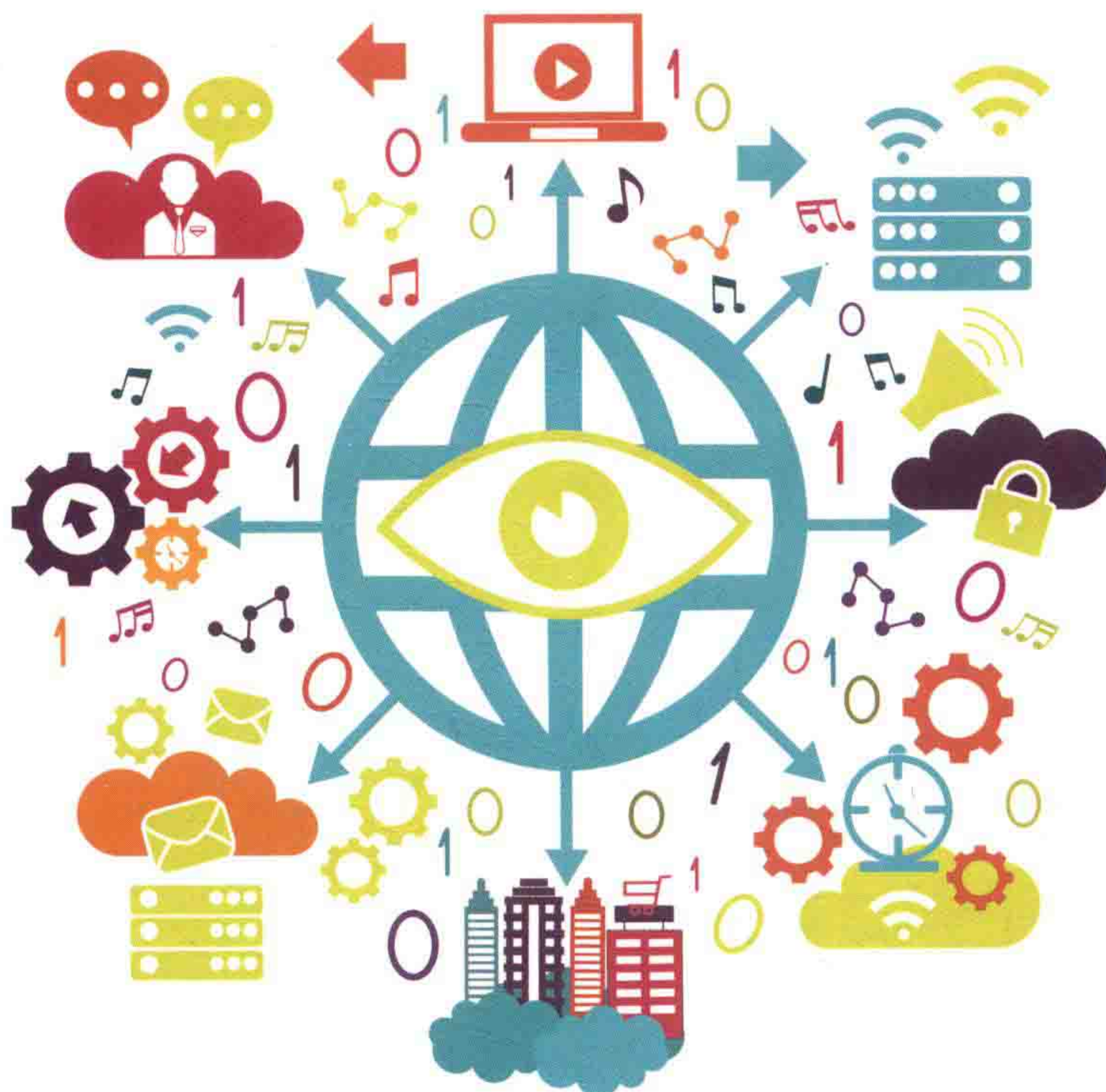


从微服务的架构设计模式和技术造型入手，以Spring Boot 2、Spring Cloud和Docker为构建框架，实现横向可扩展的高可用架构

**Broadview**<sup>®</sup>  
www.broadview.com.cn



# 微服务架构实战

张锋 / 著

洞悉微服务的构建流程，从实战的角度介绍微服务使用的关键框架  
依据敏捷开发的原则，快速迭代，以完整的示例实现整个CI/CD的流程，快速响应需求

 中国工信出版集团

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





# 微服务架构实战

张锋 / 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

目前微服务的现状是概念多于实践，管理和架构人员往往不知道如何落地微服务，本书从软件工程师的角度切入，融入了作者多年的管理及架构经验，内容完全基于实际经验所得，直击痛点。通过阅读本书，开发人员能够实现微服务的快速落地。

全书共 12 章，分为概念篇、开发篇、运维篇和实战篇。概念篇详细阐述微服务的由来和设计要点。开发篇介绍 Spring Boot、Docker 和 Spring Cloud 应用于微服务的案例，并且附有源代码。运维篇从测试、快速开发、质量管理、自动化运维和监控的角度介绍微服务涉及的知识点。实战篇将企业级开发中涉及的内容尽可能详细地列出。

本书不但适合初学者，而且对于团队的管理者及技术选型的架构师也有着非常大的参考意义。

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

微服务架构实战 / 张锋著. —北京: 电子工业出版社, 2018.6  
ISBN 978-7-121-34342-1

I. ①微… II. ①张… III. ①互联网络—网络服务器 IV. ①TP368.5

中国版本图书馆 CIP 数据核字 (2018) 第 120574 号

责任编辑: 陈晓猛

印 刷: 三河市良远印务有限公司

装 订: 三河市良远印务有限公司

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: 25.5

字数: 492 千字

版 次: 2018 年 6 月第 1 版

印 次: 2018 年 6 月第 1 次印刷

定 价: 89.00 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，联系及邮购电话：(010) 88254888, 88258888。

质量投诉请发邮件至 [zlts@phei.com.cn](mailto:zlts@phei.com.cn)，盗版侵权举报请发邮件至 [dbqq@phei.com.cn](mailto:dbqq@phei.com.cn)。

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



## 关于作者

张 锋



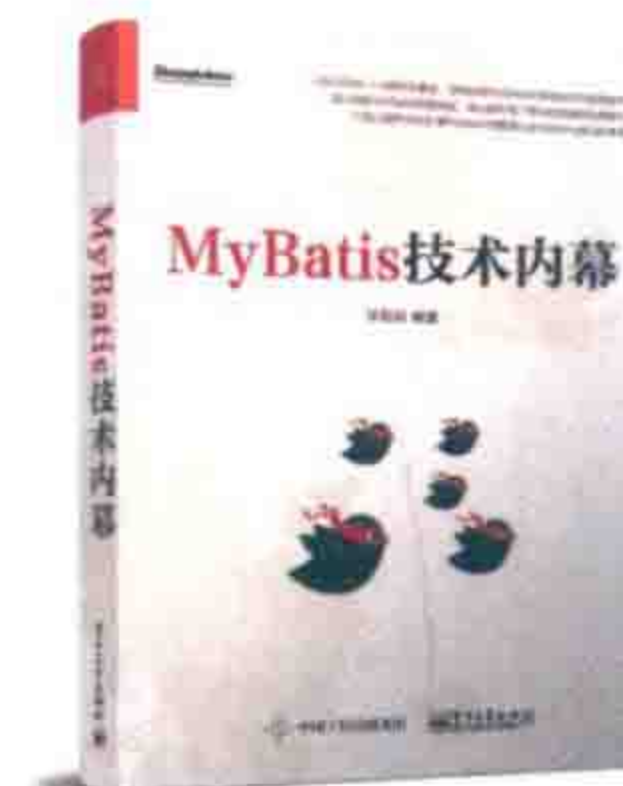
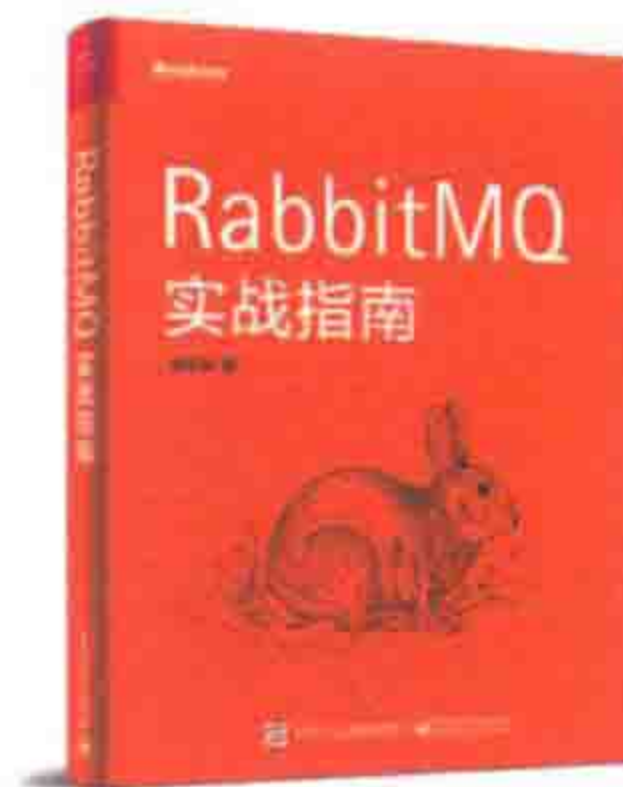
北京航空航天大学软件工程硕士，资深架构师，有十多年管理和架构经验，在业界颇具威望和影响力。

曾就职于神州数据、亚信科技、中文在线及多家互联网公司，担任架构师及技术总监等职位，现就职于中青旅，任架构组组长，曾成功管理和指导三农综合服务信息平台、西北企业云服务平台、省级电信平台及多个互联网平台的架构升级改造。

拥有工信部认证高级信息系统项目管理师资格。

博客园推荐博客，阿里云社区认证专家，腾讯云社区认证专家。

## 好书分享



拒绝堆砌臃肿,支持纯正原创

欢迎投稿: [chenxm@phei.com.cn](mailto:chenxm@phei.com.cn)



# 专家评价

(排名不分先后)

架构的演进是循序渐进并非一蹴而就的，随着时代的更替，我们对于不同时期、阶段的技术发展，也有着不一样的理解和选择。在互联网+和大数据的时代下，如今的软件架构已经从单体、垂直、SOA，发展到微服务架构。本书的作者是业内资深架构师，有着多年的架构从业经验，常年活跃在 Spring Cloud 社区，致力于企业级微服务的解决方案和落地实现。全书以简洁、循序渐进的风格，由浅入深，从易到难，清晰地向读者介绍了微服务的技术思想和整体生态，不仅适合初学者入门，也适合有经验的人员进阶，值得一看。

美团无人配送部

高级技术专家

刘审川 博士

本书以实践为主，内容涵盖了微服务的整个生态，详细解释了实践微服务必须要面对的架构模式。

本书系统地介绍了微服务的设计、开发、运维等方面，结合了 Spring Boot 和 Docker 等热点技术，对微服务的整个生命周期做了全面介绍，适合对微服务实践感兴趣，以及想成为微服务架构师的人员阅读。

当当网

技术总监 穆幽方

为适应市场的变化，企业自身的技术需要不断迭代。如何减少频繁迭代带来的开销成为架构师、开发和运维等人不断思考的问题，“微服务”似乎成为了一个非常好的选择。本书全面地介绍了微服务相关知识。

作者从基本概念谈起，循序渐进地介绍了各种实用技术，无论你处在技术链中的哪个环节，都能找到对应的知识点并快速学习，同时能了解其他环节的工作内容，值得技术爱好者收藏和学习。

中文在线

运维经理 姜楠



随着微服务的兴起，我们可以看到微服务正逐步应用在企业中，与之带来的改变不言而喻，微服务的独立部署、可扩展性架构等特点可以帮助企业提高生产效率、节省成本。本书围绕一个中心，从微服务的概念、架构、管理、应用等方面层层深入，并通过翔实的案例结合实践进行展开，能够助力企业快速实施与部署微服务，值得一读。

柏睿数据科技（北京）有限公司  
产品总监 韩辉辉

在软件工程开发的技术序列里，常常是极强必简。《微服务架构实战》就是这种用简洁的论述解决复杂问题的好书。作者张锋从实践中来到实践中去，将强大的微服务开发武器，用简约易懂的方式展现在读者面前，值得广大求道而解惑的技术人员深入领会。

丽华数据分析引擎工作室  
技术总监 吉更

微服务目前已经成为主流互联网技术架构，在处理复杂业务解耦的问题上屡试不爽。本书作者张锋长期活跃在 Spring Cloud 社区，致力于帮助企业完成微服务改造，并提供咨询服务。本书作为全面的企业微服务架构实践指南，不仅可以帮助初识微服务的开发者完成微服务架构的搭建，也可以帮助正在实践微服务架构的团队解决实际问题。作者长期从事技术类博文的撰写，擅于深入浅出地讲解枯燥晦涩的技术细节，娴熟的文笔在很大程度上可以帮助读者轻松掌握书中的知识点。

北京环宇万维科技有限公司  
技术经理 苏忠亮

当下，越来越多的企业选择与互联网深度融合，基于互联网搭建信息化平台。在平台搭建的过程中，必然要考虑到后期的日常运营带来的爆炸式数据增长所催生出的如高并发、秒级响应速度、分布式部署等一系列问题。微服务的应用正是应对这类严峻考验的极佳解决方案。本书是微服务领域比较完善的实践探索经验的结晶，从微服务的思想基础、设计原则，衍生到 Spring Boot、Docker、Spring Cloud 及其他框架的介绍，不仅涉及微服务的自动化测试与质量管理、自动化部署、日志收集与监控，还提供完整的实战示例，并进行了核心功能推荐的全方位介绍。全书清晰、透彻地剖析了微服务的整个生态，书中还配有許多系统插图，有助于读者快速提升对微服务的认知，构建自己的架构体系。此书具有极高的可读性和应用价值，可谓开卷有益，是践行微服务的上佳选择。

北京亿心宜行汽车技术开发服务有限公司技术部  
高软开发工程师 邓威



# 前言

从分布式服务到 SOA，再到微服务，服务化的脚步一直在不断地前进。正所谓“分久必合，合久必分”，在企业高速发展的今天，单体架构已经很难适应业务的快速变化，微服务的出现，为应对快速变化的业务需求、冗长的开发周期提供了一种新的解决方案。它以模块化的思维应对快速变化的业务需求，使用比如自动化部署、自动化业务监控预警、调用链监控、容器化，以及快速开发等思想加快软件的开发周期，实现更快速、更高质量的交付，整体提高客户的满意度。

## 内容介绍

本书系统地介绍了微服务涉及的各种知识点，横跨软件开发的整个生命周期，采用目前前沿的技术进行知识点的展开。微服务是一个概念，就像 SOA 一样，可能在不同的环境中会产生不同的设计方案。但是总的来说，微服务是为了解决高并发、大数据量的问题而产生的分布式的综合系统解决方案。

本书的内容安排非常有层次感，对于软件开发和从业人员从整体上了解和掌握微服务所需要的知识点进行了全面的梳理。

本书可以分为概念篇、开发篇、运维篇和实战篇。

## 概念篇

首先从概念的角度出发，让读者对微服务的发展有一个感观的了解，然后从设计理念上给出一些建议。

**第 1 章**从微服务的起源和现实业务的角度探讨微服务，使读者能够对微服务有一个感观的认识。

**第 2 章**是针对微服务的设计理念进行整理，包括服务如何拆分、前后端分离、CAP 理论和 CQRS 等，是一个高层次的指导原则。



## 开发篇

开发篇以 Java 中常用的微服务框架 Spring Boot 为基础，介绍 Spring Boot 的快速开发，以及 Docker 技术的基础，并且完成两者的无缝结合。接着对 Spring Cloud 的整体架构进行介绍。

第 3 章详细地介绍 Spring Boot 的开发，包括使用它的优缺点，以及在企业级开发中常用的工具包的整合，包括面向切面编程、Web 开发、文档管理和调度管理，最后结合 Dubbo 完成一个示例性的分布式工程。

第 4 章主要讲解 Docker 的基础操作，介绍微服务中所用到的容器相关的技术，最后给出通用的基于容器的私有云架构。

第 5 章对 Spring Cloud 实现微服务的几个重要框架进行展开描述，让读者了解注册中心、负载均衡、容错、分布式配置、网关和消息总线，能够完成开发层面的微服务架构。

第 6 章对 Spring Cloud 的非核心框架进行介绍，包括 Consul、ZooKeeper、安全框架和数据流框架。

通过对以上几章的了解，读者应该能够从开发的角度基本掌握微服务的开发。

## 运维篇

在微服务中，涉及的不仅仅是开发，还会涉及很多的点，包括运维、测试、监控和日志管理。

第 7 章主要对测试和质量管理的介绍，测试部分包括单元测试、A/B 测试、冒烟和回归测试，质量管理部分主要使用静态代码分析，并且基于 SonarQube 对代码进行静态检查，以及分析代码的总体质量。

第 8 章对微服务的最佳实践 JHipster 进行系统的介绍，并且对 JHipster 部分内容做了处理，将在国内不是很流行的部分进行了处理，尽可能详细地介绍 JHipster 的应用和配置。

第 9 章主要对自动化部署进行介绍，因为微服务的目的不仅仅是简化开发，而且能够提高整个团队的运行效率。所以私服的使用和自动化运维就显得非常重要。

第 10 章主要讲解日志收集和 APM 监控，对于线上系统来说，出现问题的概率还是非常大的，如何快速定位并第一时间找到问题所在的点就显得非常重要。APM 部分对常用的监控工具进行列举，重点介绍 Pinpoint，对使用和邮件告警也进行了重点介绍。

通过以上几章的了解，读者应该能够充分理解 DevOps 的概念，并且了解微服务并不单纯是开发人员的工作，而是整个团队的协同合作。



## 实战篇

第 11 章通过对 PiggyMetrics 的全面讲解,让读者能够了解一个简单的微服务架构所包含的技术点和构建原则,并且实际部署微服务,完成业务的基础操作。

第 12 章对在微服务构建过程中可能涉及的技术点进行讲解,包括 workflow 引擎、规则引擎、调度系统、分布式配置及单点登录。

通过以上几章的学习,读者应该能够在技术选型的过程中扩展思路,了解更多的分布式业务涉及的扩展知识,并且有选择地应用到业务中。

## 勘误和支持

随着技术的进步,微服务的架构也会有不同的演化,就像 Spring 5 的发布一样,为前后端分离进一步奠定了基础。本书中涉及的内容大多是个人的理解和认知,难免有不足之处,所以本书中提及的知识点主要作为抛砖引玉之用,如果有错误之处,还请读者指正。

由于笔者水平有限,且编写时间仓促,书中难免会出现一些错误或者不准确的地方,恳请读者批评指正。如果您有问题或者宝贵意见,欢迎发送邮件至邮箱 [cloudskyme@163.com](mailto:cloudskyme@163.com),期待能够得到您的反馈。

## 致谢

首先要感谢我的家人,没有他们默默的支持,我不可能坚持完成本书的创作。

然后要感谢电子工业出版社博文视点的陈晓猛等编辑对本书内容的校对、勘正及反复核对,感谢你们的付出。

最后感谢在写作过程中给予我帮助的朋友们。

张 锋



----- 读者服务 -----

轻松注册成为博文视点社区用户 ([www.broadview.com.cn](http://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/34342>





# 目录

<b>第 1 章 微服务概述</b> .....	1
1.1 什么是微服务 .....	2
1.2 为什么微服务 .....	3
1.2.1 区别 .....	4
1.2.2 从单体应用说起 .....	5
1.2.3 第一步切分 .....	5
1.2.4 服务化所带来的问题 .....	6
1.2.5 微服务的可扩展性 .....	8
1.2.6 微服务与 SOA 的区别 .....	8
1.3 常见的微服务组件 .....	9
1.4 常用的微服务框架 .....	11
1.4.1 Spring 顶级项目 .....	11
1.4.2 背景 .....	12
1.4.3 社区活跃度 .....	14
1.4.4 架构完整度 .....	16
1.4.5 总结 .....	17
1.5 微服务架构设计模式 .....	17
1.6 如何实施微服务 .....	20
1.7 小结 .....	20
<b>第 2 章 微服务设计原则</b> .....	21
2.1 设计原则之分层架构 .....	22
2.2 设计原则之统一通信协议 .....	24



2.3	设计原则之单一职责 .....	25
2.4	设计原则之服务拆分 .....	26
2.5	设计原则之前后端分离 .....	28
2.6	设计原则之版本控制 .....	29
2.7	设计原则之围绕业务构建.....	30
2.8	设计原则之并发流量控制.....	30
2.9	设计原则之 CAP.....	31
2.10	设计原则之 EDA 事件驱动 .....	32
2.11	设计原则之 CQRS .....	33
2.12	设计原则之基础设施自动化.....	35
2.13	设计原则之数据一致性.....	35
2.14	设计原则之设计模式 .....	36
2.15	设计原则之 DevOps .....	39
2.16	设计原则之无状态服务.....	40
2.17	小结 .....	41
<b>第 3 章</b>	<b>微服务之 Spring Boot .....</b>	<b>42</b>
3.1	一切从简单开始 .....	44
3.2	快速集成第三方的 Starter.....	53
3.3	降低开发复杂度之面向切面.....	53
3.3.1	前置通知 .....	55
3.3.2	后置返回通知 .....	57
3.3.3	后置异常通知 .....	58
3.3.4	后置最终通知 .....	58
3.3.5	环绕通知 .....	59
3.3.6	AOP 总结 .....	60
3.4	并不复杂的持久化 .....	60
3.4.1	单数据源 .....	61
3.4.2	多数据源 .....	63
3.4.3	JOOQ.....	65
3.4.4	事务处理 .....	69
3.4.5	整合 Redis .....	72



3.4.6	整合队列 .....	76
3.4.7	操作 MongoDB .....	83
3.5	Web 开发 .....	85
3.6	懒人的接口文档管理 .....	89
3.7	优化的调度 .....	92
3.8	健康是永恒的主题 .....	94
3.9	强强联合之整合 Dubbo .....	96
3.10	小结 .....	101
<b>第 4 章</b>	<b>微服务之 Docker .....</b>	<b>102</b>
4.1	Docker 原理 .....	104
4.2	更轻量级的虚拟化 .....	105
4.3	三个概念理解 Docker .....	107
4.3.1	镜像 (Image) .....	108
4.3.2	容器 (Container) .....	110
4.3.3	仓库 (Repository) .....	113
4.4	Dockerfile 定制一切 .....	113
4.4.1	Dockerfile 语法 .....	113
4.4.2	Dockerfile 命令 .....	114
4.4.3	Dockerfile 构建过程 .....	117
4.4.4	构建 Java 环境 .....	118
4.4.5	Dockerfile 小结 .....	120
4.5	Docker 网络 .....	121
4.5.1	网络模式 .....	121
4.5.2	link .....	121
4.5.3	跨主机访问 .....	122
4.6	Docker 数据卷 .....	122
4.6.1	数据卷 .....	122
4.6.2	数据卷容器 .....	124
4.7	Spring Boot 与 Docker .....	124
4.8	搭建自己的镜像仓库 .....	128
4.8.1	安装和启动 .....	128



4.8.2	使用 .....	132
4.9	Kubernetes .....	133
4.10	私有云整体架构 .....	136
4.11	小结 .....	137
<b>第 5 章 微服务之 Spring Cloud .....</b>		<b>139</b>
5.1	注册中心 .....	142
5.1.1	常用的注册中心 .....	143
5.1.2	Eureka 介绍 .....	144
5.1.3	服务发现 .....	145
5.1.4	简单注册 .....	147
5.2	负载均衡 .....	152
5.2.1	Spring Cloud 的负载实现 .....	154
5.2.2	Ribbon .....	155
5.2.3	Feign .....	158
5.2.4	加入 core .....	161
5.3	微服务容错 (Hystrix) .....	164
5.3.1	雪崩的形成 .....	164
5.3.2	应对方案 .....	164
5.3.3	降级和熔断 .....	165
5.3.4	Hystrix .....	166
5.3.5	集中监控 .....	170
5.4	分布式配置中心 .....	172
5.5	API 网关 .....	177
5.5.1	为什么需要网关 .....	178
5.5.2	Zuul .....	179
5.6	消息总线 (Spring Cloud Bus) .....	184
5.7	小结 .....	186
<b>第 6 章 微服务之 Spring Cloud 其他框架 .....</b>		<b>187</b>
6.1	Spring Cloud Consul .....	188
6.2	Spring Cloud ZooKeeper .....	190



6.3	Spring Cloud archaius.....	192
6.4	Spring Cloud Task.....	193
6.5	Spring Cloud Security.....	194
6.5.1	HTTP Basic Authentication .....	195
6.5.2	JWT .....	196
6.5.3	OAuth 2 .....	203
6.5.4	Spring Cloud Security.....	204
6.6	Spring Cloud Sleuth.....	205
6.6.1	服务端 .....	206
6.6.2	客户端 .....	207
6.7	Spring Cloud Stream.....	208
6.8	Spring Cloud Data Flow .....	211
6.9	小结 .....	212
<b>第 7 章 微服务之自动化测试与质量管理.....</b>		<b>213</b>
7.1	微服务测试 .....	214
7.2	单元测试 .....	216
7.2.1	单元测试及覆盖率评估 .....	216
7.2.2	JUnit .....	217
7.2.3	Spring Boot 单元测试.....	218
7.2.4	Mockito.....	220
7.3	API 测试.....	222
7.3.1	Jmeter.....	224
7.3.2	压力测试 .....	225
7.4	A/B 测试.....	227
7.5	冒烟和回归测试 .....	228
7.6	静态代码分析 .....	229
7.6.1	Checkstyle .....	230
7.6.2	FindBugs.....	233
7.6.3	PMD .....	234
7.7	SonarQube 质量监控 .....	237
7.7.1	为什么使用 .....	237



7.7.2	安装和使用 .....	238
7.7.3	安装插件 .....	240
7.7.4	运行流程 .....	240
7.8	小结 .....	241
<b>第 8 章</b>	<b>微服务之 JHipster .....</b>	<b>242</b>
8.1	JHipster 技术列表 .....	243
8.1.1	客户端选项 .....	243
8.1.2	服务端选项 .....	245
8.1.3	部署选项 .....	249
8.2	Angular 简介 .....	250
8.3	快速开始 JHipster .....	251
8.3.1	安装 .....	251
8.3.2	使用 .....	252
8.3.3	构建单体应用 .....	253
8.3.4	Entity sub-generator .....	255
8.3.5	开发和运行 .....	258
8.3.6	插件安装 .....	260
8.4	目录结构 .....	260
8.5	构建微服务应用 .....	261
8.5.1	注册中心 .....	261
8.5.2	创建微服务网关 .....	263
8.5.3	Traefik .....	266
8.5.4	JHipster UAA .....	266
8.5.5	构建微服务应用 .....	269
8.6	基础配置 .....	271
8.6.1	JHipster 属性配置 .....	271
8.6.2	作为 Maven 项目 .....	274
8.6.3	数据库 .....	274
8.6.4	DTO .....	275
8.6.5	分页 .....	276
8.6.6	文档 .....	277