Java开发专家

精通Spring 2.0

Mastering Spring 2.0

源于Java EE，服务Java EE

超越Java EE，直击Spring 2.0的核心思想

全面跟进Spring 2.0的最新技术

罗时飞 编著

飞思科技产品研发中心 监制

電子工業出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

http://www.phei.com.cn

Java开发专家

精通Spring 2.0

Mastering Spring 2.0

罗时飞 编著
飞思科技产品研发中心 监制

电子工业出版社
Publishing House of Electronics Industry

内 容 简 介

本书是关于Spring 2.0的权威教程，是Java/Java EE开发者必备的参考书。本书详尽系统地介绍了Java EE的基础知识、Spring 2.0的各种功能，以及Spring 2.0的高级使用技巧和最佳实践。全书共分为4篇：第1篇介绍Spring 2.0核心技术，主要围绕Spring元框架进行阐述；第2篇介绍DAO层集成技术，主要围绕JDBC、Hibernate和JPA等持久化技术展开论述，针对Spring使能应用的事务管理和集成测试，也进行了相关介绍；第3篇介绍Java EE服务及技术的集成，主要围绕企业应用中使用的各种Java EE服务及技术展开论述；第4篇介绍Spring 2.0最佳实践，主要从平台差异性和应用差异性角度给出论述。全书理论与实践并重，通过大量的实例帮助读者尽快掌握Spring 2.0的使用技巧，从而提高本书的参考、阅读价值。

本书适合作为Java/Java EE开发者、系统分析师和架构师的参考书，同时，本书非常适合于高校相关专业的学生，以及对Java/Java EE开发有兴趣的各类开发者。

图书在版编目（CIP）数据

精通Spring 2.0 / 罗时飞编著. —北京：电子工业出版社，2007.1

（Java开发专家）

ISBN 7-121-03551-0

Ⅰ.精... Ⅱ.罗... Ⅲ.Java语言—程序设计 Ⅳ.TP312

中国版本图书馆CIP数据核字（2006）第141555号

责任编辑：王树伟

印　　刷：北京东光印刷厂

装　　订：三河市皇庄路通装订厂

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

开　　本：787×1092　1/16　印张：34.75　字数：889.6千字

印　　次：2007年1月第1次印刷

印　　数：6 000册　　定价：55.00元

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，联系电话：（010）68279077；邮购电话：（010）88254888。

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

服务热线：（010）88258888。

热烈祝贺“开发专家之 Sun ONE”系列被
评为电子工业出版社“最佳品牌奖”

“开发专家之 **Sun ONE**”全新提升为“**Java** 开发专家”系列
——源自精品 成就理想

FOREWORD

出版说明

★ 从“开发专家之 Sun ONE”到“Java 开发专家”

“开发专家之 Sun ONE”系列丛书从诞生之日至今，已经四岁了。在这个系列里面，我们一直努力体现着这么一个理念：用一种较为敏锐的视角来跟踪 IT 技术的发展轨迹，并把可能为广大程序员所希望获得的知识，用图书出版的方式奉献给大家。

在这个系列中，我们陆续出版了约 30 种图书，有《Java 与模式》、《JSP 应用开发详解（第二版）》、《精通 EJB（第三版）》、《Tomcat 与 Java Web 应用开发详解》、《精通 Struts：基于 MVC 的 Java Web 设计与开发》、《JBoss 管理与开发核心技术（第三版）》、《精通 Spring》、《精通 Hibernate：Java 对象持久化技术详解》等一大批读者朋友耳熟能详的作品。很多作品都是在国内没有同类图书的情况下出版的。在这几年的出版工作中，我们时刻感受着市场的风险，也时刻收获着无数读者给我们的认可。

在这个系列中，凝聚了大量资深技术专家的心血。有大家都熟知的阎宏、刘晓华、孙卫琴、罗时飞等，还有一些正在不断腾跃的开发高手。这些非常优秀的国内原创作者们一直都在支持着“开发专家之 Sun ONE”系列的出版工作，在这里，我们要向他们说声：谢谢。

桃李不言，下自成蹊。由于这些年“开发专家之 Sun ONE”在“两个效益”中的杰出表现，电子工业出版社授予这个系列“最佳品牌奖”。

时代不断前进，技术不断变革。为了顺应 Java 领域的技术发展态势，为了赋予这个经典的图书系列更强的生命力，我们将“开发专家之 Sun ONE”升级为“Java 开发专家”。我们将继承原有的出版理念，紧密跟踪技术热点和发展趋势，会聚更多优秀作者，全力奉献更经典的作品。

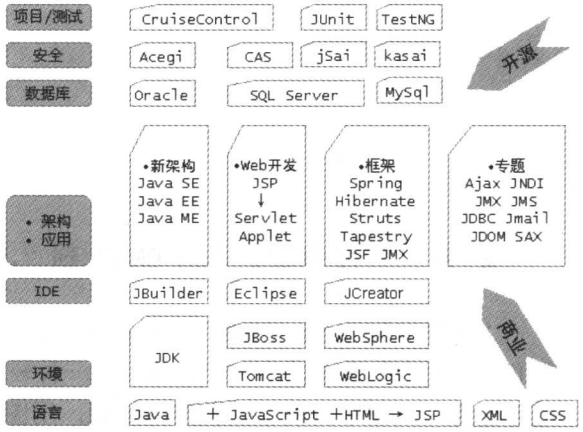
★ 规划你的 Java 开发之路

喜马拉雅山脉的最高峰不断地在温室效应中降低，而 Java 世界的颠峰永远都在技术人员的追求中不断升高。每个人都有不同的路，每个人都有不同的行路方式，不过，往往“到了山顶才发现，错误的路和正确的路就差那么几步！”

身处 Java 洪流中的程序员最累（不过大家都说 Java 程序员薪水最高，呵呵），我们简单

整理了一下 Java 领域的相关技术、工具、架构，如下图所示。这个框图中的每一个英文单词（或缩写）都可以写成一本书。Java 领域还有一个特点，那就是商业产品和开源产品层出不穷，潮流不断。相比于其他领域，如.NET，Java 开发更是体现了这句谚语：条条大路通罗马。

罗马只有一个，大路却有多条。看上去，似乎到罗马很容易，反正路多嘛。不过，路多却容易迷失方向。当你在 Java 领域中摸爬滚打几年后，发现自己在无数条道路上走了很久，却不知道罗马何日才能到达，甚至连罗马的方向都不知道，这时你肯定会很失落。

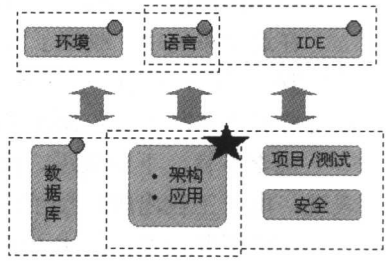


很遗憾，在这个简短的出版说明文章里面，我们无法告诉你每一条连贯的、不费周折的通往罗马的道路该如何走。或许，通过“Java 开发专家”系列中的某本书，你可以找到属于你的正确道路。在一般情况下，我们不会就某一项很窄的话题来单独写一本书，我们还是希望通过我们的一些专业和智慧，尽力把一些相关技术整合起来，用较为简明的方式表达出来，最后由你来选择。

这里有句话与大家共勉：少走弯路，就是捷径！

★ “Java 开发专家”的奉献

犹如在上面那个框图中展现的那样，我们希望在各个层面、各个方向上都能给读者奉献出优秀的图书作品，全面体现技术与应用的结合。从宏观上看，我们会从语言、IDE、环境、数据库、架构与应用、安全、项目与测试等方面进行选择，选出一些读者迫切需要的技术来先行规划。



“Java 开发专家”虽然新蓓初绽，但因其源自盛放的“开发专家之 Sun ONE”系列而根基稳健，两个系列会有一段很长的并行时间，我们会用一种优化的方式来保证读者的顺利选择。无论哪一个系列，必定都有大家喜欢的图书。

在技术上，有着持久化的方法，在学习上，也需要有持久化的精神。
从“开发专家之 Sun ONE”到“Java 开发专家”，希望可以带给你持久化的动力。

联系方式

咨询电话：（010）68134545　　88254160
电子邮件：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、飞思、飞思教育、飞思科技、FECIT

PREFACE

序

Spring 2.0 是具体与抽象的统一体，而本书正是围绕这一统一体展开论述的。

Java 与 Java EE 平台规范标准化了各种具体 Java 技术，比如，JDBC、JMX、Servlet、Annotation 注释和 JNDI API 等，这些具体技术正是 Spring 2.0 统一体中的具体侧面。Spring 2.0 是 Java 和 Java EE 架构级框架，其依托的正是这些具体 Java 技术。各种标准 Java EE 技术是 Spring 2.0 的基础，如果缺乏 JCP 制订的各项 JSR 规范，Spring 2.0 使能应用则不可能到处运行。注意，“到处”一词可以表示与操作系统无关、Java EE 应用服务器无关、JVM 无关等。

默认时，各种标准的具体的 Java EE 技术（服务）的使用并不轻松，比如，开发者要手工处理一堆受查异常、进行资源的查找和释放，应用往往要同具体的 Java EE API 耦合在一起。真正的面向对象（OO）、POJO 编程模型很难应用到 Java EE 使能应用中。客观地说，Java EE 平台并未提供良好的编程模型。Spring 框架于 2002 年就看到了 Java EE 平台中的这一断层，集成各种 Java EE 技术及服务成为了 Spring 的首要任务。此后，各种 Java EE 技术及服务的使用得到大大简化，因为 POJO 编程模型被 Spring 引入到这一领域。如今，开发基于 Java EE 的 Spring 使能应用同编写 POJO 并无实质性的差异。Spring 框架提供的 POJO 编程模型赢得了开发者的广泛支持和拥戴，而这一 POJO 编程模型正是 Spring 2.0 统一体中的抽象侧面。

开发者都知道，Java 本身是跨平台的。与此同时，JCP 组织负责制订各项 JSR 规范，而不同的 Java、Java EE 厂商负责实现它们，并相互竞争。此后，一旦开发者开发出基于 JSR 规范（比如，Java EE 5 规范、JDBC 4.0 规范）的企业应用，由不同厂商实现的 JSR 规范运行时都将便携地宿主它们，比如，同一 Web 应用能够同时运行在不同厂商提供的 Java EE 应用服务器中。为完成集成各种 Java EE API 的夙愿，Spring 尽量采用 Java 平台引入的标准技术实施这一集成工作，比如，采用动态代理、ThreadLocal、Java 反射机制、各种设计模式。这说明，Spring 框架从未对目标运行环境进行任何假定，因为它依赖的基础始终是 Java 平台中标准的 Java 技术。因此，开发者在使用 Spring 框架开发企业应用时，应用的便携性并未削弱，Java EE 应用服务器提供的各种容器服务也能够暴露给应用，而且

Spring 还针对各种 Java EE 技术、Java EE 应用服务器的专有特性进行了集成和封装。

为了使得开发者在使用 Spring 框架开发 Java EE 应用期间能够享受到 POJO 编程模型，Spring 引入了 Spring DI 容器和 Spring AOP 技术实现，它们在倡导 POJO 编程模型期间的功劳是最大的。类似于宿主 EJB 3.0 组件的 EJB 容器，Spring 2.0 提供的控制反转（DI）容器能够宿主 POJO 及各种 Java 组件。DI 容器负责 POJO 的管理，比如，为它提供事务服务、生命周期服务、线程服务、缓存服务和安全性服务等。可以看出，开发者在使用 POJO 期间，如果不存在 Spring DI 容器，POJO 的功效将大打折扣。DI 容器为 POJO 提供了运行时支持。与此同时，AOP 技术使得应用能够透明地享受到 Java EE 容器提供的企业级服务。借助于 DI 容器能够便捷地实施 Spring AOP 技术，它们一同为企业应用效力。

实际上，各种控制反转容器（比如，HiveMind）都支持并提倡 POJO 编程模型，这也是目前的主流开发模型。为差异化其他 DI 容器，Spring DI 容器针对各种 DAO 层集成技术（比如，Hibernate、JPA 和 JDBC）、Java EE 服务及技术（比如，JMX、JCA CCI 和 JMS）提供了一流的集成支持。很显然，这进一步拓展了 Spring 提倡的 POJO 编程模型的应用领域，因为它已将触角延伸到企业级 Java 领域。在集成各种企业级 Java 技术期间，DI 容器和 AOP 技术实现是基础，也正因为这一优异的基础使得 Spring 集成企业级 Java 技术变得轻松、自然。在某种程度上，DI 容器和 AOP 技术实现是元框架（meta-framework），Spring 框架本身具有自我繁殖能力，它提供的各种企业级 Java 技术集成支持是在这一元框架基础之上造就的，甚至 Spring 2.0 还在不断丰富自身的企业级 Java 技术集成支持（比如，Spring OSGi 项目）。

Spring IoC 容器、AOP 技术实现、各种 Java EE API 集成支持，这三者形成了鼎足之势，开发者可以在自身的应用中单独使用它们中的任何一者，但如果同时使用它们，POJO 编程模型的威力将发挥到极致。我们也建议开发者直接采纳这三剑客，基于它们实施测试驱动开发（TDD）也非常容易。TDD 理论到处都存在，但唯独 Spring 实现的 TDD 支持称得上是真正支持 TDD 的，尤其是 Spring 提供的集成测试支持。除了 POJO 编程模型外，TDD 也是 Spring 最为看重的。未经单元、集成测试的代码便交付使用，这是非常危险的做法。

2006 年中旬，Java EE 5 规范最终版顺利发布，同年 10 月份，Spring 2.0 最终版也宣告顺利发布。Java EE 5 大大简化了开发和部署模型，它广泛使用 Annotation 注释技术。EJB 3.0（包括 JPA）是 Java EE 5 中的最大受益者。Java EE 5 的推出，使得它往 POJO 编程模型的方向迈出了坚实的一大步， EJB 3.0 还支持拦截器（@Interceptors）。为“应对”Java EE 5，Spring 2.0 诞生了。

在简化<bean/>配置、降低采纳 AOP 技术的门槛、增强 AOP 技术实现、引入各种 XML 命名空间等方面，Spring 2.0 付出了巨大努力。如今，Spring XML 配置文件是基于 XML Schema 的，不再是基于 DTD 的，尽管它也支持 DTD；<aop>命名空间的引入使得 AOP 技术的使用变得更简单，尽管 Spring 1.x 风格的 AOP 使用也很简单；同 AspectJ 5 进行了一流的集成支持，现在 Spring 2.0 使能应用能够享受到来自 AspectJ 5 中的 pointcut 表达语言；提供一流的 JPA 集成支持，引入消息监听器容器，以支持 MDP 编程模型等这些特点使得 Spring 2.0 开辟了新的企业级 Java 应用领域。

同 Spring 1.x 相比，Spring 2.0 的变化实在是太大了，简直可以使用 Spring 2.0 时代来形容。我们有理由相信 Spring 2.0 又将在引领 POJO 编程模型方面作出卓越的贡献，开发者

将是其中的最大受益者。

在完成《精通 Spring》（第 1 版）后，作者已经在开始着手准备第 2 版的写作计划了。在通过 http://www.open-v.com 网站收集读者（包括中国台湾读者）的写作建议、总结处女作第 1 版的各种得失后，《精通 Spring 2.0》（第 2 版）终于浮出水面。从这本书开始，图书创作成了作者的重要工作内容，本书也正是作者投入大量精力而收获的成果，意义非同寻常。期间，尤其要感谢出版社、家人、周边朋友及各位读者的鼓励和大力支持。另外，还要感谢作者曾经工作过的 E3 团队，这是一个富有激情的 Team。

Spring 2.0 涉及的知识面非常多，内容很广泛，错误在所难免，希望读者、同行批评指正。

最后，作者将此书献给所有 Java/Java EE 从业人员，希望在校学生、Java 开发者和架构师等角色能够从本书找到所需要的精彩内容，衷心祝愿成功和进步相伴你们每一天。

Life with Passion!

罗时飞
2006 年于广州

网站介绍

关于 www.open-v.com

我们专注于 Java EE 平台、敏捷方法及开源技术咨询工作。图书创作能够将我们的咨询经验带给整个社区，我们也希望在不远的将来，open-v.com 能够成为图书创作的重要平台。有特色的技术培训能够缩短开发者、开发团队采纳敏捷技术、方法的时间，因为我们的培训注重方法、理念、学习方法和学习能力的提高。有价值的咨询服务也是我们的重点，我们非常注重咨询的效果，从而真正为企业带来实实在在的价值。

FOREWORD

前　言

毫不夸张地说，Spring 2.0 是一套有关 Java EE API 的百科全书，它针对各种 Java EE API 的使用都提供了一流的、一致的抽象和集成工作，从而统一 Java EE API 暴露给开发者的客户视图。开发者都知道，Java EE API 本身的使用非常繁琐，许多与业务无关的技术细节需要开发者悉心打理。稍有不慎，各种 Java EE 问题随之而来，而 Spring 2.0 正是为解决 Java EE 编程模型中的这些问题而出现的。

为完成各种 Java EE API 的集成工作，Spring 开发团队提供了 Spring 元框架，即控制反转容器（IoC）和 AOP 技术实现。所有的 Java EE API 集成工作都是在这一元框架基础之上构建的。从目前来看，Spring 2.0 主要提供了 3 方面的 Java EE API 集成：DAO 层集成技术；Java EE 服务及技术；Web 层支持。

本书正是围绕 Spring 2.0 中的上述各项内容而准备的。

本书特点

时隔两年后，《精通 Spring 2.0》（第二版）成功写作完成，并出版发行。同《精通 Spring》（第一版）相比，本次改进、新增的内容非常多，下面总结了本书的特点。

- 全面跟进 Spring 2.x。同 Spring 1.x 相比，Spring 2.x 改进的内容非常多。其一，引入基于 XML Schema 的配置，从而大大简化了 Spring 配置文件的管理，比如，事务管理、JNDI 查找等；其二，同 AspectJ 5 进行了无缝集成，如今，Spring 2.0 开发者能够享受到@AspectJ 风格的切面、pointcut 表达语言，甚至，开发者可以针对领域对象实施依赖注入，并在 Spring DI 容器外享受到@Transactional 注释带来的 Spring 受管事务；其三，<bean/>的作用范围被扩充了，在 Spring 1.x 中，仅存在单例和原型作用范围的<bean/>。自 Spring 2.0 开始，开发者能够享受到处于 request、session、globalSession 作用范围的<bean/>，从而提升 Spring 2.0 在企业中的应用强度和深度；其四，Spring 2.0 全面拥抱 Java SE 5/Java EE 5，各种 Annotation 注释（比如，@Required、@Configurable）被引入到 Spring 中、一流的 JPA 集成也被包括在 Spring 2.0 中、JDBC 集成引入了命名参数和泛型支持等；其五，异步 JMS 支持，Spring 2.0 引入了消息监听器容器，如今，开发者可以享受到 MDP 编程模型，甚至，JMS 远程服务也被 Spring

2.0 囊括了；其六，动态脚本语言（比如，Groovy、JRuby 和 BeanShell）集成支持；其七，TaskExecutor 抽象；其八，提升测试驱动开发（TDD）支持，比如 Spring 2.0 于 AbstractTransactionalSpringContextTests 集成测试支持类中新引入了 endTransaction() 和 startNewTransaction()方法，甚至，Spring 2.0 还针对 JPA 的集成测试引入了 org.springframework.test.jpa 包，而且它还引入了基于 Annotation 注释技术（Java SE 5+）的 org.springframework.test.annotation 包以简化集成测试工作的展开。上述所有内容，本书进行了全方位跟进。

- 尽量将 Spring 最实用的动人的一面展现给读者。
- 在写作过程中，理论与实践知识并重。事实上，Spring 2.0 为那些打算涉足 Java EE 开发领域的开发者创造了条件，因为 Spring 降低了 Java EE 平台技术的学习曲线。一旦开发者初步熟悉 Spring 后，再深入到各 Java EE API 也是不错的选择。本书在介绍 Java EE API 集成工作前，对它们的背景和基础知识进行了详尽阐述。与此同时，各章内容采用的示例都是单独的自成一体的经典 Eclipse 项目。
- 在代码示例的选材上，力求经典和权威。Spring 2.0 内置了展示 Spring 特性的各种示例，比如，countries、petclinic、jpetstore、fortune 和 imagedb，本书在结合它们阐述各 Spring 知识点过程中，不时修改和扩展，甚至新增了基于不同技术栈的示例实现，比如，实现了 Hibernate 版本的 imagedb 和扩展的 fortune 等。这些示例的升值空间很大，因为 Spring 开发团队在不断完善它们，它们也体现了 Spring 的最新特性。现在，开发者可以一劳永逸地享受到这些示例带来的快乐。
- 无论知识体系，还是写作风格，各章内容统一、自成一体，开发者阅读起来非常舒服。
- 作者尽量将自身架构和开发大型 Java EE/Spring 使能项目的经验、进行 Java EE 咨询期间获得的 Spring 高级技巧和最佳实践体现在书中。
- 不断改进图书内容。如今，图书创作是作者的主要工作内容之一，因此自身有更大的责任和更多时间来完善此书。

让我们共同期待第三版的出现吧！

服 务 网 站

针对书中展示的各种 Java 代码、Ant build.xml 和其他脚本，我们特别提供了相关网站（http://www.open-v.com）。同时，为保证图书同 Spring 2.0 最新发布版的同步，我们会时常更新图书中的源代码，并公布到这一网站中，欢迎广大读者下载使用。

著　者

联系方式

咨询电话：（010）68134545　88254160

电子邮件：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、飞思、飞思教育、飞思科技、FECIT

CONTENTS

目　录

第 1 篇　Spring 2.0 核心技术

第 2 篇　DAO 层集成技术

PART

One

第 1 篇　Spring 2.0 核心技术

随着 Java EE 5 平台标准的发布，新一轮的技术革新继而上演。在 POJO 编程模型领域，Spring 一直以领先者的身份出现。Spring 1.x 获得了巨大成功，它倡导的 POJO 编程模型加速了 TDD 的实施。Spring 2.0 在 Spring 1.x 基础上，大大简化了应用配置的同时，又保证了 Spring 1.x 兼容性，它引入了新的命名空间、与 AspectJ 5 进行无缝集成、提供一流的 JPA 集成支持、内置 MDP 编程模型。

话说，“分久必合，合久必分”。在 Java EE 平台技术未出现前，各个服务器厂商以专有行为开发自身的服务器产品。在 Java EE 平台技术出现后，各个服务器产品遵循了 Java EE 相关技术规范。它们在遵循规范的同时，又引入了自身的专有行为，有些是由于 Java EE 规范未定义清晰导致的，有些是厂商的理解不到位造成的。Spring 社区出现后，这些专有行为又被统一到一起，借助于 DI 容器能够合理屏蔽各厂商的专有行为。DI 容器的使用使得应用代码不再同具体 Java EE 容器耦合在一起，而且 AOP 技术的引入使得 DI 容器中受管 POJO 的威力更大。可以预见，Spring 2.0 将在 Java EE 5 时代成为主角、颠覆者，因为它提供的 DI 容器、AOP 技术已经远远超越了 Java EE 容器提供的依赖性管理支持和 @Interceptors 拦截器支持。同 Spring 1.x 相比，借助 Spring 2.0 实施 TDD 更为敏捷。

Spring 2.0 核心技术将在本部分得到全方位的讨论。DI 容器和 AOP 技术实现是整个 Spring 2.0 框架的基础和 Spring 元框架（meta-framework）。Spring 2.0 提供的 DAO 层集成技术（第二部分）、Java EE 服务及技术集成（第三部分）是建立在 DI 容器和 AOP 技术实现基础上的。类似于 Eclipse 平台技术，Spring 框架具有自我繁殖能力，它是建立在元框架基础之上的框架。开发者可以通过 Eclipse IDE 开发 Eclipse 插件，从而扩充 Eclipse IDE 的功能。相比之下，Spring 2.0 也是如此。

本部分包括如下内容。