

Spring攻略



[Spring攻略_下载链接1](#)

著者:Gary Mak

出版者:人民邮电出版社

出版时间:2012-3

装帧:

isbn:9787115274496

《Spring攻略(第2版)》以大量的实例，全面透彻地揭示了Spring框架的各项特性以及围绕该框架新推出的许多周边框架，以实际问题-解决方案-具体做法的方式，为读者展示了这一流行框架从基本概念到各种应用，最后到企业集成的各种实际运用，是Spring框架使用者必备的完全指南。《Spring攻略(第2版)》适合于对Java开发和企业应用集成有

一定了解，希望在实际开发中掌握一种全面、快速、可伸缩、可移植的工具平台的开发人员。

作者介绍：

目录: 第1章 Spring简介

1.1 实例化Spring IoC容器

1.1.1 问题

1.1.2 解决方案

1.1.3 工作原理

1.2 配置Spring IoC容器中的Bean

1.2.1 问题

1.2.2 解决方案

1.2.3 工作原理

1.3 调用构造程序创建Bean

1.3.1 问题

1.3.2 解决方案

1.3.3 工作原理

1.4 解决构造程序歧义

1.4.1 问题

1.4.2 解决方案

1.4.3 工作原理

1.5 指定Bean引用

1.5.1 问题

1.5.2 解决方案

1.5.3 工作原理

1.6 为集合元素指定数据类型

1.6.1 问题

1.6.2 解决方案

1.6.3 工作原理

1.7 使用Spring的FactoryBean创建Bean

1.7.1 问题

1.7.2 解决方案

1.7.3 工作原理

1.8 使用工厂Bean和Utility Schema定义集合

1.8.1 问题

1.8.2 解决方案

1.8.3 工作原理

1.9 用依赖检查属性

1.9.1 问题

1.9.2 解决方案

1.9.3 工作原理

1.10 用@Required注解检查属性

1.10.1 问题

1.10.2 解决方案

1.10.3 工作原理

1.11 用XML配置自动装配Bean

1.11.1 问题

1.11.2 解决方案

1.11.3 工作原理

1.12 用@Autowired和@Resource自动装配Bean

1.12.1 问题

1.12.2 解决方案
1.12.3 工作原理
1.13 继承Bean配置
1.13.1 问题
1.13.2 解决方案
1.13.3 工作原理
1.14 从Classpath中扫描组件

1.14.1 问题
1.14.2 解决方案
1.14.3 工作原理
1.15 小结

第2章 高级Spring IoC容器
2.1 调用静态工厂方法创建Bean
2.1.1 问题
2.1.2 解决方案
2.1.3 工作原理
2.2 调用一个实例工厂方法创建Bean
2.2.1 问题
2.2.2 解决方案
2.2.3 工作原理
2.3 从静态字段中声明Bean
2.3.1 问题
2.3.2 解决方案
2.3.3 工作原理
2.4 从对象属性中声明Bean
2.4.1 问题
2.4.2 解决方案
2.4.3 工作原理
2.5 使用Spring表达式语言
2.5.1 问题
2.5.2 解决方案
2.5.3 工作原理
2.6 设置Bean作用域
2.6.1 问题
2.6.2 解决方案
2.6.3 工作原理
2.7 自定义Bean初始化和析构
2.7.1 问题
2.7.2 解决方案
2.7.3 工作原理
2.8 用Java Config简化XML配置
2.8.1 问题
2.8.2 解决方案
2.8.3 工作原理
2.9 使Bean感知容器
2.9.1 问题
2.9.2 解决方案
2.9.3 工作原理
2.10 加载外部资源
2.10.1 问题
2.10.2 解决方案
2.10.3 工作原理
2.11 创建Bean后处理器
2.11.1 问题

- 2.11.2 解决方案
 - 2.11.3 工作原理
 - 2.12 外部化Bean配置
 - 2.12.1 问题
 - 2.12.2 解决方案
 - 2.12.3 工作原理
 - 2.13 解析文本消息
 - 2.13.1 问题
 - 2.13.2 解决方案
 - 2.13.3 工作原理
 - 2.14 使用应用事件进行通信
 - 2.14.1 问题
 - 2.14.2 解决方案
 - 2.14.3 工作原理
 - 2.15 在Spring中注册属性编辑器
 - 2.15.1 问题
 - 2.15.2 解决方案
 - 2.15.3 工作原理
 - 2.16 创建自定义属性编辑器
 - 2.16.1 问题
 - 2.16.2 解决方案
 - 2.16.3 工作原理
 - 2.17 使用TaskExecutor实现并发性
 - 2.17.1 问题
 - 2.17.2 解决方案
 - 2.17.3 工作原理
 - 2.18 小结
- 第3章 Spring AOP和AspectJ支持
- 3.1 启用Spring的AspectJ注解支持
 - 3.1.1 问题
 - 3.1.2 解决方案
 - 3.1.3 工作原理
 - 3.2 用AspectJ注解声明aspect
 - 3.2.1 问题
 - 3.2.2 解决方案
 - 3.2.3 工作原理
 - 3.3 访问连接点信息
 - 3.3.1 问题
 - 3.3.2 解决方案
 - 3.3.3 工作原理
 - 3.4 指定aspect优先级
 - 3.4.1 问题
 - 3.4.2 解决方案
 - 3.4.3 工作原理
 - 3.5 重用切入点定义
 - 3.5.1 问题
 - 3.5.2 解决方案
 - 3.5.3 工作原理
 - 3.6 编写AspectJ切入点表达式
 - 3.6.1 问题
 - 3.6.2 解决方案
 - 3.6.3 工作原理
 - 3.7 在你的Bean中引入行为
 - 3.7.1 问题

3.7.2 解决方案
3.7.3 工作原理
3.8 为你的Bean引入状态
3.8.1 问题
3.8.2 解决方案
3.8.3 工作原理
3.9 用基于XML的配置声明aspect
3.9.1 问题

3.9.2 解决方案
3.9.3 工作原理
3.10 Spring中的AspectJ加载时织入aspect
3.10.1 问题

3.10.2 解决方案
3.10.3 工作原理
3.11 在Spring中配置AspectJ aspect
3.11.1 问题

3.11.2 解决方案
3.11.3 工作原理
3.12 将Spring Bean注入领域对象
3.12.1 问题

3.12.2 解决方案
3.12.3 工作原理
3.13 小结

第4章 Spring中的脚本

4.1 用脚本语言实现Bean
4.1.1 问题
4.1.2 解决方案
4.1.3 工作原理

4.2 将Spring Bean注入脚本中
4.2.1 问题
4.2.2 解决方案
4.2.3 工作原理

4.3 从脚本中刷新Bean
4.3.1 问题
4.3.2 解决方案
4.3.3 工作原理

4.4 定义内联脚本源码
4.4.1 问题
4.4.2 解决方案
4.4.3 工作原理

4.5 小结
第5章 Spring Security

5.1 加强URL访问安全
5.1.1 问题
5.1.2 解决方案
5.1.3 工作原理

5.2 登录到Web应用
5.2.1 问题
5.2.2 解决方案
5.2.3 工作原理

5.3 验证用户
5.3.1 问题
5.3.2 解决方案
5.3.3 工作原理

5.4 做出访问控制决策

5.4.1 问题

5.4.2 解决方案

5.4.3 工作原理

5.5 加强方法调用的安全

5.5.1 问题

5.5.2 解决方案

5.5.3 工作原理

5.6 处理视图中的安全性

5.6.1 问题

5.6.2 解决方案

5.6.3 工作原理

5.7 处理领域对象安全性

5.7.1 问题

5.7.2 解决方案

5.7.3 工作原理

5.8 小结

第6章 将Spring与其他Web框架集成

6.1 在一般Web应用中访问Spring

6.1.1 问题

6.1.2 解决方案

6.1.3 工作原理

6.2 在你的Servlet和过滤器中使用Spring

6.2.1 问题

6.2.2 解决方案

6.2.3 工作原理

6.3 将Spring与Struts 1.x集成

6.3.1 问题

6.3.2 解决方案

6.3.3 工作原理

6.4 将Spring与JSF集成

6.4.1 问题

6.4.2 解决方案

6.4.3 工作原理

6.5 将Spring与DWR集成

6.5.1 问题

6.5.2 解决方案

6.5.3 工作原理

6.6 小结

第7章 Spring Web Flow

7.1 用Spring Web Flow管理简单的UI流程

7.1.1 问题

7.1.2 解决方案

7.1.3 工作原理

7.2 用不同状态类型建立Web流程模型

7.2.1 问题

7.2.2 解决方案

7.2.3 工作原理

7.3 加强Web流程安全

7.3.1 问题

7.3.2 解决方案

7.3.3 工作原理

7.4 持续存储Web流程中的对象

7.4.1 问题

7.4.2 解决方案

7.4.3 工作原理

7.5 将Spring Web Flow与JSF集成

7.5.1 问题

7.5.2 解决方案

7.5.3 工作原理

7.6 使用RichFaces与Spring WebFlow协作

7.6.1 问题

7.6.2 解决方案

7.6.3 方法

7.7 小结

第8章 Spring @MVC

8.1 用Spring MVC开发简单的Web应用

8.1.1 问题

8.1.2 解决方案

8.1.3 工作原理

8.2 用@RequestMapping映射请求

8.2.1 问题

8.2.2 解决方案

8.2.3 工作原理

8.3 用处理器拦截器拦截请求

8.3.1 问题

8.3.2 解决方案

8.3.3 工作原理

8.4 解析用户区域

8.4.1 问题

8.4.2 解决方案

8.4.3 工作原理

8.5 外部化区分区域的文本信息

8.5.1 问题

8.5.2 解决方案

8.5.3 工作原理

8.6 按照名称解析视图

8.6.1 问题

8.6.2 解决方案

8.6.3 工作原理

8.7 视图和内容协商

8.7.1 问题

8.7.2 解决方案

8.7.3 工作原理

8.8 映射异常视图

8.8.1 问题

8.8.2 解决方案

8.8.3 工作原理

8.9 用@Value在控制器中赋值

8.9.1 问题

8.9.2 解决方案

8.9.3 工作原理

8.10 用控制器处理表单

8.10.1 问题

8.10.2 解决方案

8.10.3 工作原理

8.11 用向导表单控制器处理多页表单

8.11.1 问题

8.11.2 解决方案

8.11.3 工作原理

8.12 使用注解(JSR-303)的Bean校验

8.12.1 问题

8.12.2 解决方案

8.12.3 工作原理

8.13 创建Excel和PDF视图

8.13.1 问题

8.13.2 解决方案

8.13.3 工作原理

8.14 小结

第9章 Spring REST

9.1 用Spring发布一个REST服务

9.1.1 问题

9.1.2 解决方案

9.1.3 工作原理

9.2 用Spring访问REST服务

9.2.1 问题

9.2.2 解决方案

9.2.3 工作原理

9.3 发布RSS和Atom信息源

9.3.1 问题

9.3.2 解决方案

9.3.3 工作原理

9.4 用REST服务发布JSON

9.4.1 问题

9.4.2 解决方案

9.4.3 工作原理

9.5 访问具有复杂XML响应的REST服务

9.5.1 问题

9.5.2 解决方案

9.5.3 工作原理

9.6 小结

第10章 Spring和Flex

10.1 Flex入门

10.1.1 问题

10.1.2 解决方案

10.1.3 工作原理

10.2 离开沙箱

10.2.1 问题

10.2.2 解决方案

10.2.3 工作原理

10.3 为应用添加Spring BlazeDS支持

10.3.1 问题

10.3.2 解决方案

10.3.3 工作原理

10.4 通过BlazeDS/Spring暴露服务

10.4.1 问题

10.4.2 解决方案

10.4.3 工作原理

10.5 使用服务器端对象

10.5.1 问题

10.5.2 解决方案

10.5.3 工作原理

10.6 使用BlazeDS和Spring消费面向消息的服务

10.6.1 问题

10.6.2 解决方案

10.6.3 工作原理

10.7 将依赖注入带给你的ActionScript客户

10.7.1 问题

10.7.2 解决方案

10.7.3 工作原理

10.8 小结

第11章 Grails

11.1 获取和安装Grails

11.1.1 问题

11.1.2 解决方案

11.1.3 工作原理

11.2 创建Grails应用

11.2.1 问题

11.2.2 解决方案

11.2.3 工作原理

11.3 Grails插件

11.3.1 问题

11.3.2 解决方案

11.3.3 工作原理

11.4 在Grails环境中开发、生产和测试

11.4.1 问题

11.4.2 解决方案

11.4.3 工作原理

11.5 创建应用的领域类

11.5.1 问题

11.5.2 解决方案

11.5.3 工作原理

11.6 为一个应用的领域类生成CRUD控制器和视图

11.6.1 问题

11.6.2 解决方案

11.6.3 工作原理

11.7 国际化(I18n)信息属性

11.7.1 问题

11.7.2 解决方案

11.7.3 工作原理

11.8 改变永久性存储系统

11.8.1 问题

11.8.2 解决方案

11.8.3 工作原理

11.9 日志

11.9.1 问题

11.9.2 解决方案

11.9.3 工作原理

11.10 运行单元和集成测试

11.10.1 问题

11.10.2 解决方案

11.10.3 工作原理

11.11 使用自定义布局和模板

11.11.1 问题

11.11.2 解决方案

11.11.3 工作原理

11.12 使用GORM查询

11.12.1 问题

11.12.2 解决方案

11.12.3 工作原理

11.13 创建自定义标记

11.13.1 问题

11.13.2 解决方案

11.13.3 工作原理

11.14 小结

第12章 Spring Roo

12.1 设置Spring Roo开发环境

12.1.1 问题

12.1.2 解决方案

12.1.3 工作原理

12.2 创建第一个Spring Roo项目

12.2.1 问题

12.2.2 解决方案

12.2.3 工作原理

12.3 把现有项目导入SpringSource Tool Suite

12.3.1 问题

12.3.2 解决方案

12.3.3 工作原理

12.4 更快地构建更好的应用程序

12.4.1 问题

12.4.2 解决方案

12.4.3 工作原理

12.5 从项目中删除Spring Roo

12.5.1 问题

12.5.2 解决方案

12.5.3 工作原理

12.6 小结

第13章 Spring测试

13.1 用JUnit and TestNG创建测试

13.1.1 问题

13.1.2 解决方案

13.1.3 工作原理

13.2 创建单元测试和集成测试

13.2.1 问题

13.2.2 解决方案

13.2.3 工作原理

13.3 Spring MVC控制器的单元测试

13.3.1 问题

13.3.2 解决方案

13.3.3 工作原理

13.4 管理集成测试中的应用上下文

13.4.1 问题

13.4.2 解决方案

13.4.3 工作原理

13.5 向集成测试注入测试夹具

13.5.1 问题

13.5.2 解决方案

13.5.3 工作原理

13.6 管理集成测试中的事务

13.6.1 问题

13.6.2 解决方案

13.6.3 工作原理

13.7 在集成测试中访问数据库

13.7.1 问题

13.7.2 解决方案

13.7.3 工作原理

13.8 使用Spring的常用测试注解

13.8.1 问题

13.8.2 解决方案

13.8.3 工作原理

13.9 小结

第14章 Spring Portlet MVC框架

14.1 用Spring Portlet MVC开发一个简单的Portlet

14.1.1 问题

14.1.2 解决方案

14.1.3 工作原理

14.2 将Portlet请求映射到处理程序

14.2.1 问题

14.2.2 解决方案

14.2.3 工作原理

14.3 用简单的表单控制器处理portlet表单

14.3.1 问题

14.3.2 解决方案

14.3.3 工作原理

14.4 小结

第15章 数据访问

15.1 Direct JDBC的问题

15.1.1 建立应用数据库

15.1.2 理解数据访问对象设计模式

15.1.3 用JDBC实现DAO

15.1.4 在Spring中配置数据源

15.1.5 运行DAO

15.1.6 更进一步

15.2 使用JDBC模板更新数据库

15.2.1 问题

15.2.2 解决方案

15.2.3 工作原理

15.3 使用JDBC模板查询数据库

15.3.1 问题

15.3.2 解决方案

15.3.3 工作原理

15.4 简化JDBC模板创建

15.4.1 问题

15.4.2 解决方案

15.4.3 工作原理

15.5 在Java 1.5中使用简单的JDBC模板

15.5.1 问题

15.5.2 解决方案

15.5.3 工作原理

15.6 在JDBC模板中使用命名参数

15.6.1 问题

15.6.2 解决方案

15.6.3 工作原理

15.7 在Spring JDBC框架中处理异常

15.7.1 问题
15.7.2 解决方案
15.7.3 工作原理
15.8 直接使用ORM框架的问题

15.8.1 问题
15.8.2 解决方案
15.8.3 工作原理
15.8.4 使用Hibernate API, 用Hibernate XML映射持久化对象
15.8.5 使用Hibernate API, 以JPA注解持久化对象
15.8.6 使用JPA, 以Hibernate为引擎持久化对象
15.9 在Spring中配置ORM资源工厂

15.9.1 问题
15.9.2 解决方案
15.9.3 工作原理
15.10 用Spring ORM模板持久化对象

15.10.1 问题
15.10.2 解决方案
15.10.3 工作原理
15.11 用Hibernate的上下文会话持久化对象

15.11.1 问题
15.11.2 解决方案
15.11.3 工作原理
15.12 用JPA的上下文注入持久化对象

15.12.1 问题
15.12.2 解决方案
15.12.3 工作原理
15.13 小结

第16章 Spring中的事务管理

16.1 事务管理的问题
16.2 选择一个事务管理器实现
16.2.1 问题
16.2.2 解决方案
16.2.3 工作原理

16.3 用事务管理器API编程管理事务
16.3.1 问题
16.3.2 解决方案
16.3.3 工作原理

16.4 用事务模板编程管理事务
16.4.1 问题
16.4.2 解决方案
16.4.3 工作原理

16.5 用事务通知声明式地管理事务
16.5.1 问题
16.5.2 解决方案
16.5.3 工作原理

16.6 用@Transactional注解声明式地管理事务
16.6.1 方法
16.6.2 解决方案
16.6.3 工作原理

16.7 设置事务传播属性
16.7.1 问题
16.7.2 解决方案
16.7.3 工作原理
16.8 设置隔离事务属性

- 16.8.1 问题
 - 16.8.2 解决方案
 - 16.8.3 工作原理
 - 16.9 设置Rollback事务属性
 - 16.9.1 问题
 - 16.9.2 解决方案
 - 16.9.3 工作原理
 - 16.10 设置超时和只读事务属性
 - 16.10.1 问题
 - 16.10.2 解决方案
 - 16.10.3 工作原理
 - 16.11 用加载时织入管理事务
 - 16.11.1 问题
 - 16.11.2 解决方案
 - 16.11.3 工作原理
 - 16.12 小结
- 第17章 EJB、Spring Remoting和Web服务
- 17.1 通过RMI暴露和调用服务
 - 17.1.1 问题
 - 17.1.2 解决方案
 - 17.1.3 工作原理
 - 17.2 用Spring创建EJB 2.x组件
 - 17.2.1 问题
 - 17.2.2 解决方案
 - 17.2.3 工作原理
 - 17.3 在Spring中访问遗留的EJB 2.x组件
 - 17.3.1 问题
 - 17.3.2 解决方案
 - 17.3.3 工作原理
 - 17.4 在Spring中创建EJB 3.0组件
 - 17.4.1 问题
 - 17.4.2 解决方案
 - 17.4.3 工作原理
 - 17.5 在Spring中访问EJB 3.0组件
 - 17.5.1 问题
 - 17.5.2 解决方案
 - 17.5.3 工作原理
 - 17.6 通过HTTP暴露和调用服务
 - 17.6.1 问题
 - 17.6.2 解决方案
 - 17.6.3 工作原理
 - 17.7 选择SOAP Web服务开发方法
 - 17.7.1 问题
 - 17.7.2 解决方案
 - 17.7.3 工作原理
 - 17.8 使用JAX-WS暴露和调用Contract-Last SOAP Web服务
 - 17.8.1 问题
 - 17.8.2 解决方案
 - 17.8.3 工作原理
 - 17.9 定义Web服务契约
 - 17.9.1 问题
 - 17.9.2 解决方案
 - 17.9.3 工作原理
 - 17.10 使用Spring-WS实现Web服务

- 17.10.1 问题
 - 17.10.2 解决方案
 - 17.10.3 工作原理
 - 17.11 使用Spring-WS调用Web服务
 - 17.11.1 问题
 - 17.11.2 解决方案
 - 17.11.3 工作原理
 - 17.12 用XML编组开发Web服务
 - 17.12.1 问题
 - 17.12.2 解决方案
 - 17.12.3 工作原理
 - 17.13 用注解创建服务端点
 - 17.13.1 问题
 - 17.13.2 解决方案
 - 17.13.3 工作原理
 - 17.14 小结
- 第18章 企业中的Spring
- 18.1 将Spring Bean输出为JMX MBean
 - 18.1.1 问题
 - 18.1.2 解决方案
 - 18.1.3 工作原理
 - 18.2 发布和监听JMX通知
 - 18.2.1 问题
 - 18.2.2 解决方案
 - 18.2.3 工作原理
 - 18.3 在Spring中访问远程JMX MBean
 - 18.3.1 问题
 - 18.3.2 解决方案
 - 18.3.3 工作原理
 - 18.4 用Spring电子邮件支持发送邮件
 - 18.4.1 问题
 - 18.4.2 解决方案
 - 18.4.3 工作原理
 - 18.5 用Spring的Quartz支持进行调度
 - 18.5.1 问题
 - 18.5.2 解决方案
 - 18.5.3 工作原理
 - 18.6 用Spring 3.0的调度命名空间进行调度
 - 18.6.1 问题
 - 18.6.2 解决方案
 - 18.6.3 工作原理
 - 18.7 小结
- 第19章 消息
- 19.1 用Spring发送和接收JMS消息
 - 19.1.1 问题
 - 19.1.2 解决方案
 - 19.1.3 工作原理
 - 19.2 转换JMS消息
 - 19.2.1 问题
 - 19.2.2 解决方案
 - 19.2.3 方法
 - 19.3 管理JMS事务
 - 19.3.1 问题
 - 19.3.2 方法

19.3.3 解决方案

19.4 在Spring中创建消息驱动POJO

19.4.1 问题

19.4.2 解决方案

19.4.3 工作原理

19.5 建立连接

19.5.1 问题

19.5.2 解决方案

19.5.3 工作原理

19.6 小结

第20章 Spring Integration

20.1 用EAI集成一个系统到另一个系统

20.1.1 问题

20.1.2 解决方案

20.1.3 工作原理

20.2 使用JMS集成两个系统

20.2.1 问题

20.2.2 解决方案

20.2.3 工作原理

20.3 查询Spring Integration消息得到上下文信息

20.3.1 问题

20.3.2 解决方案

20.3.3 工作原理

20.4 用一个文件系统集成两个系统

20.4.1 问题

20.4.2 解决方案

20.4.3 工作原理

20.5 将消息从一种类型转换为另一种类型

20.5.1 问题

20.5.2 解决方案

20.5.3 工作原理

20.6 使用Spring Integration进行错误处理

20.6.1 问题

20.6.2 解决方案

20.6.3 工作原理

20.7 集成控制分支：分解器和聚合器

20.7.1 问题

20.7.2 解决方案

20.7.3 工作原理

20.8 用路由器实现条件路由

20.8.1 问题

20.8.2 解决方案

20.8.3 工作原理

20.9 使外部系统适应总线

20.9.1 问题

20.9.2 解决方案

20.9.3 工作原理

20.10 用Spring Batch产生事件

20.10.1 问题

20.10.2 解决方案

20.10.3 工作原理

20.11 使用网关

20.11.1 问题

20.11.2 解决方案

20.11.3 工作原理

20.12 小结

第21章 Spring Batch

21.1 建立Spring Batch的基础架构

21.1.1 问题

21.1.2 解决方案

21.1.3 工作原理

21.2 读取和写入(无计算)

21.2.1 问题

21.2.2 解决方案

21.2.3 工作原理

21.3 编写自定义ItemWriter和ItemReader

21.3.1 问题

21.3.2 解决方案

21.3.3 工作原理

21.4 在写入前处理输入

21.4.1 问题

21.4.2 解决方案

21.4.3 工作原理

21.5 通过事务改善生活

21.5.1 问题

21.5.2 解决方案

21.5.3 工作原理

21.6 重试

21.6.1 问题

21.6.2 解决方案

21.6.3 工作原理

21.7 控制步骤异常

21.7.1 问题

21.7.2 解决方案

21.7.3 工作原理

21.8 启动一个作业

21.8.1 问题

21.8.2 解决方案

21.8.3 工作原理

21.9 参数化一个作业

21.9.1 问题

21.9.2 解决方案

21.9.3 工作原理

21.10 小结

第22章 网格上的Spring

22.1 使用Terracotta聚合对象状态

22.1.1 问题

22.1.2 解决方案

22.1.3 工作原理

22.2 将执行分布到网格上

22.2.1 问题

22.2.2 解决方案

22.2.3 方法

22.3 方法的负载平衡

22.3.1 问题

22.3.2 解决方案

22.3.3 方法

22.4 并行处理

22.4.1 问题
22.4.2 解决方案
22.4.3 方法
22.5 在GridGain上部署

22.5.1 问题
22.5.2 解决方案
22.5.3 工作原理
22.6 小结

第23章 jBPM和Spring

软件过程

23.1 理解工作流模型

23.1.1 问题

23.1.2 解决方案

23.1.3 工作原理

23.2 安装jBPM

23.2.1 问题

23.2.2 解决方案

23.2.3 工作原理

23.3 将jBPM4与Spring整合

23.3.1 问题

23.3.2 解决方案

23.3.3 工作原理

23.4 用Spring构建一个服务

23.4.1 问题

23.4.2 解决方案

23.4.3 工作原理

23.5 构建业务过程

23.5.1 问题

23.5.2 解决方案

23.5.3 工作原理

23.6 小结

第24章 OSGi和Spring

24.1 OSGi入门

24.1.1 问题

24.1.2 解决方案

24.1.3 工作原理

24.2 开始使用Spring Dynamic Modules

24.2.1 问题

24.2.2 解决方案

24.2.3 工作原理

24.3 用Spring Dynamic Modules输出服务

24.3.1 问题

24.3.2 解决方案

24.3.3 工作原理

24.4 在OSGi注册表中寻找一个具体服务

24.4.1 问题

24.4.2 解决方案

24.4.3 工作原理

24.5 发布多个接口的一个服务

24.5.1 问题

24.5.2 解决方案

24.5.3 工作原理

24.6 定制Spring Dynamic Modules

24.6.1 问题

24.6.2 解决方案

24.6.3 工作原理

24.7 使用SpringSource dm Server

24.7.1 问题

24.7.2 解决方案

24.7.3 工作原理

24.8 SpringSource的各类工具

24.8.1 问题

24.8.2 解决方案

24.8.3 工作原理

24.9 小结

• • • • • (收起)

[Spring攻略](#) [下载链接1](#)

标签

Spring

编程

Recipes

Programming

软件开发

计算机

计算机科学

编译原理

评论

虽然是厚厚的一本书，究其本质，无非是XML配置而已

东西不够深入， 价格贵。

很棒的手册~

非常好

搭框架搭得丧失信心了。。。

一本Spring相对覆盖全面的问题解决手册

虽然以前看过英文版， 书荒的时候还是免不了去买本二手书看下中文版， 至于翻译嘛，
呵呵~~~~， 何必太在意！

还不错， 深浅适中

当手册放在旁边看也不错， 就是内容有点旧了， 新的内容还得看官方文档

目录看似不错， 基本内涵不多， 如果你想了解Spring的真谛的话可以略过了， 对于这个
价格来说个人感觉不太划算， 收获不了多少知识的赶脚

比较全面的工具书

三周读完1000页全书，对spring整体有了全面的了解，对经常使用的知识有了更全面的认识，没使用过的知识如JMX,JMS也有了一定的了解，每一章都有让我有所收获，适合对spring有一定经验的开发者

这么厚，是人看的么！

[Spring攻略 下载链接1](#)

书评

Java用了一年多了，期间就实习的时候简单接触了一些Struts1，对于框架接触的不多。为了学习Spring也找了几本书，但是看来看去，也没找到合适的，在这里就不吐槽找到的几本国内的技术书籍，尼玛~~~。在本书还是不错的，目前看了一章了，感觉思路还算清晰。推荐大家深度阅读。

[Spring攻略 下载链接1](#)