

历时一载 倾力打造 深度经验 汇聚此书

精通 Spring 企业应用 开发详解

陈雄华 著

光盘包含：
本书所有实例代码



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



精通 Spring[®]

—企业应用开发详解

陈雄华 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书深刻揭示了 Spring 的技术内幕,对 IoC、AOP、事务管理等根基性的技术进行了深度的挖掘。读者阅读本书后,不但可以熟练使用 Spring 的各项功能,对各种 Spring 内部机制也将了然于胸,真正做到知其然知其所以然。此外,我们还对 Spring 技术所涉及到的各种 Java 技术、Java 设计模式进行了适时的介绍,通过这些背景知识的准备,读者在理解 Spring 的各项原理时将不会有任何的障碍。

在内容的剪裁上,我们力争做到实用性和深入性的统一。Spring 技术涵盖面宽广,内容全面而丰富。我们经过谨慎分析,遴选出满足大多数企业应用的技术进行重点的分析和深入的讲解,忽略掉那些比较偏门的技术。这样我们就很好地达到了主次分明、重点突出的目的,而没有陷入“什么都讲,什么都沒有讲透”的窠臼之中。

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

精通 Spring 2.x: 企业应用开发详解 / 陈雄华著. —北京: 电子工业出版社, 2007.9
(Java 技术大系)
ISBN 978-7-121-04793-0

I. 精… II. 陈… III. JAVA 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 116516 号

责任编辑: 朱沐红 江 立

印 刷: 北京市通州大中印刷厂

装 订: 三河市金马印装有限公司

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

开 本: 787×1092 1/16 印张: 50.5 字数: 1210 千字

印 次: 2007 年 9 月第 1 次印刷

印 数: 5000 册 定价: 79.00 元(含光盘 1 张)

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

关于 Spring

Spring 为 Java 世界带来了一场震撼性的技术革新，它颠覆了传统 Java 应用开发的方式，影响和正在影响着 Java 开发者思考问题的方法。从 2004 年发布第一个版本以来，Spring 逐渐占据了 Java 开发人员的视线，博得了开源社区一片赞誉之声，开源社区里“春”城无处不飞花。

Spring 以反向控制和 AOP 为框架和核心，统一了应用对象的配置、查找、生命周期管理等工作，清晰地划分出业务逻辑与基础服务两者不同的关注面，开发人员可以使用简单 Java 对象轻松拥有 EJB 一样强大的功能。

Spring 是一站式的开发框架，它通过自身实现和第三者集成两种途径提供了 Java 企业应用展现层、业务层、持久层等各项技术。

在业务层里，Spring 提供了一个相当完善的解决方案，这包括声明式事务、远程服务、JMS、邮件发送、任务调度多种企业服务。

在展现层里，Spring 自身提供了一个出色的 Spring MVC 框架，该框架和流行的其他 MVC 框架相比毫不逊色。此外，Spring 还集成了各种流行的 Web 框架，这包括 Struts、WebWork、JSF、Tapestry 等。

在持久层里，Spring 自身提供了 Spring JDBC，它大大降低了 JDBC API 的使用，此外还提供了对 JPA、Hibernate、iBatis、JDO、Toplink 等持久化技术框架的集成。此外，Spring 还提供了一个位于具体持久化技术之上的 DAO 封装层，借助这一封装层，可以制定出和具体的底层实现技术无关的 DAO 接口。

从 2004 年以来，Spring 经历了翻天覆地的变化，承载了众多新功能的 Spring 2.0 在历时一年持续开发后，于 2006 年 10 月正式发布。在此之后，Spring 新功能的添加和旧功能的调整一直没有间断过，2007 年 5 月，Spring 2.1 版本发布。从 Spring 令人目不暇接的升级更新中，我们可以感受到 Spring 开发团队以及 Spring 开源社区的活跃。

本书的特点

- **揭示内幕、深入浅出：**笔者对 Spring 的源码进行了彻底的分析，深刻揭示了 Spring 框架的技术内幕，让读者知其然，更知其所以然。Spring 中许多设计经验、技巧、模式具有很高的借鉴性，在透彻学习 Spring 体系结构的同时，读者可以直接将这些方法引借到具体的程序开发中。
- **同步更新、与时俱进：**虽然在 2006 年初就发布了 Spring 2.0 的里程碑版本，后来又发布了多个 RC 版本（候选版本），并最终于 2006 年 10 月发布了 Spring 2.0 的正式

版本。在正式版本发布后，Spring 又以 Spring 2.0.x 的形式陆续发布了 5 个小版本，在 2007 年 5 月发布了 Spring 2.1 正式版本。新功能的添加以及旧功能的调整从来就没有停止过。笔者在本书写作过程中时刻关注 Spring 新版本功能的变化，并及时调整全书内容与其同步，目前全书内容保持和 Spring 2.1 版本同步。此外，Hibernate 已经更新到 3.2 版本，Struts 更新到 2.0 版本，XFire、JPA 已经成为新兴的受欢迎技术，本书在集成第三方技术时也 and 最新版本进行同步。

- **突出重点，淡化边缘：**虽然全书洋洋洒洒 700 余页，我们没有片面追求内容的面面俱到，相反，我们特别注意内容的剪裁和取舍。对于实用性强的知识点深入分析、深度挖掘，而对于不常使用的知识点点到为止，甚至不纳入本书的范围。举例来说，我们对使用 XFire 开发基于 Spring 的 Web Service 应用、Acegi 安全控制框架以及在 Spring 中使用 JPA 这些实用性强的技术都进行了深入的分析，而对如何集成 EJB、JMX、JCA 等这些不常使用的功能完全不涉及。很好地做到了实用性和深入性二者的统一。
- **理论透彻、面向实践：**本书在透彻分析原理、讲解技术知识点的同时，特别注意与实际应用的结合，笔者将自身丰富的实践经验糅合到全书的相关知识点上，很好地做到知识讲解和实践经验的结合。让读者在掌握纯技术知识的同时，能够对如何活用技术做到胸有成竹。如笔者在第 13 章讲解任务调度的内容时，专门辟出 13.6 小节讲解实际应用中任务调度的使用经验；在第 14 章中讲述使用 JavaMail 发送电子邮件时，专门通过第 14.4 小节讲述了在实际应用中发送邮件的各种注意事项。此外，我们还适时提供了“实战经验”的插文，它们在不影响上下文连贯性的同时让读者学习到相关知识的实战经验。诸如此类以实际应用为导向的内容贯穿全书，这是本书区别于其他书籍特色之一。
- **代码简洁、图例丰富：**全书的代码在排版布局以及内容的剪裁上颇费心思，实例代码重点关注当前知识点涉及的内容，弱化边缘代码，并采用特殊的排版方式适时添加简明扼要的注释，方便程序代码的阅读和重点内容的把握。全书拥有大量精美的图表，这些图表很好地解构了上下文中一些难点的知识，大大提高了阅读性，降低了理解的难度。
- **注重趣味、轻松阅读：**由于技术书籍的严谨性、知识性的特点，阅读技术书籍往往是枯燥乏味的，更遑论趣味性。笔者对此深有感触，为寻求一些突破，我们在全书大部分章节都精心设计一个“轻松一刻”，它们和上下文内容存在某种程度的关联性，但其本身是一段趣味性的短文，它们在增强全书趣味性的同时，还为读者提供另外一个思考问题的角度。
- **相关知识、一网打尽：**Spring 不但本身涉及众多 Java 技术，其集成的第三方技术本身也涵盖了丰富的知识。我们在介绍 Spring 相关技术时，都会简明扼要地讲解相关的基础知识，这包括 JDK 5.0 的新知识和被集成技术的知识，准备好知识背景，

而不是完全脱离背景知识的情况下孤立讲解 Spring 的知识。

- **历时一载、倾力打造：**本书从筹划到全书成稿，历经近一年的时间，笔者为了能够全力投入本书的撰写工作中，辞掉了工作，闭门谢客，专职写作。

本书的结构

本书分为 6 大部分，其中第一部分为 Spring 概述性知识；第二部分讲解了 Spring 框架核心技术的内容；第三部分涵盖了在 Spring 中使用各种数据访问技术的内容；第四部分讲解业务层各种技术的知识；第五部分是展现层技术的知识；第六部分讲解了如何测试 Spring 应用和 Spring 各种工具类的知识。其中有些章节的相似内容已经在技术网站 IT168 技术频道发表，网友和读者给予了充分的评价。最终成书的内容，又在网友评论和反馈的基础上做了补充和修订。下面简要介绍一下每章的内容。

第 1 章：对 Spring 框架进行宏观性的概述，力图使读者建立起对 Spring 整体性的认识。

第 2 章：通过一个简单的例子展现开发 Spring Web 应用的整体过程，通过这个实例，读者可以快速跨入 Spring Web 应用的世界。

第 3 章：讲解 Spring IoC 容器的知识，通过具体的实例详细地讲解 IoC 概念。同时，对 Spring 框架的三个最重要的框架级接口进行了剖析，并对 Bean 的生命周期进行讲解。

第 4 章：讲解了如何在 Spring 配置文件中使 Spring 2.0 的 Schema 格式配置 Bean 的内容，并对各个配置项的意义进行了深入的讲解。

第 5 章：对 Spring 容器进行解构，从内部探究 Spring 容器的体系结构和运行流程。此外，我们还将对 Spring 容器一些高级主题进行深入的阐述。

第 6 章：我们从 Spring AOP 的底层实现技术入手，一步步深入到 Spring AOP 的内核中，分析它的底层结构和具体实现。

第 7 章：对如何使用 Spring 2.0 新增的基于 AspectJ 配置 AOP 的知识进行了深入的分析，这包括使用 XML Schema 配置文件、使用注解进行配置等内容。

第 8 章：介绍了 Spring 所提供的 DAO 封装层，这包括 Spring DAO 的异常体系、数据访问模板等内容。

第 9 章：介绍了 Spring 事务管理的工作机制，通过 XML、注解等方式进行事务管理配置，同时还讲解了 JTA 事务配置知识。

第 10 章：讲解了如何使用 Spring JDBC 进行数据访问操作，我们还重点讲述了 LOB 字段处理、主键产生和获取等难点知识。

第 11 章：JPA 规范是 Java EE 5.0 标准的数据访问技术，本章对如何在 Spring 中使用 JPA 进行了全面的讲解。

第 12 章：讲解了如何在 Spring 中集成 Hibernate、iBatis 等数据访问框架，同时，读者还将学习到 ORM 框架的混用和 DAO 层设计的知识。

第 13 章: 本章重点对在 Spring 中如何使用 Quartz 进行任务调度进行了讲解, 同时还涉及了使用 JDK Timer 和 JDK 5.0 执行器的知识。

第 14 章: 本章讲解了如何在 Spring 中使用 JavaMail 发送电子邮件的内容, 同时还对 JavaMail 的技术体系进行了介绍。

第 15 章: 在 Java EE 应用中, JMS 是经常使用的技术之一, 本章讲解了如何在 Spring 中使用 JMS 的相关知识。

第 16 章: XFire 迅速成为 Web Service 开发框架中的闪亮新星, 相比于其他 Web Service 框架 (如 Axis、GLUE 等), XFire 拥有更多的优势。本章使用全新的视角讲述了在 Spring 中开发 Web Service 应用的各项知识。

第 17 章: Acegi 提供了系统安全的整体解决方案。Acegi 是基于 Spring Web 应用的安全框架, 在 Spring 中实施 Acegi 简单易行。本章讲解了如何在 Spring Web 应用中实施 Acegi 的整体过程。

第 18 章: 随着 Ruby, 特别是 Ruby on Rail 在应用领域所取得的成功, 近两年来动态语言成为炙手可热的技术之一。本章讲解了在 Spring 中集成 JRuby、Groovy 和 BeanShell 动态语言的过程。

第 19 章: Spring 为展现层提供了一个优秀的 Spring MVC 框架, 本章涵盖 Spring MVC 体系框架、处理器映射、控制器、文件上传、本地化、主题、异常处理等知识。

第 20 章: Spring MVC 的优秀之处在于它把视图层技术与 MVC 框架的其他部分分离开来, 用户可以在控制层代码完全不变的情况下改变视图的类型。本章重点讲述了 Spring MVC 各种视图的使用过程。

第 21 章: 虽然 Spring MVC 是一个优秀的 MVC 框架, 但或许读者更喜欢使用其他诸如 Struts、Tapestry、JSF 等框架。在本章中, 我们讲解了三个流行 Web 框架的最新版本如何集成到 Spring 中的具体步骤。

第 22 章: 程序测试对保障应用程序正确性而言, 其重要性怎样强调都不为过。本章虽然也涉及 JUnit 和 EasyMock 的知识, 但我们重点讲述如何使用 Spring 测试帮助类对 Spring 应用进行测试等方面知识。

第 23 章: Spring 不但提供了一个功能全面的应用开发框架, 本身还拥有众多可以在程序编写时直接使用的工具类。在本章中, 我们从众多的 Spring 工具类中遴选出最具适用性的部分介绍给大家。

如何使用本书

读者应该在机器上安装 Eclipse 3.2, 并下载 Spring 2.1 的最新发布包, 在机器上重现书中实例的开发过程。毕竟程序开发是实践性极强的工作, 只有亲身体验才能掌握其真谛。

配套光盘拥有本书所有实例的代码, 读者也可以在此基础上重复本书的实例的开发过程, 省去重新录入代码之苦。

本书的插文

本书会适时加入一些提示、实战经验和轻松一刻的小段插文，在不打断行文的同时提供一些有益的开发经验、使用技巧并增强阅读的趣味性。这些插文都带有一个小图标加以突显，说明如下：

	提示：在上下文中可能存在一些读者容易忽视或容易犯错的地方，在提示信息中给予针对性的帮助信息
	实战经验：笔者将多年的开发实战经验适时介绍给大家。这些知识往往是不能从一般的书籍或资料中获得的。本书会适时地在行文中将这些实战经验分享出来，相信可以使读者朋友少走一些弯路
	轻松一刻：为了增强技术书籍阅读的趣味性，全书每章几乎都有一到两个轻松一刻的短文，它们和上下文内容都存在某种程度的关联性，不但为阅读带来了趣味性，还可以启发读者的思考

此外，由于 Spring 2.x 拥有多个版本，为了保持行文的简洁，除非特别指出，本书的 Spring 或 Spring 2.0 即代表当前最新的版本（Spring 2.1.x）。

如何与作者联系

由于 Spring 内容涵盖面宽广，涉及的内容非常多，同时由于作者水平有限，错误之处在所难免。我们不但欢迎读者朋友来信交流，更期待各界高手、专家就不足之处给予赐教和斧正。您可以通过 quickSelect@yahoo.com.cn 与笔者联系。

陈雄华 厦门

致 谢

首先，我要感谢我的妻子林婕，最让我感动且愧疚的是在漫长而紧张的写作过程中，她在已有身孕的情况下还一如既往地承担起料理家务的重任，对于我这位没有尽到责任的丈夫，她不但毫无怨言，而且始终给予理解和鼓励，并且还帮助我查阅稿件。在本书创作期间，妻子为我生了一个聪明可爱的儿子，我想这是对我所有努力的最大奖赏，母子俩是我工作的风帆、生活的甘泉。

在儿子出生之后，父亲母亲不远千里，远道而来不辞辛苦地帮忙照顾妻子和儿子，岳父岳母更承担起照料妻子和儿子的重任。没有他们无尽的爱和帮助，我无法完成本书的撰写工作。

笔者在 2005 出版了拙作《精通 JBuilder 2005》，从此和电子工业出版社博文视点结下了不解的因缘。首先要感谢郭立主任，她总是那样富有耐心且善解人意，非常庆幸能够有这样一位出色的出版社领导合作。感谢出版社的张毅，几次北京之行他都热情地招待了我并给我提出了很多很好的建议。感谢上一本书的责任编辑毕宁，在本书前期的策划过程中，毕宁给予了诸多富有建设性的意见并在各项事宜上热情相助。最后更要感谢本书的责任编辑朱沐红和江立，她们认真负责、高效高质的工作让我心存敬意，很难想象还能找到比他们更好更专业的编辑队伍，感谢美编张子健和谢丹丹为本书设计了一个优美的封面。

本书在写作过程中得到了众多朋友及同仁的帮助。IT168 的熊建国老师和赵恒编辑在本书创作过程提供了诸多的帮助，他们的友谊是我永远的财富。同时，还要感谢我的几位志同道合的伙伴，他们是陈益明、刘剑锋、何鑫。在孤寂的写作过程中，他们的友谊显得弥足珍贵。陈益明的幽默给我们的生活带来了欢声笑语，他所讲的蜈蚣和蚂蚁的笑话是其中的经典之作；刘剑锋是位出色的美工设计师，他曾经帮助我设计了一个封面初稿，这是一个颇具创意的封面，后来虽然因为各方面的原因没有采用，但这份热情和友谊让我心存感激；何鑫不但在计算机领域技术精湛，勇夺第一届 AJAX 全国大赛的冠军，更绝的是他居然还能做出一手可口的好菜。此外，需要感谢的朋友还很多，他们分别是郑珺、李美莹、杨成月、庄玉林、耿渊、涂传滨、陈尚志、邱德顺、廖玉美、廖建平、汤宗清等，谢谢你们真诚的友谊和热情的帮助。

陈雄华

目 录

第 1 篇 概述

第 1 章 Spring 概述	2
1.1 认识 Spring	3
1.2 Spring 带给我们什么	5
1.3 Spring 体系结构	6
1.4 Spring 2.0 的新功能	7
1.4.1 简化配置文件编写, 增强配置文件的扩展性	7
1.4.2 新 Bean 的作用域	8
1.4.3 AOP 的增强	8
1.4.4 持久层的增强	8
1.4.5 展现层的增强	8
1.4.6 其他	9
1.5 Spring 对 Java 版本的要求	9
1.6 如何获取 Spring	9
1.7 Spring 项目文件结构	11
1.8 Spring 的子项目	16
1.8.1 Spring Web Flow	16
1.8.2 Spring Web Services	16
1.8.3 Spring Security (Acegi Security)	17
1.8.4 Spring LDAP	17
1.8.5 Spring Rich Client	17
1.8.6 Spring Extensions Modules	17
1.8.7 Spring IDE for Eclipse	17
1.8.8 Spring BeanDoc	18
1.8.9 Spring OSGi	18
1.8.10 Spring .NET	18
1.9 和 Spring 相似的框架	18
1.9.1 PicoContainer	18
1.9.2 NanoContainer	19

1.9.3 Google Guice	19
1.9.4 Jdon Framework	19
1.10 小结	19
第 2 章 快速入门	20
2.1 实例功能概述	21
2.1.1 比 Hello World 更适用的实例	21
2.1.2 实例功能简介	21
2.2 环境准备	23
2.2.1 创建库表	23
2.2.2 建立工程	24
2.2.3 类包及 Spring 配置文件规划	25
2.3 持久层	26
2.3.1 建立领域对象	27
2.3.2 UserDao	28
2.3.3 LoginLogDao	30
2.3.4 在 Spring 中装配 DAO	31
2.4 业务层	32
2.4.1 UserService	32
2.4.2 在 Spring 中装配 Service	34
2.4.3 单元测试	35
2.5 展现层	37
2.5.1 配置 Spring MVC 框架	37
2.5.2 处理登录请求	39
2.5.3 JSP 视图页面	42
2.6 打包和部署	44
2.6.1 通过 Ant 打包	44
2.6.2 在 Tomcat 中部署	46
2.7 小结	47

第2篇 Spring 核心技术

第3章 IoC 容器概述..... 50

3.1 IoC 概述.....	51
3.1.1 通过实例理解 IoC 的概念.....	51
3.1.2 IoC 的类型.....	53
3.1.3 通过容器完成依赖关系的建立.....	55
3.2 相关 Java 基础知识.....	56
3.2.1 简单实例.....	56
3.2.2 类装载器 ClassLoader.....	58
3.2.3 Java 反射机制.....	60
3.3 BeanFactory 和 ApplicationContext.....	62
3.3.1 BeanFactory 介绍.....	62
3.3.2 ApplicationContext 介绍.....	66
3.3.3 父子容器.....	71
3.4 Bean 的生命周期.....	72
3.4.1 BeanFactory 中 Bean 的生命周期.....	72
3.4.2 ApplicationContext 中 Bean 的生命周期.....	80
3.5 小结.....	82

第4章 在 IoC 容器中装配 Bean..... 83

4.1 Spring 配置文档类型.....	84
4.2 Bean 基本配置.....	86
4.2.1 装配一个 Bean.....	86
4.2.2 Bean 的命名.....	86
4.3 依赖注入.....	87
4.3.1 属性注入.....	87
4.3.2 构造函数注入.....	90
4.3.3 工厂方法注入.....	94
4.3.4 选择注入方式的考量.....	95
4.4 注入参数详解.....	96
4.4.1 字面值.....	96
4.4.2 引用其他 Bean.....	97
4.4.3 内部 Bean.....	98
4.4.4 null 值.....	99
4.4.5 级联属性.....	99
4.4.6 集合类型属性.....	100
4.4.7 简化配置方式.....	103
4.4.8 自动装配.....	104

4.5 方法注入.....	109
4.5.1 lookup 方法注入.....	109
4.5.2 方法替换.....	110
4.6 <bean>之间的关系.....	111
4.6.1 继承.....	112
4.6.2 依赖.....	113
4.6.3 引用.....	114
4.7 Bean 作用域.....	115
4.7.1 singleton 作用域.....	115
4.7.2 prototype 作用域.....	117
4.7.3 Web 应用环境相关的 Bean 作用域.....	117
4.7.4 作用域依赖问题.....	119
4.8 FactoryBean.....	120
4.8.1 FactoryBean 的工作方式.....	121
4.8.2 集合 FactoryBean.....	122
4.8.3 属性 FactoryBean.....	124
4.9 小结.....	126

第5章 Spring 容器高级主题..... 127

5.1 Spring 容器技术内幕.....	128
5.1.1 内部工作机制.....	128
5.1.2 BeanDefinition.....	130
5.1.3 InstantiationStrategy.....	132
5.1.4 BeanWrapper.....	132
5.2 属性编辑器.....	133
5.2.1 JavaBean 的编辑器.....	133
5.2.2 Spring 默认属性编辑器.....	137
5.2.3 自定义属性编辑器.....	137
5.3 使用外部属性文件.....	140
5.3.1 使用外部属性文件.....	141
5.3.2 使用加密的属性文件.....	142
5.4 国际化信息.....	147
5.4.1 基础知识.....	147
5.4.2 MessageSource.....	153
5.4.3 容器级的国际化信息资源.....	156
5.5 容器事件.....	157
5.5.1 Spring 事件类结构.....	157

5.5.2 解构 Spring 事件体系的具体实现	159	7.2 JDK5.0 注解知识快速进阶	212
5.5.3 一个实例	159	7.2.1 了解注解	212
5.6 小结	161	7.2.2 一个简单的注解类	213
第 6 章 Spring AOP 基础	162	7.2.3 使用注解	214
6.1 AOP 概述	163	7.2.4 访问注解	215
6.1.1 AOP 到底是什么	163	7.3 着手使用@AspectJ	216
6.1.2 AOP 术语	165	7.3.1 使用前的准备	216
6.1.3 AOP 的实现者	167	7.3.2 一个简单的例子	216
6.2 基础知识	167	7.3.3 如何通过配置使用@AspectJ 切面	218
6.2.1 带有横切逻辑的实例	168	7.4 @AspectJ 语法基础	219
6.2.2 JDK 动态代理	170	7.4.1 切点表达式函数	220
6.2.3 CGLib 动态代理	173	7.4.2 在函数入参中使用通配符	221
6.2.4 AOP 联盟	175	7.4.3 逻辑运算符	221
6.2.5 代理知识小结	175	7.4.4 不同增强类型	222
6.3 创建增强类	176	7.4.5 引介增强用法	223
6.3.1 增强类型	176	7.5 切点函数详解	225
6.3.2 前置增强	177	7.5.1 @annotation()	225
6.3.3 后置增强	181	7.5.2 execution()	226
6.3.4 环绕增强	183	7.5.3 args()和@args()	228
6.3.5 异常抛出增强	184	7.5.4 within()	229
6.3.6 引介增强	186	7.5.5 @within()和@target()	230
6.4 创建切面	189	7.5.6 target()的 this()	231
6.4.1 切点类型	189	7.6 @AspectJ 进阶	233
6.4.2 切面类型	190	7.6.1 切点复合运算	233
6.4.3 静态普通方法名匹配切面	192	7.6.2 命名切点	234
6.4.4 静态正则表达式方法匹配切面	194	7.6.3 增强织入的顺序	235
6.4.5 动态切面	197	7.6.4 访问连接点信息	236
6.4.6 流程切面	201	7.6.5 绑定连接点方法入参	237
6.4.7 复合切点切面	203	7.6.6 绑定代理对象	239
6.4.8 引介切面	205	7.6.7 绑定类注解对象	240
6.5 自动创建代理	206	7.6.8 绑定返回值	240
6.5.1 实现类介绍	206	7.6.9 绑定抛出的异常	241
6.5.2 BeanNameAutoProxyCreator	207	7.7 基于 Schema 配置切面	242
6.5.3 DefaultAdvisorAutoProxyCreator	209	7.7.1 一个简单切面的配置	242
6.6 小结	210	7.7.2 配置命名切点	244
第 7 章 基于@AspectJ 和 Schema 的 AOP	211	7.7.3 各种增强类型的配置	245
7.1 Spring 2.0 对 AOP 的支持	212	7.7.4 绑定连接点信息	248
		7.7.5 Advisor 配置	249
		7.8 混合切面类型	249

7.8.1 混合使用各种切面类型	250	7.9.2 管理容器外的对象	253
7.8.2 各种切面类型总结	251	7.9.3 使用 Spring IoC 管理 AspectJ 切面	258
7.9 集成 AspectJ	252	7.10 小结	259
7.9.1 使用 AspectJ LTW	252		

第 3 篇 数据库访问

第 8 章 Spring 对 DAO 的支持

262

8.1 Spring 的 DAO 理念	263
8.2 统一的异常体系	263
8.2.1 Spring 的 DAO 异常体系	263
8.2.2 JDBC 的异常转换器	265
8.2.3 其他持久技术的异常转换器	266
8.3 统一数据访问模板	266
8.3.1 使用模板和回调机制	267
8.3.2 Spring 为不同持久化技术所提供 的模板类	269
8.4 数据源	270
8.4.1 配置一个数据源	270
8.4.2 获取 JNDI 数据源	273
8.4.3 Spring 的数据源实现类	273
8.5 小结	274

第 9 章 Spring 的事务管理

275

9.1 数据库事务基础知识	276
9.1.1 何为数据库事务	276
9.1.2 数据并发的问题	277
9.1.3 数据库锁机制	279
9.1.4 事务隔离级别	280
9.1.5 JDBC 对事务支持	281
9.2 ThreadLocal 基础知识	282
9.2.1 ThreadLocal 是什么	283
9.2.2 ThreadLocal 的接口方法	283
9.2.3 一个 ThreadLocal 实例	284
9.2.4 Thread 同步机制的比较	285
9.2.5 Spring 使用 ThreadLocal 解决线程 安全问题	286
9.3 Spring 对事务管理的支持	288
9.3.1 事务管理关键抽象	288
9.3.2 Spring 的事务管理器实现类	291
9.3.3 事务同步管理器	294

9.3.4 事务传播行为	295
9.4 编程式的事务管理	298
9.5 使用 XML 配置声明式事务	299
9.5.1 一个需要事务配置的服务接口	301
9.5.2 使用原始的 TransactionProxy FactoryBean	302
9.5.3 基于 tx/aop 命名空间的配置	304
9.6 使用注解配置声明式事务	307
9.6.1 使用 @Transactional 注解	307
9.6.2 通过 AspectJ LTW 引入事务 切面	310
9.7 JTA 事务管理	311
9.7.1 集成 JOTM, 在 Spring 中配置 JTA 事务	311
9.7.2 使用 Tomcat/JOTM/Spring 提供 JTA 事务	315
9.7.3 在特定应用服务器使用 JTA	317
9.8 小结	318

第 10 章 使用 Spring JDBC 访问数据库

319

10.1 使用 Spring JDBC	320
10.1.1 JdbcTemplate 小试牛刀	320
10.1.2 通过 JdbcDaoSupport 使用 JdbcTemplate	321
10.2 基本的数据操作	322
10.2.1 更改数据	322
10.2.2 返回数据库的表自增主键值	324
10.2.3 批量更改数据	326
10.2.4 查询数据	327
10.2.5 查询单值数据	330
10.2.6 调用存储过程	331
10.3 BLOB/CLOB 类型数据的操作	334
10.3.1 如何获取本地数据连接	334
10.3.2 相关的操作接口	335

10.3.3	插入 Lob 类型的数据	337	11.4.3	使用 JVM 启动参数注册转换器 的问题	377
10.3.4	以块数据方式读取 Lob 数据	339	11.4.4	JPA 的 ClassTransformer SPI 接口	378
10.3.5	以流数据方式读取 Lob 数据	340	11.5	使用 LocalContainerEntityManager FactoryBean	378
10.4	自增键和行集	341	11.5.1	相应的配置信息	378
10.4.1	自增键的使用	341	11.5.2	使用 JVM 代理启动参数实施 织入器	382
10.4.2	以行集返回数据	344	11.5.3	在测试环境下实施织入器	382
10.5	其他类型的 JdbcTemplate	344	11.5.4	作为 Web 应用部署时实施 织入器	383
10.5.1	NamedParameterJdbcTemplate	345	11.5.5	处理多个持久化单元	385
10.5.2	SimpleJdbcTemplate	346	11.6	使用原生的 JPA	386
10.6	以 OO 方式访问数据库	348	11.7	事务处理	387
10.6.1	使用 MappingSqlQuery 查询 数据	348	11.8	小结	388
10.6.2	使用 SqlUpdate 更新数据	349	第 12 章	整合其他 ORM 框架	389
10.6.3	使用 StoredProcedure 执行存储 过程	350	12.1	Spring 整合 ORM 技术	390
10.6.4	SqlFunction 类	351	12.2	在 Spring 中使用 Hibernate	391
10.7	小结	352	12.2.1	配置 SessionFactory	391
第 11 章	使用 JPA 访问数据库	353	12.2.2	使用 HibernateTemplate	394
11.1	JPA 概述	354	12.2.3	处理 LOB 类型数据	396
11.1.1	什么是 JPA	354	12.2.4	添加 Hibernate 事件监听器	398
11.1.2	JPA 的发展及影响	354	12.2.5	使用原生 Hibernate API	399
11.1.3	目前已经有哪些实现	355	12.2.6	使用 JPA 注解的 Hibernate	399
11.2	JPA 快速进阶	355	12.2.7	事务处理	401
11.2.1	实体对象	356	12.2.8	延迟加载的问题	401
11.2.2	使用注解元数据	357	12.3	在 Spring 中使用 iBatis	402
11.2.3	使用 XML 元数据	361	12.3.1	配置 SqlMapClient	403
11.2.4	JPA 的体系结构及重要的 API	363	12.3.2	使用 SqlMapClientTemplate	404
11.2.5	JPA 的查询语言	365	12.3.3	处理 LOB 类型数据	405
11.3	使用 LocalEntityManagerFactoryBean	368	12.4	混合使用多种持久化技术	407
11.3.1	认识 JpaTemplate	368	12.4.1	透析 Spring 中混合持久化技术 的使用	407
11.3.2	使用 JpaTemplate 编写数据 访问类	369	12.4.2	一个使用混合持久化技术的 实例	408
11.3.3	编写相应的配置信息	370	12.5	DAO 层设计	410
11.3.4	测试 BbtForum	373	12.5.1	DAO 基类的设计	410
11.4	JVM Class 文件字节码转换基础 知识	374	12.5.2	查询接口方法的设计	412
11.4.1	java.lang.instrument 包的工作 原理	374	12.6	小结	414
11.4.2	如何向 JVM 中注册转换器	375			

第4篇 业务层应用

第13章 任务调度和异步执行器 416

13.1 任务调度概述	417
13.2 Quartz 快速进阶	417
13.2.1 Quartz 基础结构	418
13.2.2 使用 SimpleTrigger	420
13.2.3 使用 CronTrigger	422
13.2.4 使用 Calendar	425
13.2.5 任务调度信息存储	427
13.3 在 Spring 中使用 Quartz	430
13.3.1 创建 JobDetail	430
13.3.2 创建 Trigger	432
13.3.3 创建 Scheduler	434
13.4 Spring 中使用 JDK Timer	435
13.4.1 Timer 和 TimerTask	436
13.4.2 Spring 对 JDK Timer 的支持	438
13.5 Spring 对 JDK 5.0 Executor 的支持	439
13.5.1 了解 JDK 5.0 的 Executor	440
13.5.2 Spring 对 Executor 所提供的 抽象	442
13.6 实际应用中的任务调度	443
13.6.1 如何产生任务	444
13.6.2 任务调度对应用程序集群的 影响	446
13.6.3 Web 应用程序中调度器的启动和 关闭问题	447
13.7 小结	449

第14章 JavaMail 发送邮件 450

14.1 JavaMail 快速进阶	451
14.1.1 JavaMail 概述	451
14.1.2 JavaMail 的关键对象	452
14.1.3 使用 JavaMail 发送邮件	456
14.1.4 处理 SSL 加密传输协议邮件	458
14.2 Spring 的邮件支持	459
14.2.1 邮件发送高级抽象层	459
14.2.2 Spring 对 JavaMail 的支持	461
14.3 发送各种形式的邮件	464
14.3.1 发送纯文本邮件	465

14.3.2 发送 HTML 类型的邮件	466
14.3.3 发送带内嵌文件的邮件	467
14.3.4 发送带附件的邮件	468
14.3.5 发送纯文本和 HTML 双版本的 邮件	469
14.4 在实际应用中发送邮件	471
14.4.1 使用邮件模板	472
14.4.2 异步发送邮件	474
14.5 小结	475

第15章 在 Spring 中使用 JMS 477

15.1 JMS 快速进阶	478
15.1.1 JMS 的两种消息类型	479
15.1.2 JMS 受管理的对象	479
15.1.3 JMS 的一些重要接口	480
15.1.4 JMS 消息的结构	481
15.1.5 消息收发的机制	483
15.1.6 发送消息	484
15.1.7 接收消息	485
15.2 JMS 提供者	487
15.2.1 JMS 提供者概述	487
15.2.2 使用 ActiveMQ	488
15.3 Spring 对 JMS 的支持	489
15.3.1 JMS 模板类	490
15.3.2 JMS 异常映射	491
15.4 使用回调接口发送/接收消息	491
15.4.1 使用 send() 发送消息	491
15.4.2 使用 receive() 接收消息	494
15.4.3 使用 execute() 方法	495
15.5 使用消息转换器发送/接收消息	497
15.5.1 消息转换器	497
15.5.2 发送 POJO 消息	498
15.5.3 接收 POJO 消息	500
15.6 消息过滤及异步消息接收	500
15.6.1 对消息进行过滤接收	501
15.6.2 消息监听器容器	501
15.6.3 异步消息接收	503
15.6.4 异步消息接收适配器	504

15.7 JMS 事务管理	506
15.7.1 本地事务	506
15.7.2 JTA 事务	507
15.7.3 消息重复接收问题	508
15.8 小结	509
第 16 章 在 Spring 中开发 Web Service	511
16.1 Web Service 简介	512
16.1.1 Web Service 相关概念	512
16.1.2 SOAP——简单对象传输协议	513
16.1.3 WSDL——Web Service 描述语言	513
16.1.4 UDDI——统一描述、发现和集成	514
16.2 XFire 简介	514
16.2.1 XFire 特性	515
16.2.2 XFire 体系及重要 API	515
16.2.3 XFire 类库说明	517
16.3 将 POJO Bean 导出为 Web Service	518
16.3.1 使用导出器导出 Web Service	518
16.3.2 配置 web.xml	519
16.3.3 使用 JSR 181 注解导出 Web Service	521
16.4 各种客户端调用方式	522
16.4.1 使用服务端的窄接口类	522
16.4.2 使用 WSDL 文件构造客户端程序	524
16.4.3 根据 WSDL 生成客户端代码	525
16.5 Web Service 的测试	526
16.5.1 基于 SOAP 报文的纯服务端测试	527
16.5.2 在 JVM 模式通过客户端进行测试	529
16.6 Web Service 安全	530
16.6.1 认识 WS-Security	530
16.6.2 XFire 应用 WS-Security 的总体方案	531
16.7 使用 WS-Security	532
16.7.1 准备工作	532
16.7.2 使用用户名/密码进行身份认证	534
16.7.3 对 SOAP 报文进行数字签名	539
16.7.4 对 SOAP 报文体进行加密	542
16.7.5 组合安全策略	546
16.8 小结	550
第 17 章 使用 Acegi 实施应用系统安全	552
17.1 认识 Acegi	553
17.1.1 Acegi 概述	553
17.1.2 Acegi 体系结构	554
17.1.3 重要组件类介绍	556
17.1.4 用户认证过程	558
17.1.5 安全对象访问控制	559
17.1.6 Acegi 版本升级的一些重大变化	560
17.2 将 Acegi 集成到 Web 应用程序中	561
17.2.1 代理 Acegi 的过滤器	562
17.2.2 使用代理过滤器链对处理 HTTP 请求	563
17.3 身份认证管理	565
17.3.1 基于内存存储用户信息的身份认证	565
17.3.2 基于数据库存储用户信息的认证	570
17.3.3 密码加密问题	572
17.3.4 成功登录系统的后置处理	574
17.3.5 在多个请求之间共享 SecurityContext	575
17.3.6 退出系统的后置处理	576
17.3.7 实施 Remember-Me 认证	577
17.4 安全对象访问控制	581
17.4.1 匿名用户权限信息	582
17.4.2 URL 资源访问控制	584
17.4.3 异常转换过滤器	587
17.4.4 业务类方法访问控制	588
17.5 其他	590
17.5.1 Acegi 标签库	590
17.5.2 Session 并发控制	591
17.5.3 HTTPS 安全通道	593
17.6 小结	597
第 18 章 动态语言支持	599
18.1 动态语言介绍	600
18.1.1 动态语言的特性及适用场合	600

18.1.2 Ruby 快速入门	601	18.2.3 基于 Groovy 的 Bean	614
18.1.3 Groovy 快速入门	606	18.2.4 基于 BeanShell 的 Bean	614
18.1.4 BeanShell 快速入门	608	18.3 其他支持功能	615
18.2 在 Spring 中使用动态语言	611	18.3.1 动态刷新	615
18.2.1 Spring 对动态语言的支持	611	18.3.2 内联脚本	616
18.2.2 基于 JRuby 的 Bean	612	18.4 小结	617

第 5 篇 展现层应用

第 19 章 Spring MVC 之一 620

19.1 Spring MVC 概述	621
19.1.1 体系结构	621
19.1.2 认识并配置 DispatcherServlet	622
19.1.3 一个包含以上步骤的实例	628
19.2 将请求映射到处理器	631
19.2.1 HandlerExecutionChain 处理 器链	631
19.2.2 处理器映射 HandlerMapping	633
19.2.3 处理器的拦截器	635
19.2.4 多个处理器映射器	636
19.3 各种类型的控制器	637
19.3.1 参数映射控制器	639
19.3.2 文件名映射控制器	639
19.3.3 简单类型控制器	640
19.3.4 命令控制器	641
19.3.5 表单控制器	643
19.3.6 中文乱码问题	652
19.3.7 向导控制器	653
19.3.8 多动作控制器	659
19.3.9 一次性控制器	661
19.4 使用 Spring 文件上传功能	662
19.4.1 配置 multipart 请求解析器	662
19.4.2 编写控制器和文件上传表单 页面	663
19.5 本地化解析	665
19.5.1 使用 Cookie 本地化解析器	666
19.5.2 使用 Session 本地化解析器	667
19.5.3 使用本地化更改拦截器	668
19.6 使用主题	669
19.6.1 主题资源及样式文件的组织	669
19.6.2 主题样式的引用	671

19.6.3 主题资源和主题解析器	672
19.7 异常处理	673
19.8 小结	673

第 20 章 Spring MVC 之二 675

20.1 视图和视图解析器	676
20.1.1 认识视图	676
20.1.2 认识视图解析器	677
20.1.3 使用视图解析器	678
20.2 Spring MVC 表单标签	682
20.2.1 form 标签	683
20.2.2 输入组件标签	684
20.2.3 单选框和复选框组件标签	684
20.2.4 下拉框组件标签	688
20.2.5 错误标签	690
20.3 使用模板视图	690
20.3.1 FreeMarker 概述	690
20.3.2 常见的 FreeMarker 特殊标记	692
20.3.3 编写渲染模型数据的模型	693
20.3.4 配置 FreeMarker 环境和视图	694
20.3.5 使用 Spring 宏	695
20.3.6 使用 Velocity 模板	698
20.4 使用 Tiles 进行页面布局	699
20.4.1 编写一个 Tiles 模板	699
20.4.2 使用 Tiles 视图	702
20.5 产出非 HTML 的输出	703
20.5.1 使用 PDF 文档进行输出	703
20.5.2 使用 Excel 进行输出	706
20.6 小结	709

第 21 章 集成其他 Web 框架 710

21.1 集成 Struts 2	711
------------------------	-----