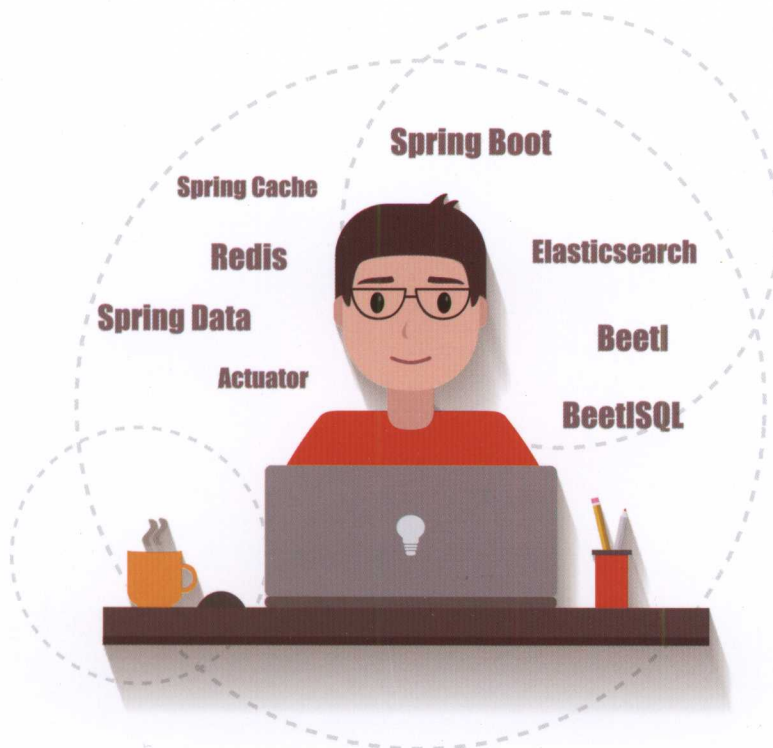


版权注意事项：

- 1、书籍版权归作者和出版社所有
- 2、本PDF仅限用于个人获取知识，进行私底下的知识交流
- 3、PDF获得者不得在互联网上以任何目的进行传播
- 4、如觉得书籍内容很赞，请购买正版实体书，支持作者
- 5、请于下载PDF后24小时内删除本PDF。



Spring Boot 2 精髓

从构建小系统到架构分布式大系统

李家智 著

关于作者



李家智

出生在贵州，毕业于西南民族大学，曾在亚信、中国HP、网易就职，现在东方金科担任架构师。

从事软件开发近20年，致力于Java和Java EE系统的架构和实现，对一切技术充满好奇，以知行合一要求自己。除了本书，也是国内流行开源Beetl模板语言和Dao工具Beetl-SQL的作者。

Spring Boot 2 精髓

从构建小系统到架构分布式大系统

李家智 著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

Spring Boot 是目前 Spring 技术体系中炙手可热的框架之一，既可用于构建业务复杂的企业应用系统，也可以开发高性能和高吞吐量的互联网应用。Spring Boot 框架降低了 Spring 技术体系的使用门槛，简化了 Spring 应用的搭建和开发过程，提供了流行的第三方开源技术的自动集成。

本书系统介绍了 Spring Boot 2 的主要技术，侧重于两个方面，一方面是极速开发一个 Web 应用系统，详细介绍 Spring Boot 框架、Spring MVC、视图技术、数据库访问技术，并且介绍多环境部署、自动装配、单元测试等高级特性；另一方面，当系统模块增加，性能和吞吐量要求增加时，如何平滑地用 Spring Boot 实现分布式架构，也会在本书后半部分介绍，包括使用 Spring 实现 RESTful 架构，在 Spring Boot 框架下使用 Redis、MongoDB、ZooKeeper、Elasticsearch 等流行技术，使用 Spring Session 实现系统水平扩展，使用 Spring Cache 提高系统性能。

阅读本书的人，可以是 Java 新手，从未使用过任何 Spring 技术的工程师。也可以是用过 Spring，但想进一步了解 Spring Boot 的开发者。如果你已经使用过 Spring Boot，那么本书也非常适合你全面深入了解 Spring Boot。

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

Spring Boot 2 精髓: 从构建小系统到架构分布式大系统 / 李家智著. —北京: 电子工业出版社, 2017.10
ISBN 978-7-121-32825-1

I. ①S… II. ①李… III. ①JAVA 语言—程序设计 IV. ①TP312.8

中国版本图书馆 CIP 数据核字 (2017) 第 243885 号

责任编辑: 陈晓猛

印 刷: 三河市双峰印刷装订有限公司

装 订: 三河市双峰印刷装订有限公司

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: 787×980 1/16 印张: 24

字数: 460 千字

版 次: 2017 年 10 月第 1 版

印 次: 2017 年 10 月第 1 次印刷

定 价: 79.00 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，联系及购电话：(010) 88254888, 88258888。

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。

本书咨询联系方式: 010-51260888-819, faq@phei.com.cn。

专家评价

作者集其近二十年来沉浸于技术架构的理论探索和实践经验，特别是近年来扎根于互联网金融机构对于开源技术架构的前沿研究实践，方成此心血之作。全书由易及难、自浅入深，为读者徐徐展开基于 Spring Boot 2.0 构建企业复杂应用的恢弘篇章。此书非常适合作为开发人员及架构师从新手到高手、自低阶至高阶的重要指导书和参考书。

—— 东方资产信息科技部 贺锦

如何快速适应业务的变化发展一直是每个企业 IT 部门都面临的困扰，现在 Spring Boot 开发框架提供了最新的、经过实践验证的技术解决方案。

本书作者，一位近二十年 Java 程序员和架构师，结合他高超的技术能力和丰富的实战经验，给读者精心阐述了 Spring Boot 2.0 从初级快速构建系统到复杂的分布式系统开发的完整开发技术，本书不仅是开发人员不可多得的教科书，对非软件企业的 IT 人员也有极强的参考价值。

—— 东方资产信息科技部 黄友平

过去几年，微服务架构在软件开发领域逐渐深入人心，Spring Boot 在经历了快速演变之后，正在成为 Java 微服务开发的主流成熟框架。本书对 Spring Boot 的特性进行了全方位讲解，辅以大量翔实的案例，对分布式系统开发和应用提供了实战指导。书中还详细介绍了作者倾注了大量心血研发的开源软件 Beeti 和 BeetiSQL，它们易于与 Spring Boot 集成，并已被众多大公司采纳使用。本书对于开发人员和架构师来说，都极具参考价值。

—— 《Kubernetes 权威指南》作者/HPE 高级顾问 龚正

和家智相识多年，他是我所认识的非职业足球运动员中球商最高的，多年来他一直深耕于技术的第一线，有着丰富的技术储备，是我非常钦佩的老大哥。

我们曾经在同一家公司的同一个项目集效力，都非常喜欢踢球，都出了书，不得不觉得很神奇。

这本书的内容非常丰富，也是作者多年潜心钻研的积累，这本书和我的书有一些联系，将 Spring Boot 进行了展开讲解，既有广度，也有深度，非常值得技术人员去学习。

——《微服务那些事儿》作者 纪晓峰

Spring Boot 上手简单、功能丰富、易于扩展，可谓目前业界快速开发、快速生产的利器；然而，它的版本演进非常迅速，中文文档质量也参差不齐。本书由浅入深地讲解了 Spring Boot，帮助读者系统理解 Spring Boot。不仅如此，本书对 2.0 版本带来的新特性亦有非常详尽的描述，绝对值得一读。

——《Spring Cloud 与 Docker 微服务架构实战》作者 周立

和家智相识有五 years 了，最早是因为模板引擎技术结识。家智是国内顶级的模板引擎专家，也是我认识的为数不多的 Spring 技术专家，他在这两方面都曾带给我不同角度的思考与印证，让我获益匪浅。本书是家智二十年技术钻研的一次厚积薄发，其中不仅从作者自身的经验详细讲述了 Spring Boot 框架，还引入了作者在模板引擎、ORM 以及单元测试方面多年研发的开源作品，是 Java 程序员和架构师不可多得的参考资料。

—— ACTFramework 作者 罗格林

十多年前，Spring 颠覆了传统的 Java EE 技术，迎来了 Java 企业级应用开发的春天，然而今天的 Spring Boot 却站在 Spring 巨人的肩膀上，让我们可以更高效地开发与交付。李家智是著名开源框架 Beetl 的作者，他写的《Spring Boot 2 精髓：从构建小系统到架构分布式大系统》一定非常精彩。

——特赞科技 CTO 黄勇

最早熟知作者是从 Beetyl 模板引擎开始的，当时正在寻找一个易用高效的模板引擎，Beetyl 几乎满足了我所有的需要。同时也十分敬佩作者在开源项目上的认真与负责。本书可以说是作者多年的心血所著，从 Spring Boot 的前世今生到使用扩展，都做了非常全面而易懂的概括，细节上也秉承了作者的细致与认真，讲解清晰并语言干练，既适合初学者系统化学习，也适合有经验的工程师作为参考。

—— 开源工具集 Hutool 作者 路小磊

从事 Web 开发有些年头了，经过技术选型，Spring Boot 走进了我们的视野，开箱即用，非常方便，也是目前很多大公司的选择之一。除了研究源码，如果有一本关于 Spring Boot 的指导书籍，既可以方便地解决开发中的问题，又可以帮助读者掌握 Spring Boot，提高生产效率。

家智兄的这本书正是这样不可多得优秀资源，本书是家智兄多年钻研的技术积累，书中详细讲解了 Web 开发的各个知识点，包含 Web 请求处理、ORM 处理、Redis 缓存、MongoDB、Elasticsearch、ZooKeeper、监控等方面的知识点。相信读者在仔细阅读并掌握本书的知识点后，可以极大地提高自身的 Web 开发水平，为读者的软件开发事业助一臂之力！

—— 上海秦苍（买单侠）基础架构组架构师 刘志强

作者在 Java EE 体系内的多年实战经验使得本书的内容极具价值，书中清晰细致地讲解了快速构建 Web 应用系统的各个知识点，尤其是在后端模板引擎和 ORM 两个章节中，作者以自己的两款成熟开源产品 Beetyl 和 BeetylSQL 为切入点进行讲解，剖析角度十分新颖并且有启发性。

通过这本书可以学习到关于 Spring Boot 框架的核心技术，从而掌握快速构建分布式 Web 应用的必备知识。无论你是 Spring Boot 新手，还是已经使用过 Spring Boot 的开发者，相信都可以从这本书中受益。

—— XXL-JOB 系列作者 许雪里

近两年来，随着微服务的兴起，Spring Boot 突然流行起来了，越来越多的公司采用这一技术，其已经成为大多数 Java 微服务开发者的首选开源框架。Spring Boot 有非常显著的特点：配置简单，易于开发，可快速部署。本书结合丰富的实例，从 Spring Boot 的快速开发 Web 应用

入手，逐渐深入地分析 Spring Boot 的高级特性，最后再重点介绍分布式架构的应用。通过深入浅出的阐述，让你从单体应用到分布式、微服务都有全方位的了解，是不可多得的一本好书，当然我认为最重要的还是作者耗费心血的开源项目 Beetl 和 BeetlSQL。

—— 《分布式数据库架构及企业实践——基于 Mycat 中间件》作者，
开源中间件 Mycat 负责人 周继锋

Spring 风靡多年，Spring Boot 在最近几年微服务框架浪潮下更是出尽风头，本书作者由浅入深地把 Spring Boot 2.0 各种特性阐述得淋漓尽致，不管你是 Spring Boot 新手还是老司机都值得一读。Java Web 后端也好，App 后台也罢，甚至独立后台应用，等等，Spring Boot 都是你不可或缺的高效率工具。

移动易项目团队深深的体会就是使用了 Spring Boot 可以节省 50%以上的代码。

—— 上海亿琪软件有限公司 CEO，移动易开源项目负责人，
华为开发者社区专家（HDE） 褚建琪

前言

Java 的各种开发框架发展了很多年，影响了一代又一代的程序员，现在无论是程序员，还是架构师，使用这些开发框架都面临着两方面的挑战。

- 一方面是要快速开发出系统，这就要求使用的开发框架尽量简单，无论是新手还是老手都能快速上手，快速掌握页面渲染、数据库访问等常用技术。也要求开发框架能尽量多地集成第三方工具，以便信手拈来。最后，还希望在开发调试过程中，方便代码更改后能快速重启。
- 另外一方面，当系统模块增加，用户使用量增加时，面对这样的挑战，系统拆分成为新的架构，程序员和架构师当然不希望换掉已有的开发框架，希望能由小而美的系统过渡到大而强的分布式系统。

环顾当前 Java 开源世界中的流行技术框架，能同时胜任这项工作的微乎其微，Play 和 ActFramework 都是不错的选择，国内的 Nutz 和 JFinal 的口碑也不错。但能同时满足快速开发和分布式系统架构的框架，还是群众基础最好、功能最全、基于 Spring 技术的 Spring Boot 框架。

内容介绍

本书系统介绍了 Spring Boot 2.0 的主要技术，侧重于两个方面，一方面是极速开发一个 Web 应用系统（第 1~6 章，包含 Spring 介绍、MVC、视图技术、数据库访问技术），随后介绍了 Spring Boot 的高级特性（第 7~9 章），包括多环境部署、自动装配、单元测试等技术。另外一方面，当系统模块增加，性能和吞吐量要求增加时，如何平滑地用 Spring Boot 来实现分布式架构，会在本书的第 10~17 章介绍。

阅读本书的读者，可以是 Java 新手，从未使用过任何 Spring 技术。也可以是用过 Spring，但想进一步了解 Spring Boot 的开发者。如果你已经使用过 Spring Boot，那么本书也非常适合你全面深入了解 Spring Boot。

希望读者阅读完本书后，既能轻松快速构建 Web 应用系统，也能掌握分布式系统架构的实现。

上半部分介绍 **Spring Boot** 的基础技术。

第 1 章：介绍 Java EE，然后指出其缺点，引入了流行的 Spring，同时也说明 Spring 经过这么多年发展后暴露的一些缺点，从而引出 Spring Boot，并以两个简要例子作为说明。

第 2 章：对 Spring Boot 应用的开发环境做了说明，包括 Java 开发环境的安装和配置，Maven 的安装和配置，设置国内仓库镜像，还有常用的 Maven 命令。本章最后介绍 Spring 历史以及现有开发团队，并介绍 Spring 框架的 AOP 和 IoC 两个核心技术

第 3 章：介绍 MVC 技术，前半部分重点介绍 URL 映射到 Controller，以及映射到 Controller 方法的参数、参数类型转化、参数验证。后半部分简单介绍 MVC 中的视图技术 Freemaker、Beetl，以及 Jackson 序列化技术。Beetl 和 Jackson 将在第 4 章详细介绍。

第 4 章：介绍笔者的开源技术 Beetl 后端模板引擎，作为国内流行的模板引擎之一，具有简单易学、功能/性能强大、支持 MVC 分离开发等特点。另外一部分详细介绍 Jackson 的 JSON 序列化技术。Jackson 不仅作为 Spring MVC 中的 JSON 默认工具，也是 Spring Boot 分布式技术中常采用的 JSON 序列化技术。

第 5 章：介绍以 SQL 为中心的数据库访问工具 BeetlSQL，这是笔者的另外一款流行 Dao 工具，SQL 在 markdown 文件中管理，内置增删改查、轻量级 ORM 功能、代码生成、主从支持、跨多种数据库等特点，适合那些更喜欢以 SQL 方式访问数据库的开发者。

第 6 章：介绍以面向对象为中心的数据库访问工具 Spring Data JPA。本章由易到难，先从 Spring Data 提供的功能入手，介绍如何完成数据库简单的增删改查功能，然后引入 JPA 来解决应用中不可避免的复杂 SQL 查询。

第 7 章：介绍 Spring Boot 高级特性，如常用的 Spring Boot 的配置、日志配置、应用配置的读取、Spring Boot 自动装配技术和 Spring Boot Starter 实现。

第 8 章：介绍如何部署 Spring Boot 应用，包括可执行 jar，以及通过 war 部署到应用服务器上。应用经常面对多个环境，如开发、测试，还有准线上、线上，以及多个 Demo 环境，Spring Boot 提供 Profile 来实现多环境部署。

第 9 章：介绍单元测试概念，以及 Spring Boot 下的单元测试支持，包括 MVC 单元测试、Mock 测试，以及面向数据库应用的测试方案。

下半部分介绍与 **Spring Boot** 相关的分布式技术。

第 10 章：介绍 RESTful 风格的架构，然后介绍 Spring Boot 如何集成以提供 REST 服务，使用 RestTemplate 调用 REST 服务。本章最后也重点介绍了 Swagger 3.0 技术，以方便 REST 的接口的交流、开发和测试。

第 11 章：介绍 MongoDB 的安装和使用，然后介绍 Spring Boot 如何集成 MongoDB，同时还介绍了如何用 MongoTemplate 访问 MongoDB。

第 12 章: 介绍 Redis 服务器的安装和使用, Redis 常用的数据结构和操作命令。然后介绍 Spring Boot 如何集成 Redis, 如何使用 RedisTemplate 来操作 Redis。本章后半部分深入介绍了 RedisTemplate 提供的序列化机制。

第 13 章: 介绍 Elasticsearch 的安装和使用, Elasticsearch 既具有全文搜索功能, 也能像 MongoDB 那样, 具备 NoSQL 的功能。本章介绍通过 REST 和 Spring Data 两种方式访问 Elasticsearch。

第 14 章: 介绍 Spring Boot Cache, 并重点介绍 Redis 作为分布式缓存的实现。在此基础上, 改进了 Redis 分布式缓存, 通过较少的代码实现了一个具备一二级缓存的技术方案。

第 15 章: Spring Boot 应用水平扩展, 需实现无会话状态技术, Spring Session 提供了分布式会话管理, 本章介绍了 Nginx 作为反向代理的内容, 以及 Spring Session 的 Redis 实现及其源码分析。

第 16 章: 基于第 15 章 Spring Boot 应用水平扩展技术必然带来分布式协调要求, ZooKeeper 是一个广泛使用的分布式协调器。本章介绍 ZooKeeper 的安装和使用, 对领导选取、分布式锁和服务注册三个常用功能做了重点描述, 并在 Spring Boot 应用中采用 Curator 来完成这三个功能。

第 17 章: Spring Boot 提供了内置监控功能, 使得用户通过 HTTP 请求就能知晓服务器的健康状态, 如数据源是否可用、NoSQL 服务是否可用、最近的 HTTP 访问的内容等监控信息。本章讲述了线程栈、内存、在线日志、HTTP 访问、RequestMapping 等常用监控功能。其中还讲述通过 dump 线程栈和内存来解决 Spring Boot 应用中的一些性能故障。

如何阅读本书

笔者作为一个从事 Java 开发 17 年的程序员, 这里给新手一些诚恳的建议, 用于帮助新手掌握 Spring Boot 2.0。

建议按照本书每章的例子先模仿一遍, 不要急于按照自己的项目要求去改, 否则很容易知识掌握不牢固、不全面。如果遇到自己暂时无法理解的知识, 也建议优先记住这些知识点。

理解完书中的知识, 能运行书中提供的例子(推荐手写, 或者从官网下载例子)后, 可以尝试主动制造一些错误。看看 Spring Boot 会给你什么样的错误提示, 以 1.4.5 节例子为例, 如果去掉 @RequestMapping 注解, 或者如果写成 `value="/usercredit/{id123}"` 会怎么样, 甚至将 `getCreditLevel` 改成 `getCreditLevelTest` 会有什么后果(这个改动没有任何影响)。通过主动制造错误, 观察 Spring Boot 应用的错误信息来深入学习 Spring Boot。实际上, 不仅仅是学习 Spring Boot, 这也是学习其他框架, 甚至是编程语言或者其他任何编程技术的一种窍门。

本书每章都会提及 Spring Boot 框架的一些接口或者关键类, 不了解这些类的实现细节, 你

仍然可以运用 Spring Boot，如果想深入掌握 Spring Boot，建议阅读这些类的源代码以了解这些类的职责以及如何实现职责。可以通过 IDE 的快捷键打开这些类，以 Eclipse 为例，用快捷键 Ctrl+T 打开这些类去阅读 Spring 源码，还可以在这些类的方法中打上断点，在运行本书例子的时候，查看在断点处发生了什么来帮助你理解 Spring Boot。比如在第 14 章中使用 Redis 实现分布式缓存的时候，提到了 RedisCacheManager，你可以阅读这个类的源码，并在关键的 getCache 方法上打上断点，观察如何实现 Redis 缓存。

如果对于这些类还是无法理解，则可以通过搜索引擎搜索这些类，总有些博客和技术文章在讨论这些类的职责和实现方式。

谨慎对待互联网搜索结果，这是因为 Spring Boot 2.0 技术本身较为新，发展也较快，通过互联网搜索结果需要关注一下文章的发布日期、文章适用的版本，如果你在使用 Spring Boot 2.0 中遇到任何问题，也欢迎到社区进行交流，社区地址是 ibeetl.com。

最后，请购买正版书籍，鼓励作者和出版社出版更好的技术书籍。

致谢

首先感谢我的妻子苗琚对我写书的大力支持，没有她的支持，是不可能完成一本书的写作的。还有我的儿子，知道我正在做一件很重要的事情的时候就不再让我陪他聊天。

还要感谢电子工业出版社的编辑同志，给予我绝对的信任和支持，并对本书的出版做了非常多的指导，感谢你们付出的辛勤汗水。

最后要感谢的是公司和集团领导对我写书的大力支持，特别是在我需要帮助的时候委派左丽娜同事完成部分章节的编写，没有公司领导的支持，对于中国程序员来说，写一本书几乎是不可能完成的任务。

本书是我写的第一本书，由于 Spring 和 Spring Boot 技术体系博大精深，而我技术有限，写作过程中精力也有限，难免有纰漏，敬请读者指正。

东方邦信金融科技有限公司 李家智（闲大赋）

读者服务

轻松注册成为博文视点社区用户 (www.broadview.com.cn)，扫码直达本书页面。

- 下载资源：本书如提供示例代码及资源文件，均可在 [下载资源](#) 处下载。

- **提交勘误：**您对书中内容的修改意见可在 [提交勘误](#) 处提交，若被采纳，将获赠博文视点社区积分（在您购买电子书时，积分可用来抵扣相应金额）。
- **交流互动：**在页面下方 [读者评论](#) 处留下您的疑问或观点，与我们和其他读者一同学习交流。

页面入口：<http://www.broadview.com.cn/32825>



目录

| | |
|---------------------------------|----|
| 第 1 章 Java EE 简介..... | 1 |
| 1.1 Java EE..... | 1 |
| 1.1.1 Java EE 架构..... | 1 |
| 1.1.2 Java EE 的缺点..... | 3 |
| 1.2 Spring..... | 4 |
| 1.2.1 Spring IoC 容器和 AOP..... | 4 |
| 1.2.2 Spring 的缺点..... | 7 |
| 1.3 Spring Boot..... | 8 |
| 1.4 Hello, Spring Boot..... | 9 |
| 1.4.1 创建一个 Maven 工程..... | 10 |
| 1.4.2 增加 Web 支持..... | 10 |
| 1.4.3 Hello Spring Boot 示例..... | 13 |
| 1.4.4 使用热部署..... | 15 |
| 1.4.5 添加 REST 支持..... | 16 |
| 第 2 章 Spring Boot 基础..... | 17 |
| 2.1 检查 Java 环境与安装 Java..... | 17 |
| 2.2 安装和配置 Maven..... | 19 |
| 2.2.1 Maven 介绍..... | 20 |
| 2.2.2 安装 Maven..... | 22 |
| 2.2.3 设置 Maven..... | 23 |
| 2.2.4 使用 IDE 设置 Maven..... | 23 |
| 2.2.5 Maven 的常用命令..... | 24 |
| 2.3 Spring 核心技术..... | 27 |

| | | |
|--------------|---------------------------|-----------|
| 2.3.1 | Spring 的历史..... | 27 |
| 2.3.2 | Spring 容器介绍..... | 28 |
| 2.3.3 | Spring AOP 介绍..... | 33 |
| 第 3 章 | MVC 框架..... | 37 |
| 3.1 | 集成 MVC 框架..... | 38 |
| 3.1.1 | 引入依赖..... | 38 |
| 3.1.2 | Web 应用目录结构..... | 38 |
| 3.1.3 | Java 包名结构..... | 39 |
| 3.2 | 使用 Controller..... | 40 |
| 3.3 | URL 映射到方法..... | 41 |
| 3.3.1 | @RequestMapping..... | 41 |
| 3.3.2 | URL 路径匹配..... | 42 |
| 3.3.3 | HTTP method 匹配..... | 43 |
| 3.3.4 | consumes 和 produces..... | 44 |
| 3.3.5 | params 和 header 匹配..... | 45 |
| 3.4 | 方法参数..... | 46 |
| 3.4.1 | PathVariable..... | 47 |
| 3.4.2 | Model&ModelAndView..... | 48 |
| 3.4.3 | JavaBean 接受 HTTP 参数..... | 50 |
| 3.4.4 | @RequestBody 接受 JSON..... | 52 |
| 3.4.5 | MultipartFile..... | 53 |
| 3.4.6 | @ModelAttribute..... | 55 |
| 3.4.7 | @InitBinder..... | 56 |
| 3.5 | 验证框架..... | 56 |
| 3.5.1 | JSR-303..... | 56 |
| 3.5.2 | MVC 中使用@Validated..... | 58 |
| 3.5.3 | 自定义校验..... | 59 |
| 3.6 | WebMvcConfigurer..... | 60 |
| 3.6.1 | 拦截器..... | 61 |
| 3.6.2 | 跨域访问..... | 62 |
| 3.6.3 | 格式化..... | 63 |
| 3.6.4 | 注册 Controller..... | 64 |

| | | |
|--------------|---------------------------|-----------|
| 3.7 | 视图技术 | 64 |
| 3.7.1 | 使用 Freemarker | 64 |
| 3.7.2 | 使用 Beetyl | 66 |
| 3.7.3 | 使用 Jackson | 67 |
| 3.7.4 | Redirect 和 Forward | 68 |
| 3.8 | 通用错误处理 | 69 |
| 3.9 | @Service 和 @Transactional | 72 |
| 3.9.1 | 声明一个 Service 类 | 72 |
| 3.9.2 | 事务管理 | 73 |
| 3.10 | curl 命令 | 74 |
| 第 4 章 | 视图技术 | 77 |
| 4.1 | Beetyl 模板引擎 | 77 |
| 4.1.1 | 安装 Beetyl | 78 |
| 4.1.2 | 设置定界符号和占位符 | 78 |
| 4.1.3 | 配置 Beetyl | 79 |
| 4.1.4 | groupTemplate | 79 |
| 4.2 | 使用变量 | 80 |
| 4.2.1 | 全局变量 | 80 |
| 4.2.2 | 局部变量 | 81 |
| 4.2.3 | 共享变量 | 81 |
| 4.2.4 | 模板变量 | 82 |
| 4.3 | 表达式 | 82 |
| 4.3.1 | 计算表达式 | 82 |
| 4.3.2 | 逻辑表达式 | 83 |
| 4.4 | 控制语句 | 83 |
| 4.4.1 | 循环语句 | 83 |
| 4.4.2 | 条件语句 | 85 |
| 4.4.3 | try catch | 86 |
| 4.5 | 函数调用 | 87 |
| 4.6 | 格式化函数 | 87 |
| 4.7 | 直接调用 Java | 88 |
| 4.8 | 标签函数 | 89 |