谨、高效、协作”的团队精神

全方位、立体化的服务意识

实力雄厚的作者群和开发队伍

当然，最重要的是我们拥有：

恒久不变的理想和永不枯竭的激情和灵感

正因如此，我们敢于宣称：

飞思科技=丰富的内容+完美的形式

这也是我们共同精心培育的品牌  的承诺。

“问渠哪得清如许，为有源头活水来”。路再远，终需用脚去量；风景再美，终需自然抚育。

年轻的飞思人愿为清风细雨、阳光晨露，滋润您发芽、成长；更甘当坚实的铺路石，为您铺就成功之路。

序

如果用户正在阅读本文，那我首先要感谢你们购买我们的产品。这也是用户支持 JBoss 的开发成果及保证 JBoss 能够继续生存和分发技术上最为先进的 Web 应用服务器的可能性的一种方式。当前，开源（开放源代码，Open Source）的革新正处于有趣的阶段。由于大量的开源项目出现后，人们并没有持续地对其给出专业级的关注，因此它们往往是昙花一现。JBoss 渴望通过商业化行为资助免费核心产品的开发，从而引领“专业的开源”潮流。

JBoss 模块（module）的增长很迅速。JMX 基础框架借助于 MBeanServer 作为所有单独模块的生命周期和管理的基本抽象，使得其能够将所有单独模块集成在一起。与此同时，JBoss 还有很优秀的开发者对 JBoss 服务器模块的各个状态和全面的快照进行专业化的文档工作。小到基础框架，大到用于安全性的 JBossSX 和 CMP 引擎的高级模块，开发者都能够找到充分的资料来解决各自的问题。另外，在用户熟悉和使用 JBoss 的过程中，我们会一直提供充分的资料，使得他们更好地理解各个模块，最终使得他们能够更好地理解和精通 JBoss。

JBoss 在技术上已经取得了卓越的成绩和广泛的赞赏。我们期待这种状况能够更持久。当然，不要误解我的意思。我们非常自豪，因为在过去三年多的时间中有许多很优秀的人聚集在 JBoss 周围，也希望有更多的人加入到我们的团队中来。这其中包括所有阅读本书的读者。JBoss 不仅仅是非常棒的应用服务器，而且它还是能够带来生机、新观点的社区。我们不只是为了获得更多的用户而努力，而且为用户提供更多的工具和必需的知识使得他们能够精通 JBoss 的产品而最终成为 JBoss 的贡献者。对于日常使用 JBoss 和产品开发而言，理解 JBoss 的配置和架构是必须的。另外，我们也希望大家能够领略到技术的快乐及开源所带来的全新体验。

无论是模块功能的理解，还是新工具、新功能，甚至新模块，我们都衷心地希望这本书能够尽可能地给更多的人提供上述帮助。一旦开发者达到了上述目标，请一定要联系我们，开发者一定会发现一个欣欣向荣的、为专业人员分享开发成果的、专业的社区。另外，在 www.jboss.org 还提供其他信息、论坛及最近的软件发布信息。

再一次感谢您购买 JBoss 的文档。JBoss 衷心地希望经常和用户取得联系。与此同时，请好好分享学习、工作所带来的乐趣。

马克·弗鲁里
JBoss 公司总裁

译者序

JBoss 是最优秀的、基于 Open Source 的 J2EE 应用服务器之一。一方面,业界对 JBoss 应用服务器强烈关注,因为 JBoss 具有精良的、基于 JMX 的总线架构,这使得它在开发者中更加流行和实用。另一方面,JBoss Group 本身积极参与 J2EE 相关规范的制定和相关技术的研究(比如,参与 EJB 3.0 规范制定,提供轻量级 O/R Mapping 产品 Hibernate,参与 Tomcat 开发,提供 JBoss AOP 及 JBossCache 等优秀产品),这使得 JBoss 应用服务器本身能够遵循和推动 J2EE 规范的革新。这些都使得 JBoss 更为健壮,更为持久,更为“Professional Open Source”,现在已经发展成为以 JBoss 平台为中心的 JBoss 社区。

Java 管理扩展(Java Management Extension, JMX)在 JBoss 架构中具有不可替代的作用。借助于 JMX,JBoss 应用服务器的一切都是模块化的。JBoss 3.2.x 系列产品,在 JBoss 2.x/3.0.x 的基础上,已经发展成为很完善、成熟的产品,而且其源代码也成为了 JBoss 4.x (J2EE 1.4 认证)的重要基础。这些产品系列都以 JMX 作为架构总线。同时,本书的第三版,即读者手中的这本书,正是深入介绍和研究 JBoss 3.2.x 系列产品的管理和开发的。这本书在 JBoss 社区中广泛传播,已经成为了 JBoss 管理和开发的重要基础资源。它将 JBoss 的各个层面娓娓道来,尤其是管理和开发 JBoss 应用服务器本身,使得开发者能够定制更适合自身业务需求的 JBoss 应用服务器;同时,也使开发者深入到 JBoss 应用服务器源码之中,而领略其带来的无限魅力及完美的体验。

本书由 JBoss Group 官方编写,绝对是目前对 JBoss 应用服务器诠释最好的参考书之一。它首先从介绍 JBoss 的安装和构建入手(第 1 章),然后介绍了 JBoss JMX 微内核(第 2 章),这也是本书最为重要、最为基础的内容之一。进而,本书结合 J2EE 平台技术和 JBoss 服务,依次展开对 JNDI 命名(第 3 章)、JTA 事务服务(第 4 章)、EJB 容器配置和架构(第 5 章)、JMS 配置和架构(第 6 章)、JCA 配置和架构(第 7 章)、J2EE 安全性配置和架构(第 8 章),以及 CMP 引擎(第 11 章)的深入研究和分析,从而奉献出了 JBoss 中蕴含的丰富内容和业界经验。另外,本书还阐述了 JBoss 其他内容,比如集成 Servlet 容器(第 9 章)、MBean 服务杂记(第 10 章)及 Web 服务(第 12 章)。还犹豫什么,赶紧进行 JBoss 之旅吧!

另外,在网站 <http://www.open-v.com> 和 www.fecit.cn 上,还提供了本书的最新信息及 JBoss 相关资料。

由于译者水平有限,加上时间仓促,书中翻译难免有不妥之处,希望广大读者和同行批评和指正。

译者 罗时飞
2004 年 8 月于广州

前 言

关于开源

开源背后所隐藏的基本哲理非常简单：当开发者能够阅读、重新发布并修改软件时，软件改进了。人们改进它、使用它并修复 Bugs。在通常情况下，如果开发者厌烦传统软件开发的慢速度，他们将会对开源产生极大的兴趣。开源和免费软件，这两个概念经常被大家误解。Open Source Initiative（简称 OSI）网站给出了很多从不同角度定义开源的参考资料。具体的网址在：

<http://www.opensource.org/docs/definition.html>

下面与 JBoss 关系极为密切的一段概述开源的话摘自 OSI 主页：

开源社区的人们体会到，其快速的革新速度能够比传统的封闭式开发模型创造更好的软件。在封闭式开发模型中，只有很少的几个人能够看到源代码，而其他人仅仅是盲目地使用二进制软件。

OSI 的出现使得这种开放式的开发在商业领域获得可能。

开源软件的时代已经到来。经过近二十年的发展，这种模式一直在建立 Internet 和 WWW 技术文化的过程中积蓄能量。现在，它已经冲入了商业领域，并改变了现有的所有规则。你们做好准备了吗？

关于 JBoss

JBoss，一个领先的 Java 开源、集成和开发基于 J2EE 的完整服务实现。JBoss 提供 JBossServer、基本的 EJB 容器及 JMX 框架。它也为 JMS 系统提供 JBossMQ，为 JTA 事务提供 JBossTX，为 CMP 持久化提供 JBossCMP，为基于 JAAS 的安全性提供 JBossSX，以及为 JCA 连接提供 JBossCX。为支持 Web 组件，比如 Servlet 和 JSP，JBoss 提供了抽象集成层。而这些抽象层的集成服务实现可以由第三方 Servlet 引擎提供，比如 Tomcat 和 Jetty。JBoss 使得开发者能够通过 JMX 混合应用这些组件，即借助于替换 JMX 兼容的组件实现完成这项任务。同时，这些 JBoss 组件之间甚至不会产生任何影响。因此，JBoss 现在全部都是模块化的。

JBoss 是基于 JMX 的完整 J2EE 实现。JBoss 的目标是提供完整的开源 J2EE 服务栈，其实我们已经达到了这个目标，而成功的秘诀在于 JMX。JMX，即 Java 管理扩展，是我们发现集成软件的最好利器。JMX 提供了公共的平台以允许集成模块、容器及插件。图 1 展示了 JMX 总线是如何和 JBoss 架构进行交互的。

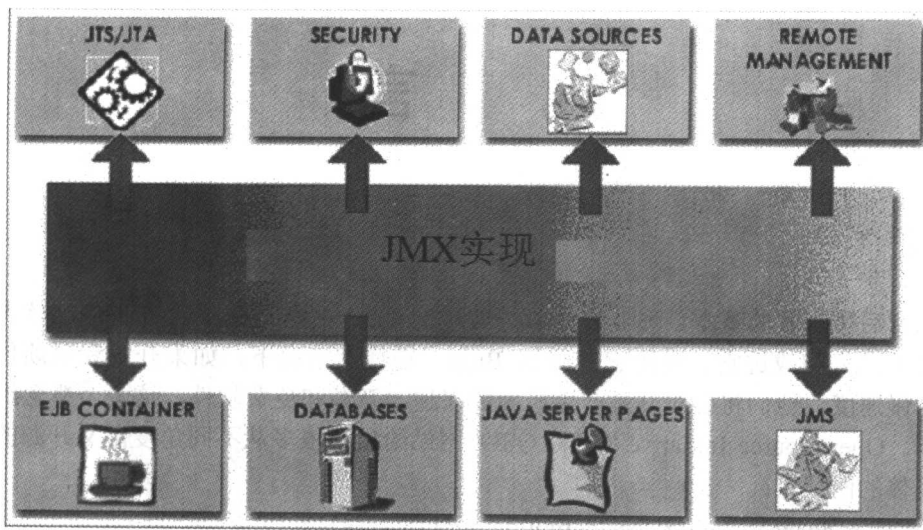


图1 JBoss JMX 集成总线及标准的 JBossXX 组件

关于作者

斯科特·斯塔克 (Scott Stark)，博士，1964 年生于美国的华盛顿州。他原先是一名化学工程师，后来于华盛顿大学获得硕士学位，于特拉华大学获得博士学位。在攻读博士学位期间，他对计算机和编程已经流露出浓厚的兴趣。因此，使用功能强大的并行计算机来解决难度极大的化学工程问题便成了他读博士期间的研究方向。至今，分布式编程一直是他的主攻目标。当前，Scott 是 JBoss 公司的首席技术官。

马克·弗鲁里 (Marc Fleury)，博士，1968 年生于巴黎。马克原先在太阳微视（法国）从事过销售工作。毕业于法国声名显赫的 Ecole Poly 技术学院的他，同时也是伞兵队的前任上尉。后来，马克从巴黎高等师范学校获得理论物理专业的硕士和博士学位。在从事物理方面的研究过程中，他也在麻省理工学院做过 X 射线激光器方面的访问学者。目前，马克是 JBoss 公司，即一个坐落在亚特兰大郊外的、提供优质服务的公司的总裁。

JBoss 公司，由 Marc Fleury 领衔的、由世界范围内 100 多位开发者组成的公司。他们分发各种用途的 J2EE 工具，并且致力于将 JBoss 发展为 J2EE 平台中的主流企业 Java 应用服务器。

JBoss 是开源支持、兼容 J2EE 标准及纯 Java 实现的 J2EE 应用服务器。JBoss 服务器及其附属产品都是基于开放的授权分发的。每月至少达到 100 000 次的下载量，使得 JBoss 成为业界下载最多的 J2EE 基础服务器。

本书内容

本书的主要目的是从配置和架构这两个角度弄清楚标准 JBoss 3.2.x 的架构组件。使用 JBoss 的开发者将理解如何配置标准的组件。其中，需要开发者注意的是，本书不是介绍

J2EE 和如何应用 J2EE 的，而是关注 JBoss 服务器架构的内部细节及具体 J2EE 容器实现的配置和扩展。

JBoss 开发者将很好地理解标准组件的架构、集成，以及扩展或替代标准组件，从而满足用户的基础信息框架需求。同时，本书也将指导开发者如何获得 JBoss 源码，并在此基础上构建和调试 JBoss 服务器。

本书最后提供了索引表，以方便读者更加方便快速地找到相关技术点。随书光盘内容为书中范例源代码，希望对读者有所帮助。

本书由飞思科技产品研发中心策划引进并组织翻译，由具有多年 J2EE 开发、培训及写作经验的罗时飞先生执笔翻译。罗先生在翻译过程中数易其稿，其精益求精的治学态度令我们深深感动。此书在出版过程中也得到了 JBoss 公司 Ben Wang（王文彬）博士的大力支持，在此一并表示感谢。由于翻译时间仓促，书中难免会有一些疏漏，希望广大读者给予批评指正。

我们的联系方式如下：

咨询电话：(010) 68134545 68131648

答疑邮件：support@fecit.com.cn

服务网址：<http://www.fecit.com.cn> <http://www.fecit.net>

通用网址：计算机图书、FECIT、飞思教育、飞思科技、飞思

飞思科技产品研发中心

目 录

第 1 章 安装和构建 JBoss 服务器	1
1.1 获得二进制文件	1
1.1.1 预备条件	1
1.1.2 安装二进制存档	2
1.1.3 default 服务器配置文件集合	3
1.2 基本安装测试	8
1.3 从网络服务器启动	10
1.4 基于源代码构建服务器	14
第 2 章 JBoss JMX 微内核 (Microkernel)	21
2.1 JMX 介绍	21
2.1.1 装配层	22
2.1.2 代理层	23
2.1.3 分布式服务层	23
2.1.4 JMX 组件介绍	23
2.2 JBoss JMX 实现架构	26
2.2.1 JBoss 类装载机架构	26
2.2.2 类装载和 Java 中的类型	26
2.2.3 JBoss XMBean	47
2.3 连接到 JMX 服务器	53
2.3.1 浏览服务器——JMX 控制台 Web 应用	53
2.3.2 使用 RMI 连接到 JMX	56
2.3.3 命令行方式访问 JMX	66
2.3.4 使用任何协议连接到 JMX	71
2.4 将 JMX 作为微内核	71
2.4.1 启动过程	71
2.4.2 JBoss MBean 服务	72
2.4.3 开发 JBoss MBean 服务	81
2.4.4 部署排序和依赖性	110
2.5 JBoss 部署器架构	127
2.6 借助于 SNMP 展示 MBean 事件	131
2.6.1 SNMP 适配器服务	131
2.6.2 陷阱服务事件	132
2.7 远程访问服务——分离式 Invoker	132
2.7.1 分离式 Invoker 实例: MBeanServer Invoker 适配器服务	137
2.7.2 分离式 Invoker 参考	143

第 3 章 JBoss 之命名——JNDI 命名服务	149
3.1 JNDI 概述	149
3.1.1 JNDI 应用编程接口	149
3.1.2 J2EE 和 JNDI——应用组件环境	151
3.2 JBossNS 架构	164
3.2.1 命名 InitialContext 工厂	166
3.2.2 基于 HTTP 访问 JNDI	170
3.2.3 保护基于 HTTP 访问 JNDI	176
3.2.4 保护只读、未保护上下文 JNDI 的访问	178
3.2.5 其他命名 MBean	179
第 4 章 JBoss 之事务——JTA 事务服务	185
4.1 事务/JTA 概述	185
4.1.1 悲观锁/乐观锁	186
4.1.2 分布式事务的组件	186
4.1.3 两阶段 XA 协议	187
4.1.4 启发式异常	187
4.1.5 事务 ID 和分支	187
4.2 JBoss 事务内核	188
4.2.1 为 JBoss 适配事务管理器	188
4.2.2 默认事务管理器	189
4.2.3 UserTransaction 支持	189
第 5 章 JBoss 之 EJB——EJB 容器配置和架构	191
5.1 EJB 客户端视图	191
5.2 EJB 服务器端视图	198
5.3 EJB 容器	204
5.3.1 EJBDeployer MBean	204
5.3.2 容器插件式框架	215
5.4 实体 Bean 锁和死锁检测	227
5.4.1 JBoss 为什么需要锁	227
5.4.2 实体 Bean 的生命周期	228
5.4.3 默认锁行为	228
5.4.4 插入式拦截器和锁策略	229
5.4.5 死锁	230
5.4.6 高级配置和调优	232
5.4.7 在群集中运行	236
5.4.8 修理故障	236
第 6 章 JBoss 之消息——JMS 配置和架构	239
6.1 JMS 实例	239
6.1.1 点对点实例	240

6.1.2	发布/订阅实例.....	242
6.1.3	使用持久 topic 的发布/订阅实例.....	249
6.1.4	使用 MDB 的点对点实例.....	252
6.2	JBoss 消息概述.....	261
6.2.1	调用层.....	261
6.2.2	安全性管理器.....	262
6.2.3	目的地管理器.....	263
6.2.4	消息缓存.....	263
6.2.5	状态管理器.....	263
6.2.6	持久化管理器.....	263
6.2.7	目的地.....	264
6.3	JBoss 消息配置和 MBean	264
6.3.1	org.jboss.mq.il.jvm.JVMServerILService	265
6.3.2	org.jboss.mq.il.rmi.RMIServerILService (已丢弃)	265
6.3.3	org.jboss.mq.il.oil.OILServerILService (已丢弃)	266
6.3.4	org.jboss.mq.il.uil.UILServerILService (已丢弃)	266
6.3.5	org.jboss.mq.il.uil2.UILServerILService	267
6.3.6	org.jboss.mq.il.http.HTTPServerILService	269
6.3.7	org.jboss.mq.server.jmx.Invoker	270
6.3.8	org.jboss.mq.server.jmx.InterceptorLoader	270
6.3.9	org.jboss.mq.sm.file.DynamicStateManager	270
6.3.10	org.jboss.mq.security.SecurityManager	271
6.3.11	org.jboss.mq.server.jmx.DestinationManager	273
6.3.12	org.jboss.mq.server.MessageCache	274
6.3.13	org.jboss.mq.pm.file.CacheStore.....	274
6.3.14	org.jboss.mq.pm.file.PersistenceManager	274
6.3.15	org.jboss.mq.pm.rollinglogged.PersistenceManager	275
6.3.16	org.jboss.mq.pm.jdbc2.PersistenceManager.....	275
6.3.17	目的地 MBean.....	276
6.3.18	借助于 JMX 管理.....	279
6.4	指定 MDB JMS 供应商.....	280
6.4.1	org.jboss.jms.jndi.JMSProviderLoader MBean.....	281
6.4.2	org.jboss.jms.asf.ServerSessionPoolLoader MBean	282
6.4.3	集成其他 JMS 供应商.....	283
第 7 章	JBoss 之连接器——JCA 配置和架构	285
7.1	JCA 概述	285
7.2	JBossCX 架构概述	287
7.2.1	BaseConnectionManager2 MBean.....	288
7.2.2	RARDeployment MBean	289
7.2.3	JBossManagedConnectionPool MBean	290

7.2.4	CachedConnectionManager MBean.....	291
7.2.5	JCA 资源适配器实例纲要.....	291
7.3	配置 JCA 适配器.....	298
7.3.1	配置 JDBC 数据源.....	299
7.3.2	配置常见 JCA 适配器.....	303
7.3.3	配置实例.....	306
第 8 章	JBoss 之安全性 ——J2EE 安全性配置和架构.....	307
8.1	J2EE 安全性声明概述.....	307
8.1.1	安全性引用.....	309
8.1.2	安全性身份.....	310
8.1.3	安全性角色.....	311
8.1.4	EJB 方法许可.....	312
8.1.5	Web 内容安全性约束.....	315
8.1.6	使用 JBoss 中的安全性声明.....	316
8.2	JAAS 介绍.....	316
8.3	JBoss 安全性模型.....	321
8.4	JBoss 安全性扩展架构.....	328
8.4.1	JaasSecurityManager 如何使用 JAAS	329
8.4.2	JaasSecurityManagerService MBean	331
8.4.3	扩展 JaasSecurityManager, JaasSecurityDomain MBean.....	333
8.4.4	基于 XML 的 JAAS 登录配置 MBean.....	334
8.4.5	JAAS 登录配置管理 MBean.....	336
8.4.6	使用和开发 JBossSX 登录模块.....	336
8.4.7	开发自定义登录模块.....	347
8.5	安全远程密码协议.....	357
8.5.1	为 SRP 提供密码信息.....	360
8.5.2	深入 SRP 算法.....	362
8.6	使用 Java 2 安全性管理器运行 JBoss.....	367
8.7	使用 JSSE 为 JBoss 提供 SSL.....	369
8.8	配置用于防火墙后的 JBoss.....	376
8.9	如何保护 JBoss 服务器.....	377
8.9.1	jmx-console.war.....	377
8.9.2	web-console.war.....	377
8.9.3	http-invoker.sar	377
8.9.4	jmx-invoker-adaptor-server.sar	377
第 9 章	集成 Servlet 容器.....	379
9.1	AbstractWebContainer 类.....	379
9.1.1	AbstractWebContainer 契约.....	380
9.1.2	创建 AbstractWebContainer 子类.....	386

9.2	JBoss/Tomcat-4.1.x 绑定	388
9.2.1	嵌入式 Tomcat 配置元素	389
9.2.2	JBoss/Tomcat 绑定使用 SSL	394
9.2.3	为 JBoss/Tomcat-4.x 绑定配置虚拟主机	401
9.2.4	使用外部静态内容	403
9.2.5	为 JBoss/Tomcat-4.x 绑定使用 Apache	405
9.2.6	使用群集	407
第 10 章	MBean 服务杂记	409
10.1	系统属性管理	409
10.2	属性编辑器管理	410
10.3	服务绑定管理	410
10.4	定时任务	417
10.5	JBoss 日志功能框架	420
10.6	RMI 动态类装载	421
第 11 章	CMP 引擎	423
11.1	启程	423
11.2	jbossCMP-jdbc 结构	430
11.3	实体 Bean	431
11.4	容器管理持久域	437
11.4.1	容器管理持久域抽象访问方法	437
11.4.2	容器管理持久域声明	438
11.4.3	容器管理持久域列映射	438
11.4.4	read-only 域	440
11.4.5	评审实体 Bean 访问	440
11.4.6	依赖值类	442
11.5	容器管理关系	446
11.5.1	cmr-field 抽象访问方法	446
11.5.2	关系声明	447
11.5.3	关系映射	448
11.6	查询	454
11.6.1	finder 和 ejbSelect 声明	455
11.6.2	EJB-QL 声明	455
11.6.3	覆盖 EJB-QL 到 SQL 的映射	457
11.6.4	JBossQL	457
11.7	优化装载	464
11.7.1	装载场景	465
11.7.2	装载组	466
11.7.3	read-ahead	467
11.7.4	装载过程	471

11.7.5 事务.....	477
11.8 乐观锁	480
11.9 实体命令和主键生成	485
11.10 Defaults.....	488
11.11 自定义数据源.....	491
11.11.1 函数映射	493
11.11.2 类型映射	493
11.11.3 用户类型映射	494
第 12 章 Web 服务	495
12.1 XDoclet.....	495
12.2 将 Hello World EJB 发布为 Web 服务	496
附录 A JBoss Group 和我们的 LGPL 授权	507
附录 B JBoss DTDs	518
附录 C 实例安装	519
附录 D 索引.....	520

第 1 章 安装和构建 JBoss 服务器

JBoss，免费的、兼容于 J2EE 标准的应用服务器，是目前市场上应用最广泛的开源产品。JBoss 灵活高效、易于使用的服务架构，使得它成为初学者在学习 J2EE 过程中首选的服务器平台。甚至，高级架构师也使用 JBoss 来定制其中间件平台。另外，JBoss 发布版还集成了 Servlet 容器，即 Tomcat 或者 Jetty。每当新版 JBoss 发布时，通过 SourceForge 网站（<http://sourceforge.net/projects/jboss>），开发者还可以找到相应的源代码，这使得调试 JBoss 服务器成为可能。另外，开发者还可以学习到 JBoss 的内部工作机理，并构建适合自己（或商业用途）的定制版本。

本章通过具体实例阐述如何安装和配置 JBoss 3.2.x 服务器。总而言之，开发者将学到如下几方面的知识：

- 从 JBoss SourceForge 项目网站获得持续更新的二进制版本
- 安装二进制版本
- 测试安装

还包括如下知识：

- 安装目录结构
- 管理员用于设置 JBoss 的若干主要配置文件
- 如何从 SourceForge CVS 库获得 3.2.x 发布版的源代码
- 如何构建 JBoss 服务器发布版

1.1 获得二进制文件

从 SourceForge JBoss 项目的“Files”页面可以获得最新的 JBoss 发布版，具体的网址位于：

<http://sourceforge.net/projects/jboss>

同时，开发者还将找到 JBoss 的历史、beta 版及 RC 版本。

1.1.1 预备条件

开发者在安装和运行 JBoss 服务器之前，必须保证目标系统安装了 JDK 1.3 以上（包括 JDK 1.3）版本。执行 `java -version` 命令除能验证上述需求是否满足外，还能确认 Java 命令工具是否位于 `path` 环境变量之中。比如，在提供 Sun 1.3.1 JDK 的 Linux 系统中运行上述命令，将获得如下结果：

```
/tmp 1206>java -version
java version "1.3.1_03"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.3.1_03-b03)
```