

Spring

微服务实战

Spring

Microservices IN ACTION



[美] 约翰·卡内尔 (John Carnell) 著
陈文辉 译
张卫滨 审校

 MANNING

Spring 微服务实战

Spring Microservices IN ACTION

[美] 约翰·卡内尔 (John Carnell) 著

陈文辉 译

张卫滨 审校

人民邮电出版社

北京

图书在版编目 (CIP) 数据

Spring微服务实战 / (美) 约翰·卡内尔
(John Carnell) 著 ; 陈文辉译. — 北京 : 人民邮电出版社, 2018. 6

书名原文: Spring Microservices in Action
ISBN 978-7-115-48118-4

I. ①S… II. ①约… ②陈… 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052548号

版权声明

Original English language edition, entitled *Spring Microservices in Action* by John Carnell published by Manning Publications Co., 209 Bruce Park Avenue, Greenwich, CT 06830. Copyright © 2017 by Manning Publications Co.

Simplified Chinese-language edition copyright © 2018 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由 Manning Publications Co. 授权人民邮电出版社独家出版。未经出版者书面许可, 不得以任何方式复制或抄袭本书内容。

版权所有, 侵权必究。

-
- ◆ 著 [美] 约翰·卡内尔 (John Carnell)
 - 译 陈文辉
 - 审 校 张卫滨
 - 责任编辑 杨海玲
 - 责任印制 马振武

 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路 11 号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
大厂聚鑫印刷有限责任公司印刷

 - ◆ 开本: 800×1000 1/16
印张: 20
字数: 435 千字 2018 年 6 月第 1 版
印数: 1—5 000 册 2018 年 6 月河北第 1 次印刷
- 著作权合同登记号 图字: 01-2017-9013 号
-

定价: 79.00 元

读者服务热线: (010) 81055410 印装质量热线: (010) 81055316

反盗版热线: (010) 81055315

广告经营许可证: 京东工商广登字 20170147 号

内容提要

本书以一个名为 EagleEye 的项目为主线，介绍云、微服务等概念以及 Spring Boot 和 Spring Cloud 等诸多 Spring 项目，并介绍如何将 EagleEye 项目一步一步地从单体架构重构为微服务架构，进而将这个�项目拆分成众多微服务，让它们运行在各自的 Docker 容器中，实现持续集成/持续部署，并最终自动部署到云环境（Amazon）中。针对在重构过程中遇到的各种微服务开发会面临的典型问题（包括开发、测试和运维等问题），本书介绍了解决这些问题的核心模式，以及在实战中如何选择特定 Spring Cloud 子项目或其他工具解决这些问题。

本书适合拥有构建分布式应用程序的经验、拥有 Spring 的知识背景以及对学习构建基于微服务的应用程序感兴趣的 Java 开发人员阅读。对于希望使用微服务构建基于云的应用程序，以及希望了解如何将基于微服务的应用部署到云上的开发人员，本书也具有很好的学习参考价值。

要 要 容 内

致我的兄弟 Jason! 即使在你处在最黑暗的时刻, 你也向我展示了力量和尊严的真正含义。作为兄弟、丈夫和父亲, 你是一个好榜样。

译者序

让我们把时间调回到 2003 年 6 月。那一年，承载着“传统 J2EE 寒冬之后的崭新起点”美好愿景的 Spring 项目开始立项，并以 1.0 版本进行推进。时光荏苒，从 Spring Framework 1.0 发展到现在的 Spring Framework 5.0，Spring 早已从当初 Java 企业级开发领域的挑战者、颠覆者，变成了标准的制定者，成为 Java 企业级开发事实上的标准开发框架。

经过十多年的发展，Spring 家族现在已枝繁叶茂，涵盖 J2EE 开发、依赖维护、安全、批处理、统一数据库访问、大数据、消息处理、移动开发以及微服务等众多领域。在 Spring 家族的诸多项目里面，最耀眼的项目莫过于 Spring Framework、Spring Boot 和 Spring Cloud。Spring Framework 就像是 Spring 家族的树根，是 Spring 得以在 Java 开发领域屹立不倒的根本原因，它的目标就是帮助开发人员开发出好的系统；Spring Boot 就像是树干，它的目标是简化新 Spring 应用的初始搭建以及开发过程，致力于在蓬勃发展的快速应用开发领域成为领导者；Spring Cloud 就如同是 Spring 这棵参天大树在微服务开发领域所结出的硕果。

在近几年，微服务这一概念十分火热，因为它确实能解决传统的单体架构应用所带来的顽疾（如代码维护难、部署不灵活、稳定性不高、无法快速扩展），以至于涌现出了一批帮助实现微服务的工具。在它们之中，Spring Cloud 无疑是最令人瞩目的，不仅是因为 Spring 在 Java 开发中的重要地位，更是因为它提供一整套微服务实施方案，包括服务发现、分布式配置、客户端负载均衡、服务容错保护、API 网关、安全、事件驱动、分布式服务跟踪等工具。

本书对微服务的概念进行了详细的介绍，并介绍了微服务开发过程中遇到的典型问题，以及解决这些问题的核心模式，并介绍了在实战中如何选择特定 Spring Cloud 子项目解决这些问题。本书非常好地把握了理论和实践的平衡，正如本书作者所言，本书是“架构和工程学科之间良好的桥梁与中间地带”。相信读者阅读完本书之后，会掌握微服务的概念，明白如何在生产环境中实施微服务架构，学会在生产中运用 Spring Cloud 等工具，并将项目自动部署到云环境中。

我第一次接触 Spring Cloud，是由于我所负责的一个项目需要从典型的单体应用架构重构成微服务架构，而当时部门主管选定的技术方案就是 Spring Cloud。从那时起，我才真正开始深入了解 Spring Cloud。当时，Spring Cloud 算是比较新的技术，国内有关 Spring Cloud 和微服务方面

的优秀技术书籍凤毛麟角，我只能选择参阅 Spring 的官方文档以及国外的一些技术博客。当时 Manning 出版社尚未出版的 *Spring Microservices in Action* 走进了我的视野。通读完这本书的早期预览版之后，我认为它是目前市面上将微服务和 Spring Cloud 结合介绍得最好的技术书籍，于是我便毛遂自荐，向人民邮电出版社的杨海玲编辑表达了希望成为这本书的中文译者的意愿。不久之后，她回复了我，请我担任这本书的译者，我欣然答应，从此开启了披星戴月的翻译日子。

虽然翻译本书花费了我大量的业余时间，但我也在这个过程中学到了很多。感谢杨海玲编辑和张卫滨老师在翻译过程中对我的指导与指正。同时，我想要感谢我的爱人在这个过程中对我的支持与奉献，还要感谢我那即将出生的孩子，你们是我坚持的动力来源。

限于时间和精力，也囿于我本人的知识积累，在翻译过程中难免犯错。如果读者发现本书翻译中存在哪些不足或纰漏之处，欢迎提出宝贵意见。读者可以通过 memphychan@gmail.com 联系我。希望本书能够对您有用！

陈文辉

2018年4月于东莞

前言

具有讽刺意味的是，在写书的时候，所写的书的最后一部分往往是这本书的前言。这往往也是最棘手的部分。为什么？因为你必须向所有人解释为什么你对一个主题如此热情，以至于你最后花了一年半的时间来写一本关于这个主题的书。很难说清楚为什么有人会花这么多的时间在一本技术书上。人们很少会为名或为利撰写软件开发书籍。

我写本书的原因就是——我热爱编码。这是对我的一种召唤，也是一种创造性的活动，它类似于绘画或演奏乐器。软件开发领域之外的人很难理解这一点。我尤其喜欢构建分布式应用程序。对我来说，看到一个应用程序跨几十个（甚至数百个）服务器工作是一件令人惊奇的事情。这就好像看着一个管弦乐队演奏一段音乐。虽然管弦乐队的最终作品很出色，但完成它往往需要大量的努力与练习。编写大规模分布式应用程序亦是如此。

自从 25 年前我进入软件开发领域以来，我就目睹了软件业与构建分布式应用程序的“正确”方式作斗争。我目睹过分布式服务标准（如 CORBA）兴起与陨落。巨型公司试图推行大型的而且通常是专有的协议。有人记得微软公司的分布式组件对象模型（Distributed Component Object Model, DCOM）或甲骨文公司的 J2EE 企业 Java Bean 2（EJB）吗？我目睹过技术公司和它们的追随者涌向沉重的基于 XML 的模式来构建面向服务的架构（SOA）。

在各种情况下，这些用于构建分布式系统的方法常常在它们自身的负担下崩溃。我并不是说这些技术无法用来构建一些非常强大的应用程序。它们陨落的真相是它们无法满足用户的需求。10 年前，智能手机刚刚被引入市场，云计算还处于起步阶段。另外，分布式应用程序开发的标准和技术对于普通开发人员来说太复杂了，以致于无法在实践中理解和使用。在软件开发行业，没有什么能像书面代码那样说真话。当标准妨碍到这一点时，标准很快就会被抛弃。

当我第一次听说构建应用程序的微服务方法时，我是有点儿怀疑的。“很好，另一种用于构建分布式应用的银弹方法。”我是这样想的。然而，随着我开始深入了解这些概念，我意识到微服务的简单性可以成为游戏规则的改变者。微服务架构的重点是构建使用简单协议（HTTP 和 JSON）进行通信的小型服务。仅此而已。开发人员可以使用几乎任何编程语言来编写一个微服务。在这种简单中蕴含着美。

然而，尽管构建单个微服务很容易，实施和扩展它却很难。要让数百个小型的分布式组件协同工作，然后从它们构建一个弹性的应用程序是非常困难的。在分布式计算中，故障是无从逃避的现实，应用程序要处理好故障是非常困难的。套用我同事 Chris Miller 和 Shawn Hagwood 的话：“如果它没有偶尔崩溃，你就不是在构建。”

正是这些故障激励着我写这本书。我讨厌在不必要的时候从头开始构建东西。事实上，Java 是大多数应用程序开发工作的通用语言，尤其是在企业中。对许多组织来说，Spring 框架已成为大多数应用程序事实上的开发框架。我已经用 Java 做了近 20 年的应用程序开发（我还记得 Dancing Duke applet），并且使用 Spring 近 10 年了。当我开始我的微服务之旅时，我很高兴看到 Spring Cloud 的出现。

Spring Cloud 框架为许多微服务开发人员将会遇到的常见开发和运维问题提供开箱即用的解决方案。Spring Cloud 可以让开发人员仅使用所需的部分，并最大限度地减少构建和部署生产就绪的 Java 微服务所需的工作量。通过使用其他来自 Netflix、HashiCorp 以及 Apache 基金会等公司和组织的久经考验的技术，Spring Cloud 实现了这一点。

我一直认为自己是一名普通的开发人员，在一天结束的时候，需要按期完成任务。这就是我写这本书的原因。我想要一本可以在我的日常工作中使用的书。我想要一些直接简单的（希望如此）代码示例。我总是想要确保本书中的材料既可以作为单独的章使用也可以作为整体来使用。我希望读者会觉得本书很有用，希望读者会喜欢读它，就如同我喜欢写它一样。

资源与支持

本书由异步社区出品，社区（<https://www.epubit.com/>）为您提供相关资源和后续服务。

配套资源

本书提供如下资源：

- 本书源代码；
- 书中彩图文件；

要获得以上配套资源，请在异步社区本书页面中点击 **配套资源**，跳转到下载界面，按提示进行操作即可。注意：为保证购书读者的权益，该操作会给出相关提示，要求输入提取码进行验证。

提交勘误

作者和编辑尽最大努力来确保书中内容的准确性，但难免会存在疏漏。欢迎您将发现的问题反馈给我们，帮助我们提升图书的质量。

当您发现错误时，请登录异步社区，按书名搜索，进入本书页面，点击“提交勘误”，输入勘误信息，点击“提交”按钮即可。本书的作者和编辑会对您提交的勘误进行审核，确认并接受后，您将获赠异步社区的 100 积分。积分可用于在异步社区兑换优惠券、样书或奖品。

详细信
写书评
提交勘误

页码: 页内位置 (行数): 勘误次数:

B / U / / /

 /

 / / / 文字颜色

提交

扫码关注本书

扫描下方二维码，您将会在异步社区微信服务号中看到本书信息及相关的服务提示。



与我们联系

我们的联系邮箱是 contact@epubit.com.cn。

如果您对本书有任何疑问或建议，请您发邮件给我们，并在邮件标题中注明本书书名，以便我们更高效地做出反馈。

如果您有兴趣出版图书、录制教学视频，或者参与图书翻译、技术审校等工作，可以发邮件给我们；有意出版图书的作者也可以到异步社区在线提交投稿（直接访问 www.epubit.com/selfpublish/submission 即可）。

如果您是学校、培训机构或企业，想批量购买本书或异步社区出版的其他图书，也可以发邮件给我们。

如果您在网上发现有针对异步社区出品图书的各种形式的盗版行为，包括对图书全部或部分内容的非授权传播，请您将怀疑有侵权行为的链接发邮件给我们。您的这一举动是对作者权益的保护，也是我们持续为您提供有价值的内容的动力之源。

关于异步社区和异步图书

“异步社区”是人民邮电出版社旗下 IT 专业图书社区，致力于出版精品 IT 技术图书和相关学习产品，为作译者提供优质出版服务。异步社区创办于 2015 年 8 月，提供大量精品 IT 技术图书和电子书，以及高品质技术文章和视频课程。更多详情请访问异步社区官网 <https://www.epubit.com>。

“异步图书”是由异步社区编辑团队策划出版的精品 IT 专业图书的品牌，依托于人民邮电出版社近 30 年的计算机图书出版积累和专业编辑团队，相关图书在封面上印有异步图书的 LOGO。异步图书的出版领域包括软件开发、大数据、AI、测试、前端、网络技术等。



异步社区



微信服务号

致谢

当我坐下来写下这些致谢时，我忍不住回想起 2014 年我第一次跑马拉松的情景。写一本书就如同跑马拉松。写这本书的提案和大纲很像训练过程，它会让你的想法成型，会把你的注意力集中在未来的事情上。是的，在这个过程中接近尾声的时候，它可能会变得有点乏味和残酷。

开始写这本书的那一天就像是比赛日。你充满活力与激情地开始了马拉松比赛。你知道你在尝试做比以往所有事情都要重大的事情，这既令人兴奋又让人神经紧张。虽然这是你已经训练过的，但同一时间，在你的脑海中总会有一些怀疑的声音，说你完成不了你开始的事情。

我从跑步中学到了比赛不是一次一公里地完成的，相反，跑步是一只脚在另一只脚前面地跑。长跑是个人脚步的总和。当我的孩子们在为某件事而挣扎时，我笑着问他们：“你如何写一本书？那就是一次一词，一次一步。”他们通常会不以为然，但到最后，除了这条无可争辩的铁律之外就没有别的办法了。

然而，当你跑马拉松的时候，你可能是一名竞赛的参与者，但你永远不会孤军奋斗。在这个过程中，有整个团队在那里为你提供支持、时间和建议。撰写这本书的经历亦是如此。

我首先想感谢 Manning 出版社为我撰写这本书所给予的支持。策划编辑 Greg Wild 耐心地与我一直工作，帮助我精炼了本书中的核心概念，并引导我完成了整个提案过程。在此过程中，我的开发编辑 Maria Michaels 让我保持坦诚，并鞭策我成为一名更优秀的作者。我还要感谢我的技术编辑 Raphael Villela 和 Joshua White，他们不断检查我的工作，并确保我编写的示例和代码的整体质量。我非常感谢这些人在整个项目中投入的时间、才华和承诺。我还要感谢在撰写和开发过程中对书稿提供反馈意见的审稿人：Aditya Kumar、Adrian M. Rossi、Ashwin Raj、Christian Bach、Edgar Knapp、Jared Duncan、Jiri Pik、John Guthrie、Mirko Bernardoni、Paul Balogh、Pierluigi Riti、Raju Myadam、Rambabu Posa、Sergey Evsikov 和 Vipul Gupta。

致我的儿子 Christopher：你正成长为一个不可思议的年轻人。我迫不及待地想要到你真正点燃热情的那一天，因为这个世界上没有什么能阻止你实现你的目标。

致我的女儿 Agatha：我愿付出我所有的财富来换取用仅仅 10min 的时间透过你的眼睛去看这个世界。这段经历会让我成为一名更好的作家，更重要的是，成为一个更好的人。你的智慧，

你的观察力和创造力让我谦逊。

致我4岁的儿子 Jack：小伙子，感谢你在我每次说“我现在不能玩，因为爸爸必须要投身于这本书的写作”的时候对我保持耐心。你总是逗我笑，让整个家庭变得完整。没有什么比我看到你是一个“开心果”并与家里的每个人一起玩耍更让我开心的了。

我和这本书的赛跑已经完成了。就像我的马拉松一样，我在写这本书时已倾尽全力。我非常感激 Manning 团队，以及早早地买下了这本书并给予我许多珍贵反馈的 MEAP 版读者。最后，我希望读者喜欢这本书，就像我喜欢写这本书一样。谢谢你！

关于本书

本书是为工作中的 Java/Spring 开发人员编写的，他们需要实际的建议以及如何构建和实施基于微服务的应用程序的示例。写这本书的时候，我希望它基于与 Spring Boot 和 Spring Cloud 示例结合的核心微服务模式，这些示例演示了这些模式。因此，读者会发现几乎每一章都会讨论特定的微服务设计模式，以及使用 Spring Boot 和 Spring Cloud 实现的模式示例。

本书适合的读者

- 拥有构建分布式应用程序经验（1~3 年）的 Java 开发人员。
- 拥有 Spring 的知识背景（1 年以上）的技术人员。
- 对学习构建基于微服务的应用程序感兴趣的技术人员。
- 对使用微服务构建基于云的应用程序感兴趣的技术人员。
- 想要知道 Java 和 Spring 是否是用于构建基于微服务的程序的相关技术的技术人员。
- 有兴趣了解如何将基于微服务的程序部署到云上的技术人员。

本书组织结构

本书包含 10 章和 2 个附录。

- 第 1 章会介绍微服务架构为什么是构建应用程序，尤其是基于云的应用程序的重要相关方法。
- 第 2 章将引导读者了解如何使用 Spring Boot 构建第一个基于 REST 的微服务。这一章将介绍如何通过架构师、应用工程师和 DevOps 工程师的角度来审视微服务。
- 第 3 章会介绍如何使用 Spring Cloud Config 管理微服务的配置。Spring Cloud Config 可帮助开发人员确保服务的配置信息集中在单个存储库中，并且在所有服务实例中都是版本控制和可重复的。

- 第 4 章介绍第一个微服务路由模式——服务发现。在这一章中，读者将学习如何使用 Spring Cloud 和 Netflix 的 Eureka 服务，将服务的位置从客户的使用中抽象出来。
- 第 5 章讨论在一个或多个微服务实例关闭或处于降级状态时保护微服务的消费者。这一章将演示如何使用 Spring Cloud 和 Netflix Hystrix（和 Netflix Ribbon）来实现客户端调用的负载均衡、断路器模式、后备模式和舱壁模式。
- 第 6 章会介绍微服务路由模式——服务网关。使用 Spring Cloud 和 Netflix 的 Zuul 服务器，开发人员将为所有微服务建立一个单一入口点。我们将讨论如何使用 Zuul 的过滤器 API 来构建可以针对流经服务网关的所有服务强制执行策略。
- 第 7 章介绍如何使用 Spring Cloud Security 和 OAuth2 实现服务验证和授权。我们将介绍如何设置 OAuth2 服务来保护服务，以及如何在 OAuth2 实现中使用 JSON Web 令牌（JSON Web Tokens, JWT）。
- 第 8 章讨论如何使用 Spring Cloud Stream 和 Apache Kafka 将异步消息传递到微服务中。
- 第 9 章介绍如何使用 Spring Cloud Sleuth 和 Open Zipkin 来实现日志关联、日志聚合和跟踪等常见日志记录模式。
- 第 10 章是本书的基石项目。读者将使用在本书中构建的服务，并将其部署到亚马逊弹性容器服务（Amazon Elastic Container Service, ECS）。我们还将讨论如何使用 Travis CI 等工具自动化构建和部署微服务。
- 附录 A 介绍如何设置桌面开发环境，以便可以运行本书中的所有代码示例。本附录介绍本地构建过程是如何工作的，以及想要在本机运行代码示例时如何本地启动 Docker。
- 附录 B 是 OAuth2 的补充资料。OAuth2 是一种非常灵活的身份验证模型，这一附录简要介绍 OAuth2 可用于保护应用程序及其相应的微服务的不同方式。

关于代码

本书每一章中的所有代码示例都可以在作者的 GitHub 存储库中找到，每一章都有自己的存储库。读者可以通过到每一章代码存储库的链接 <http://github.com/carnellj/spmia-overview> 找到概述页面。包含所有源代码的 zip 文件也可从 Manning 出版社的网站^①获取。

本书中的所有代码使用 Maven 作为主要构建工具进行构建以运行在 Java 8 上。有关编译和运行代码示例所需的软件工具的完整详细信息，参见附录 A。

我在写这本书时遵循的一个核心概念是，每章中的代码示例应该独立于其他章中的代码示例。因此，我们为某一章创建的每个服务将构建到相应的 Docker 镜像。当使用前几章的代码时，它包括在源代码和已构建的 Docker 镜像中。我们使用 Docker compose 和构建的 Docker 镜像来保证每章都具有可重现的运行时环境。

① 读者可登录异步社区（<https://www.epubit.com>），在本书页面免费下载。——编者注

本书包含许多源代码的例子，它们有的在带编号的代码清单中，有的在普通的文本中。在这两种情况下，源代码都以等宽字体印刷，以将其与普通文本分开。有时，代码还会加粗，以突出显示与这一章前面的步骤相比有变化的代码，例如，将新功能添加到现有代码行时。

在很多情况下，原始的源代码已被重新调整了格式。我们添加了换行符和重新加工了缩进，以适应书的页面空间。在极少数情况下，甚至还不止如此，代码清单还包括行连续标记（**—**）。此外，在文本中描述代码时，源代码中的注释通常会从代码清单中移除。许多代码清单附带了代码注解，突出重要的概念。

关于作者

约翰·卡内尔 (John Carnell) 在 Genesys 的 PureCloud 部门工作，担任 Genesys 的高级云工程师。他大部分时间都在使用 AWS 平台构建基于电话的微服务。他的日常工作主要围绕在设计和构建跨 Java、Clojure 和 Go 等多种技术平台的微服务。

他是一位高产的演讲者和作家。他经常在当地的用户群体发表演讲，并且是“The No Fluff Just Stuff Software Symposium”的常规发言人。在过去的 20 年里，他撰写、合著了许多基于 Java 的技术书籍，并担任了许多基于 Java 的技术书籍和行业刊物的技术审稿人。

他拥有马奎特大学 (Marquette University) 艺术学士学位和威斯康星大学奥什科什分校 (University of Wisconsin Oshkosh) 工商管理硕士 (MBA) 学位。

他是一位充满激情的技术专家，他不断探索新技术和编程语言。不演讲、写作或编码时，他与妻子 Janet 和 3 个孩子 (Christopher、Agatha 和 Jack) 生活在在北卡罗来纳州的卡里。

在极少的空闲时间里，他喜欢跑步、与他的孩子嬉戏，并研究菲律宾武术。

读者可以通过 john_carnell@yahoo.com 与他取得联系。