

<<Spring 3.0就这么简单>>

图书基本信息

书名：<<Spring 3.0就这么简单>>

13位ISBN编号：9787115298393

10位ISBN编号：7115298394

出版时间：2013-1

出版时间：人民邮电出版社

作者：陈雄华 林开雄 著

页数：380

字数：530000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<Spring 3.0就这么简单>>

### 内容概要

《Spring 3.0就这么简单》的主旨就是帮助读者尽快上手，掌握Spring 3.0的核心内容，正确地进行项目实战，同时汲取Spring的思想，并最终将这种思想灵活运用到实际工作中。

《Spring 3.0就这么简单》主要介绍了Spring 3.0的核心内容，不仅讲解了Spring 3.0的基础知识，还深入讨论了Spring IoC容器、Spring AOP、使用Spring JDBC访问数据库、集成Hibernate、Spring的事务管理、Spring MVC、单元测试、敏捷开发技术等内容，帮助读者快速入门并可以立刻使用Spring进行项目实战。本书展示了如何使用Spring自己动手打造服务平台框架，并在本书的最后给出一个开发实战案例。

《Spring 3.0就这么简单》语言简洁，实例丰富，可帮助读者迅速掌握使用Spring 3.0进行开发所需的各种技能。

《Spring 3.0就这么简单》适合于具有一定Java编程基础的读者，以及在Java平台下进行各类软件开发的开发人员和测试人员等。

## <<Spring 3.0就这么简单>>

### 作者简介

陈雄华 毕业于厦门大学计算机与信息工程学院，硕士。  
十年如一日，钻研技术，倾心代码，  
对技术高峰身虽不能至，而心向往之。  
曾负责过十多个大型项目的设计和开发，  
如企业级Java快速开发平台的研发和大型B2B电子商务网络的设计及开发。  
技术开发之余，常将实践所得行诸于文字，已在IBM  
developerworks、天极网、IT168及各报刊杂志发表过数十篇技术文章。  
出版《精通JBuilder2005》、《精通Spring  
3.x》等技术图书，对云计算、Oracle、Web客户端技术感兴趣。

林开雄 毕业于集美大学软件工程专业，资深软件工程师，  
精通Spring等优秀开源技术在企业中的应用，主要研究方向为业务基础平台、BPM、智能报表、分布式等技术。

## &lt;&lt;Spring 3.0就这么简单&gt;&gt;

## 书籍目录

## 第1章 快速入门

## 1.1 Spring概述

## 1.1.1 认识Spring

## 1.1.2 Spring带给我们什么

## 1.1.3 Spring体系结构

## 1.2 实例功能概述

## 1.2.1 比Hello World更适用的实例

## 1.2.2 实例功能简介

## 1.3 环境准备

## 1.3.1 创建库表

## 1.3.2 建立工程

## 1.3.3 类包及Spring配置文件规划

## 1.4 持久层

## 1.4.1 建立领域对象

## 1.4.2 UserDao

## 1.4.3 LoginLogDao

## 1.4.4 在Spring中装配DAO

## 1.5 业务层

## 1.5.1 UserService

## 1.5.2 在Spring中装配Service

## 1.5.3 单元测试

## 1.6 展现层

## 1.6.1 配置Spring MVC框架

## 1.6.2 处理登录请求

## 1.6.3 JSP视图页面

## 1.7 运行Web应用

## 1.8 小结

## 第2章 Spring IoC容器

## 2.1 IoC概述

## 2.2 BeanFactory和ApplicationContext

## 2.2.1 BeanFactory介绍

## 2.2.2 ApplicationContext介绍

## 2.2.3 资源加载

## 2.3 Bean装配

## 2.3.1 Bean基本配置

## 2.3.2 依赖注入

## 2.3.3 注入参数详解

## 2.3.4 Bean作用域

## 2.3.5 基于注解的配置

## 2.3.6 基于Java类的配置

## 2.3.7 不同配置方式比较

## 2.4 小结

## 第3章 Spring AOP

## 3.1 AOP概述

## 3.1.1 AOP到底是什么

## &lt;&lt;Spring 3.0就这么简单&gt;&gt;

- 3.1.2 AOP术语
- 3.2 创建增强类
  - 3.2.1 增强类型
  - 3.2.2 前置增强
  - 3.2.3 后置增强
  - 3.2.4 环绕增强
  - 3.2.5 异常抛出增强
- 3.3 创建切面
  - 3.3.1 切点类型
  - 3.3.2 切面类型
  - 3.3.3 静态普通方法名匹配切面
  - 3.3.4 静态正则表达式方法匹配切面
- 3.4 自动创建代理
  - 3.4.1 实现类介绍
  - 3.4.2 BeanNameAuto ProxyCreator
  - 3.4.3 DefaultAdvisorAuto ProxyCreator
- 3.5 基于@AspectJ配置切面
  - 3.5.1 @AspectJ语法基础
  - 3.5.2 使用前的准备
  - 3.5.3 一个简单的例子
  - 3.5.4 如何通过配置使用@AspectJ切面
  - 3.5.5 不同增强类型
- 3.6 基于Schema配置切面
  - 3.6.1 一个简单切面的配置
  - 3.6.2 配置命名切点
  - 3.6.3 各种增强类型的配置
  - 3.6.4 绑定连接点信息
  - 3.6.5 Advisor配置
- 3.7 各种切面类型总结
- 3.8 小结
- 第4章 使用Spring JDBC访问数据库
  - 4.1 使用Spring JDBC
    - 4.1.1 JdbcTemplate小试牛刀
    - 4.1.2 在DAO中使用JdbcTemplate
  - 4.2 基本的数据操作
    - 4.2.1 更改数据
    - 4.2.2 返回数据库的表自增主键值
    - 4.2.3 批量更改数据
    - 4.2.4 查询数据
    - 4.2.5 查询单值数据
    - 4.2.6 调用存储过程
  - 4.3 BLOB/CLOB类型数据的操作
    - 4.3.1 插入Lob类型的数据
    - 4.3.2 以块数据方式读取Lob数据
    - 4.3.3 以流数据方式读取Lob数据
  - 4.4 其他类型的JdbcTemplate
    - 4.4.1 NamedParameterJdbcTemplate

## &lt;&lt;Spring 3.0就这么简单&gt;&gt;

- 4.4.2 SimpleJdbcTemplate
- 4.5 以OO方式访问数据库
  - 4.5.1 使用MappingSqlQuery查询数据
  - 4.5.2 使用SqlUpdate更新数据
  - 4.5.3 使用StoredProcedure执行存储过程
  - 4.5.4 SqlFunction类
- 4.7 小结
- 第5章 集成Hibernate
  - 5.1 Spring整合ORM技术
  - 5.2 在Spring中使用Hibernate
    - 5.2.1 配置SessionFactory
    - 5.2.2 使用HibernateTemplate
    - 5.2.3 处理LOB类型数据
    - 5.2.4 添加Hibernate事件监听器
    - 5.2.5 使用原生Hibernate API
    - 5.2.6 使用注解配置
    - 5.2.7 事务处理
    - 5.2.8 延迟加载的问题
  - 5.3 DAO层设计
    - 5.3.1 DAO基类的设计
    - 5.3.2 查询接口方法的设计
    - 5.3.3 分页查询接口设计
  - 5.4 小结
- 第6章 Spring的事务管理
  - 6.1 数据库事务基础知识
    - 6.1.1 何为数据库事务
    - 6.1.2 JDBC对事务支持
  - 6.2 Spring对事务管理的支持
    - 6.2.1 事务管理关键抽象
    - 6.2.2 Spring的事务管理器实现类
    - 6.2.3 事务同步管理器
    - 6.2.4 事务传播行为
  - 6.3 编程式的事务管理
  - 6.4 使用XML配置声明式事务
    - 6.4.1 一个将被实施事务增强的服务接口
    - 6.4.2 使用原始的TransactionProxyFactoryBean
    - 6.4.3 基于tx/aop命名空间的配置
  - 6.5 使用注解配置声明式事务
    - 6.5.1 使用@Transactional注解
    - 6.5.2 通过AspectJ LTW引入事务切面
  - 6.6 小结
- 第7章 Spring MVC
  - 7.1 Spring MVC概述
    - 7.1.1 体系结构
    - 7.1.2 配置DispatcherServlet
  - 7.2 注解驱动的控制层
    - 7.2.1 使用@RequestMapping映射请求

## &lt;&lt;Spring 3.0就这么简单&gt;&gt;

- 7.2.2 请求处理方法签名概述
- 7.2.3 处理方法签名详细说明
- 7.2.4 处理模型数据
- 7.3 数据校验
  - 7.3.1 Spring校验框架
  - 7.3.2 Spring MVC数据校验
  - 7.3.3 如何获取校验结果
  - 7.3.4 如何在页面中显示错误
  - 7.3.5 通过国际化资源显示错误信息
- 7.4 视图和视图解析器
  - 7.4.1 认识视图
  - 7.4.2 认识视图解析器
  - 7.4.3 JSP和JSTL
  - 7.4.4 模板视图
  - 7.4.5 输出XML
  - 7.4.6 输出JSON
  - 7.4.7 使用XmlViewResolver
  - 7.4.8 使用ResourceBundleViewResolver
  - 7.4.9 混合使用多种视图技术
- 7.5 本地化解析
  - 7.5.1 本地化概述
  - 7.5.2 使用CookieLocaleResolver
  - 7.5.3 使用SessionLocaleResolver
  - 7.5.4 使用LocaleChangeInterceptor
- 7.6 文件上传
  - 7.6.1 配置MultipartResolver
  - 7.6.2 编写控制器和文件上传表单页面
- 7.7 小结
- 第8章 单元测试
  - 8.1 单元测试概述
    - 8.1.1 为什么需要单元测试
    - 8.1.2 单元测试基本概念
  - 8.2 TestNG快速进阶
    - 8.2.1 TestNG概述
    - 8.2.2 TestNG生命周期
    - 8.2.3 使用TestNG
  - 8.3 模拟利器Mockito
    - 8.3.1 模拟测试概述
    - 8.3.2 创建Mock对象
    - 8.3.3 设定Mock对象的期望行为及返回值
    - 8.3.4 验证交互行为
  - 8.4 测试整合之王Unitils
    - 8.4.1 Unitils概述
    - 8.4.2 集成Spring
    - 8.4.3 集成DbUnit
  - 8.5 使用Unitils测试DAO层
    - 8.5.1 Unitils配置

## &lt;&lt;Spring 3.0就这么简单&gt;&gt;

- 8.5.2 准备测试数据
- 8.5.3 编写测试用例
- 8.7 使用Unitils测试Service层
- 8.8 测试Web层
  - 8.8.1 对LoginController进行单元测试
  - 8.8.2 使用Spring Servlet API模拟对象
  - 8.8.3 使用Spring RestTemplate测试
- 8.9 小结
- 第9章 敏捷开发技术
  - 9.1 敏捷开发概述
    - 9.1.1 敏捷开发原则
    - 9.1.2 敏捷开发过程
  - 9.2 敏捷开发方法Scrum
  - 9.3 测试驱动开发(TDD)实例
  - 9.4 版本管理工具GIT
    - 9.4.1 版本控制意义
    - 9.4.2 SVN
    - 9.4.3 GIT
  - 9.5 代码构建利器Maven
    - 9.5.1 Maven概述
    - 9.5.2 Maven入门
    - 9.5.3 Maven实例
  - 9.6 持续集成Hudson
  - 9.7 小结
- 第10章 自己动手打造服务平台框架
  - 10.1 服务平台概述
    - 10.1.1 SOA实现技术
    - 10.1.2 Web Service技术框架
    - 10.1.3 技术框架的局限
    - 10.1.4 TOP介绍
  - 10.2 快速了解ROP
    - 10.2.1 ROP概述
    - 10.2.2 使用ROP开发一个服务
  - 10.3 请求服务模型
    - 10.3.1 传统Web Service请求模型
    - 10.3.2 ROP请求模型
    - 10.3.3 参数数据绑定与校验
    - 10.3.4 XML和JSON参数绑定
    - 10.3.5 自定义数据转换器
    - 10.3.6 请求服务映射
  - 10.4 应用授权及验证
    - 10.4.1 应用键/应用密钥
    - 10.4.2 应用键/密钥管理器
    - 10.4.3 签名算法
    - 10.4.4 签名功能控制
  - 10.5 服务会话管理
    - 10.5.1 会话管理概述



## &lt;&lt;Spring 3.0就这么简单&gt;&gt;

- 10.5.2 注册会话管理器
- 10.5.3 开发登录和退出服务
- 10.6 错误模型
  - 10.6.1 错误模型概述
  - 10.6.2 系统级主错误编码
  - 10.6.3 系统级子错误编码
  - 10.6.4 业务级子错误编码
- 10.7 响应报文控制
  - 10.7.1 分体式报文模型
  - 10.7.2 响应报文定义
  - 10.7.3 报文输出格式
  - 10.7.4 报文的国际化支持
- 10.8 文件上传
  - 10.8.1 ROP文件上传解决思路
  - 10.8.2 文件上传实例
  - 10.8.3 文件上传控制
- 10.9 服务安全控制
  - 10.9.1 安全控制架构
  - 10.9.2 ServiceAccessController
  - 10.9.3 InvokeTimesController
- 10.10 拦截器及事件体系
  - 10.10.1 拦截器
  - 10.10.2 事件及监听
- 10.11 性能调优
  - 10.11.1 服务平台线程池参数调整
  - 10.11.2 限制服务的占用时长
  - 10.11.3 限制应用/用户的访问
- 10.12 开发客户端SDK
  - 10.12.1 ROP提供了哪些支持
  - 10.12.2 服务开放平台的SDK包
- 10.13 小结
- 第11章 实战案例开发
  - 11.1 景区网站案例概述
    - 11.1.1 景区网站整体功能结构
    - 11.1.2 景区网站用例描述
    - 11.1.3 主要功能流程描述
  - 11.2 系统设计
    - 11.2.1 技术框架选择
    - 11.2.2 Web目录结构及类包结构规划
    - 11.2.3 单元测试类包结构规划
    - 11.2.4 系统的页面交互流程设计
    - 11.2.5 PO类设计
    - 11.2.6 持久层设计
    - 11.2.7 服务层设计
    - 11.2.8 Web层设计
    - 11.2.9 数据库设计
  - 11.3 开发前的准备

## <<Spring 3.0就这么简单>>

- 11.4 持久层开发
  - 11.4.1 PO类
  - 11.4.2 DAO基类
  - 11.4.3 通过扩展基类定义DAO类
  - 11.4.4 DAO Bean的装配
  - 11.4.5 使用Hibernate二级缓存
- 11.5 对持久层进行测试
  - 11.5.1 配置Unitils测试环境
  - 11.5.2 准备测试数据库及测试数据
  - 11.5.3 编写DAO测试基类
  - 11.5.4 编写ViewSpaceDao测试用例
- 11.6 服务层开发
  - 11.6.1 ViewSpaceService的开发
  - 11.6.2 服务类Bean的装配
- 11.7 对服务层进行测试
  - 11.7.1 编写Service测试基类
  - 11.7.2 编写ViewSpaceService测试用例
- 11.8 Web层开发
  - 11.8.1 BaseController的基类
  - 11.8.2 景区网站首页
  - 11.8.3 景区查询
  - 11.8.4 景区详细信息的页面
  - 11.8.5 web.xml配置
  - 11.8.6 Spring MVC配置
- 11.9 对Web层进行测试
  - 11.9.1 编写Web测试基类
  - 11.9.2 编写ViewManageControllerTest测试用例
- 11.10 部署和运行应用
- 11.11 小结

## <<Spring 3.0就这么简单>>

### 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>