

- 版权注意事项：**
- 1、书籍版权归著者和出版社所有；**
 - 2、本PDF仅用于个人获取知识，进行私底下知识交流；**
 - 3、PDF获得者不得在互联网以任何目的进行传播；**
- 如有需要，请尽量购买正版实体书！支持书籍作者！！**



架构探险 轻量级微服务架构 下册

黄勇 著

关于作者



黄勇

现任上海特赞科技CTO，曾任阿里巴巴公司系统架构师。具有丰富的互联网产品架构经验与技术管理经验，擅长敏捷开发模式，推崇“轻量级”系统架构。国内开源软件推动者，Smart开源框架创始人，图书《架构探险：从零开始写Java Web框架》作者。技术大会讲师、培训师、咨询师。热爱技术交流，乐于分享自己的成长经验。

计算机系列

架构探险 轻量级微服务架构

可与你相处

下册

黄勇 著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书将重点关注微服务基础设施方面，其中大部分内容涉及微服务运维相关技术。全书以实践的角度进行编写，读者首先将学习轻量级微服务架构的全景视图，随后的各个章节将围绕微服务的日志、监控、通信、解耦、测试、配置六大方面进行展开。读者可亲自动手，从零开始搭建轻量级微服务架构，充分享受架构探险的乐趣。

本书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。

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

架构探险：轻量级微服务架构. 下册 / 黄勇著. —北京：电子工业出版社，2017.9
ISBN 978-7-121-32447-5

I. ①架… II. ①黄… III. ①互联网络—网络服务器—研究 IV. ①TP368.5

中国版本图书馆 CIP 数据核字（2017）第 191711 号

责任编辑：陈晓猛

印 刷：中国电影出版社印刷厂

装 订：中国电影出版社印刷厂

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：21.25

字数：408 千字

版 次：2017 年 9 月第 1 版

印 次：2017 年 9 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，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@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。

推荐序

微服务，我们如何与你相处

微服务来了，有了“服务”这两个字，这注定又是个一说就明白、一举例就糊涂、一讨论就吵架的概念。微服务的出现有其必然的商业背景和架构哲学，如何更好地认识微服务的内涵、如臂使指地应用微服务架构，还是有着很多挑战的，这也许就是本书被命名为“架构探险”的原因。

企业数字化转型驱动架构升级

互联网经济深刻改变了我们身边的商业环境，消费者的生活方式日益数字化，人们可以在任何时间、任何地点利用线上、线下渠道体验无缝购物，运用社交媒体表达自我，企业也在运用多种技术手段，发挥数字化潜力，改善客户联系，促进企业业务模式的转型。Gartner 认为，数字化就是把人、事、物和商业联系起来，建立新的商业模式。未来的企业都将是 IT 企业，IT 将从后台走向前台，从 ERP、CRM 等内部流程优化为主的业务，逐步转向内外兼修的模式，从而实现商业创新。

这一变化要求 IT 架构更加灵活地与上下游企业协作，更加快速地响应客户的个性化需求，更加弹性地应对无时不在的客户请求并提供良好的客户体验，同时云计算、大数据等技术的出现也为上述改变提供了新的技术选择，我们正面临 B/S 多层架构出现后新的一次架构升级，而微服务架构就是在这个架构升级过程中应运而生的。

分而治之的哲学是微服务的理论基础

把大的问题分解为容易解决的小问题，找到小问题的解决办法，再来解决大问题，这就是分而治之的哲学。正如万事万物由分子、原子组成一样，软件也可以分解为基本单元，以这样的基本单元进行开发、测试、维护，是解决大规模系统建设的思路。分而治之首先要解决如何分的问题，企业软件的分法应该是以业务驱动的，而不是以技术驱动的，也就是分解为独立的

业务逻辑，而这样的不可再分的业务逻辑就是微服务。

凡事有一利必有一弊，细分为微服务后，势必带来部署、测试、信息集成难度的提高，分而治之除了“分”，还需要“治”。传统恐龙型 ERP 是一个面向组织的软件，完备、复杂、响应变化慢，适合业务稳定的情况，而在数字化时代，客户个性化的要求让我们从这种面向组织的软件逐渐演变为面向个体的软件。例如，从前的 EHR 软件是为人力资源部门服务的，整体开发、整体实施，而现在我们会从个体的角度规划软件，可以先从招聘专员开始做一个面试管理的流程，逐步推出新的流程，完善现有的流程。这些面向个体的流程就是微应用，企业应用将由无数个微应用组成。微服务则是一个技术概念，能更好地解决微应用的技术实现问题，是一个事物的不同侧面，所谓“横看成岭侧成峰，远近高低各不同”，微服务和微应用是事物的一体两面。正因为微服务实际就是一个业务逻辑，因此做好微服务需要从微应用的维度考虑，将分解开的逻辑形成一个整体，要从多渠道接入、客户体验、数据管理、应用交付、运维全方位的视角考虑，这就是分而治之中实现“治”的体验，也是微服务架构需要解决的问题。

站在 SOA 的肩膀上践行微服务

微服务是一个新概念，但这绝不是一个全新架构，更不是一个包治百病的架构。由于有服务二字，很容易让人联想到面向服务架构（SOA），其实微服务架构属于应用技术架构，和以 B/S 为代表的三层架构相对应，强调将巨石型应用拆分为由微服务组成的应用，在数据上也视情况从集中的存储拆解为更小的存储单元。而 SOA 属于企业架构的范畴，从企业架构出发把业务分解为不同领域的服务，不同物理系统提供不同服务，注重系统之间通过服务互联互通的规范，对服务如何实现并不关注。因此，面向服务架构的服务应该是一个业务意义的服务，而微服务是系统中的技术服务，更关注服务的实现，虽然提供了业务意义的服务，但是不能混为一谈。微服务使用也不是无限度的，事实上由于数据一致性等问题的限制，不能无限度拆分微服务，因此可以把微服务分为系统对外提供的远程服务、系统内部的远程服务和系统内部的本地服务，显式声明、明确职责。事实上，在企业架构上使用 SOA 支撑业务，而在应用技术架构上使用微服务架构，是一个合适的选择。

黄柳青博士是我和黄勇共同的导师，他在 2004 年所著的《软件的涅槃》一书中指出：“互联网时代的企业应用定义，正发生革命性的变化……横向的部门互动、实时的企业间互动、多样的交互渠道、灵活的业务规则，使得原有意义上的独立应用不复存在……对软件设计者来说，能直观地分割并具有最小内部耦合的软件结构是简约之美……美的软件是软件企业与软件开发者的终极目标”，那时候他把这种全新的软件生产模式称为“面向构件”。回头看来，微服务正是“面向构件”在数字化时代的解读，用微服务架构实现软件之美，加速企业数字化转型。

专家推荐

(排名不分先后)

SOA 从企业级应用到互联网领域火了很多年，曾经是我招聘架构师的必考题目之一，但 SOA 在大型系统的落地从来都是高难度动作，令许多架构师欲仙欲死。如今又兴起了微服务架构，要把 SOA 进行到底，实现彻底的服务化，从此世间再无系统切分，只有微服务小而美好。那么到底如何实现微服务呢？黄老师这本书教我们轻松上手，一步步把理想变成现实，体现出多年实战派的底蕴，是一本不可多得的武功秘籍。

—— 史海峰，饿了么北京研发中心总经理

近年来，微服务俨然成为行业内广受关注的热点。不论是微服务的价值，还是微服务的阻碍，都是行业在架构技术选型中最为关心的前提。除此之外，技术的践行流程，对现有组织架构、软件模式的影响，都是决策者不敢忽视的要素。我很庆幸看到，国内能诞生这本微服务领域的巨著。本书从架构发展史的角度，阐述了微服务兴起的客观性与必然性；从技术的角度，深入分析了践行微服务的种种要点；更从实践的角度，通过案例事无巨细地帮助读者去体会、理解、掌握微服务。实属呕心沥血之作，极力推荐大家阅读。

—— 孙宏亮，DaoCloud 技术合伙人，《Docker 源码分析》作者

黄勇的这本书从微服务实操的角度，通过在微服务架构体系的不同关注点，选择多样而务实的技术栈，为大家全方位地阐述了微服务架构体系的各种最佳实践，对微服务感兴趣的同学不容错过。

—— 王福强，征数科技 CTO，《Spring Boot 揭秘》和《Spring 揭秘》作者

微服务架构，虽然诞生时间不长，却已成为软件架构领域讨论的热点。微服务的概念看似简单，但涉及诸多方法论和实践积累，这就是为什么有人说它非常好，但就是“玩不起”。随着微服务生态系统的日趋完善，微服务架构的讨论也从 API 接口、服务间通信、接口测试、基础设施自动化等，逐渐扩展到了 API 网关、微服务的注册与发现、Docker 封装与部署、持续交

付以及运维体系的优化等多方面。本书结合作者过去多年的实战经验，深入浅出地梳理了微服务构建过程中遇到的诸多挑战，并给出了切实可行的解决方案（如何使用 Spring Boot 构建服务、使用 ZooKeeper 注册服务，如何结合 Docker 封装服务和发布服务等），是一本能帮助读者立刻动手、落地微服务的好书。同时，作者从开发和运维两个角度入手，详细地剖析了微服务实施过程中，如何有效解决“最后一公里”的部署以及运维难题。纵览全书，条理清楚，图文并茂，理论结合实际，是一本非常用心，又注重实操的好书，对企业的微服务架构实施，具有很大的参考意义，相信企业的架构师、软件开发人员、运维人员读完这本书一定会受益匪浅。

—— 王磊，DevOps 教练，《微服务架构与实践》作者

微服务是近几年的一大热点，其模块化、跨语言和自治隔离等思想，有望大幅降低研发和运维成本。微服务架构，无论对传统企业，还是互联网公司，都会有很大影响。黄勇老师结合了 Spring Boot、Jenkins 和 Docker 等热点技术，对微服务的整个生命周期做了全面介绍，通俗易懂、深入浅出，致力于打造微服务领域最佳实践，不失为一本好书。

—— 吴其敏，携程框架研发部高级总监，开源分布式实时监控系統 CAT 作者

当今，微服务已经不是概念，而是势不可挡的潮流，它在大型互联网电商类企业已有丰富的实践，效果很好。但对于其他有志于向微服务架构转型的技术爱好者，微服务如何落地还存在很多不清楚的地方，本文从细节入手，结合具体实例，娓娓道来，为大家提供一个很好的微服务实践参考，带领大家走进微服务之门。

—— 王庆友，独立架构顾问，《架构的本质》作者

软件开发从来没有银弹，微服务也不是。我认为微服务本质上是要解决一个可伸缩性的问题，以应对访问的增加、业务复杂度的增加和开发团队人员的增加。黄勇在这本书中详细解释了实践微服务必须要面对的架构模式，包括服务注册与发现、API 网关以及简单部署系统的搭建，并辅以样例代码，对于正面临可伸缩性问题的开发人员有很大的参考价值。

—— 许晓斌，阿里巴巴高级技术专家，《Maven 实战》作者

近年来，软件开发领域的新思想、新方法、新工具、新实践层出不穷，简直有令人应接不暇、目眩神迷的感觉。要想走出这团迷雾，微服务是纲，容器化、自动化运维、自动化部署、服务监控与治理等，都是目。通过阅读本书，纲举目张，则一切将尽在掌握！

—— 庄表伟，华为内源平台架构师，《开源思索集》作者

随着移动互联网的崛起，Web 网关越来越重要，本书从 Web 网关的视角带领大家学习微服务架构。通过本书可以学习到如何使用 Spring Boot 与 Docker 等技术构建 Web 型微服务架构，值得 Web 开发人员学习。

—— 张开涛，“开涛的博客”博主，《亿级流量网站架构核心技术》作者

微服务是最近几年在架构方面比较热的一个话题，本书从概念到具体的落地，比较系统地介绍了微服务从构建到部署等环节的知识和具体方案，是想了解和学习微服务相关技能的一本好书。

—— 曾宪杰，美丽联合集团副总裁，《大型网站系统与 Java 中间件实践》作者

读者评价

面对近几年火热的微服务架构，但很多人都是「只可远观，不可近玩」的态度，大家对吨级的概念与方法论往往都是望而生畏。他们会觉得微服务架构是复杂的、高端的、赶潮流的、没有必要的，中小团队没有必要也没有实力去落地。本书将许多晦涩与难以理解的概念和方法论用通俗易懂的文字来描述，非常接地气，实在是不可多得的一本微服务架构经典入门书籍。阅读完本书，会有一种「微服务架构也没有想象中的恐怖嘛」的感觉。勇哥手把手地带着读者把轻量级微服务架构落地，完全是从 0 到 1，以及教给读者如何应对里面的坑。在阅读的时候，有一种勇哥在带着我一起在微服务的世界中探险的感觉，作为我的引路人，非常有安全感。本书的内容都是奔着落地去的，不会有飘在天上的各种方法论，诚意满满的干货，绝对物超所值。勇哥的《架构探险》系列书籍是 Java 从业人员在职场中进阶的宝典。所以，你值得拥有。

—— 偏头痛杨，Java 技术经理

一份轻量级微服务架构最佳实践的讲义。全书总分形式，第一章先构建了轻量级微服务架构图，然后逐个章节图文结合，并带有清单式的讲解，行文简洁易读，深入浅出。从目录可以看出，重点介绍了微服务基础设施方面的知识，是不可多得的落地实战总结，我会推荐给任何在微服务架构道路上的技术人。

—— 泥瓦匠 BYSocket，特赞开发工程师

微服务（MSA）是目前企业级应用主流架构和落地方向，黄岛主（作者雅称）与其团队站在微服务思想、架构的高度，澄清了微服务理念 and 原则，通过微服务实践形成行业领先技术栈经验，值得大家品读。黄岛主的《架构探险：轻量级微服务架构》上下册为传统 IT 企业从事微服务实践提供了一套比较完整的微服务方案和指导原则，是传统架构师向微服务架构师转型床头书，是程序员的必读之物。

—— 罗中华，资深架构师

一直有拜读黄勇老师的博客和书籍，非常佩服黄老师能将晦涩难懂的理论用通俗易懂的语言解释清楚，让我们有豁然开朗的感觉。此次黄老师的新书，用实际的案例详解了微服务架构中基础且重要的日志平台、监控中心、配置中心等。读完样章后，希望能立即阅读实体书，干货满满，很多知识点都是现在正在困扰我们的。希望赶快阅读，尽早解决我们的困惑！

—— 彭清正，宝付支付开发工程师

勇哥的书极富启发意义，曾经对我认识、理解架构起到很大的作用。根据本书的上册，很容易推测出下册将对微服务的原理、实践、运维的认知进行一场洗礼。

—— 李阳，Java 与 Golang 开发者

近三年，微服务架构风靡全球，很多互联网企业都在做微服务改造，黄勇老师的《架构探险：轻量级微服务架构（下册）》这本书记载了特赞生产环境中微服务实施的具体细节，详细地讲解了微服务实施过程中的日志、通信、消息、分布式事务、配置中心、监控等核心内容，是业界难得的一本微服务架构精品书籍，极力推荐大家阅读。相信每一位读者读完本书必定对微服务架构有着更深入的理解，并且能够快速掌握微服务架构开发。

—— 刘国柱，技术经理

前言

2017年，微服务三岁了。我们一直期盼它能快速长大，希望微服务技术社区能推出更多框架与工具，可以帮助我们更好地落地微服务，并从中获得微服务给我们带来的甜头，但实际情况却让我们感到有些失望。我们至今还在寻找适合自身技术需求的微服务架构，甚至仍然觉得微服务离自己有些距离。

因此我们更多的是在观望，希望看到有成功实践微服务的企业能够将自己的技术分享出来，以供更多的企业来使用，但实际情况仍然让我们感到失望。我们不得不选择 Spring Cloud 这样的“全家桶”式的微服务框架来实现微服务架构，此时我们就需要基于 Spring Boot 来开发微服务，Spring Cloud 提供的大量基础设施虽然可与 Spring Boot 进行无缝整合，但这样的架构给微服务的技术选型带来了一定的局限性。此外，Spring Cloud 包含了大量的 Spring 官方所提供的开源项目，目前不同的版本在兼容性方面也存在一些不稳定现象。

我们认为，微服务是一个灵活的技术架构，它不能绑定在特定的技术平台上，微服务不应该存在任何的局限性，同时还要确保有较强的兼容性。比如，虽然我们也使用 Spring Boot 开发微服务，但也允许使用其他更适合的开发框架或编程语言来实现微服务。再比如，我们目前通过 ZooKeeper 来实现服务注册，但也能轻松地切换为其他技术选型，对于整个应用程序而言，这些都是无感知的。微服务所提倡的理念就是，用最合适的技术以最高效的方式来解决实际应用中的问题。

经过两年多的实践过程，我们找到了一款能让微服务架构快速且稳定落地的解决方案，并将此方案的核心内容汇集成本书，希望该方案能给微服务世界带一点新的能量。

本书将重点关注微服务基础设施方面，其中大部分内容涉及微服务运维相关技术。全书以实践的角度进行编写，读者首先将学习轻量级微服务架构的全景视图，随后的各个章节将围绕微服务的日志、监控、通信、解耦、测试、配置六大方面进行展开。读者可亲自动手，从零开始搭建轻量级微服务架构，充分享受架构探险的乐趣。

本书是如何组织的?

第 1 章：轻量级的微服务。

本章将从宏观上描述轻量级微服务架构。首先我们将从架构与架构师开始讲起，简单回顾架构演进的过程与微服务的发展趋势。随后我们将探讨在搭建微服务架构之前需要准备的工作，认识微服务架构的“冰山模型”，介绍切分微服务边界的方法和技巧。最后我们将从部署与运行两个角度来观察微服务架构，并以一幅架构全景图来结束本章。

第 2 章：微服务日志。

本章将关注点放在微服务日志上。首先我们将从 Spring Boot 日志框架入手，使应用日志可以输出到 Docker 容器外部，以便我们可随时查看日志文件。随后我们将学习 Docker 日志驱动，使日志信息输出到 Linux 的 Syslog 中。最后我们将 Syslog 与 ELK 技术栈整合，搭建一款微服务的日志中心。

第 3 章：微服务监控。

本章将视角放在微服务监控方面。首先我们将学习 Spring Boot 应用程序自带的监控特性，接着将介绍 Spring Boot Admin 开源监控系统的使用方法。随后我们将集成 InfluxDB、cAdvisor、Grafana 等开源工具，搭建一款微服务的监控中心。最后我们将学习 Zipkin 工具的使用方法，将其用于微服务的追踪中心。

第 4 章：微服务通信。

本章将围绕微服务之间的通信来展开。首先我们将在 Spring Boot 应用程序中实现基于 HTTP 的同步调用，同时我们也会对比 Spring RestTemplate、OkHttp、Retrofit 等工具的使用方法。随后我们将使用 gRPC 框架实现基于 RPC 的同步调用，并将 gRPC 与 Spring Boot 进行整合。最后我们将亲自动手，搭建一款轻量级分布式 RPC 框架。

第 5 章：微服务解耦。

本章将使用消息队列的异步方式来解耦微服务调用问题。首先我们将对比 ActiveMQ 与 RabbitMQ，它们是两款经典的开源消息队列。随后我们将使用 RabbitMQ 来实现请求应答模式，并通过 RabbitMQ 来实现 RPC 同步调用。最后我们将使用 Event-Sourcing 与 MQ 相结合，巧妙地解决分布式事务问题。

第 6 章：微服务测试。

本章将聚焦在微服务测试方面。首先我们将以 Spring Boot 应用程序为例，分别针对 Service 层与 REST API 进行单元测试。随后我们将使用 Postman 来充当 REST API 的测试工具，并结合

Jenkins 与 Newman 搭建一款 REST API 的自动化测试框架。最后我们将分别使用 Swagger 与 apiDoc 工具来自动生成 REST API 文档，并比较这两款工具的优缺点。

第 7 章：微服务配置。

本章将解决微服务的配置参数问题。首先我们将通过一些实例，快速学习 Ansible 自动化运维工具的使用方法。随后将 Ansible 作为微服务的配置中心，并将 Jenkins 与 Ansible 相结合，可用于优化我们现有的微服务部署框架。最后我们将使用 Registrator 所提供的自注册特性，实现微服务的平滑升级目标。

如何获取本书源码？

可通过以下链接下载本书源码。

<http://git.oschina.net/huangyong/msa-book-2>

如何参与线上互动？

欢迎加入“轻量级微服务架构”QQ 群，申请加入时请注明“架构探险”。

群号：528265294

此外，请关注“架构探险图书”微信公众号，可免费获取关于本书后续的更多内容。



致谢

首先需要感谢的是我们的技术团队，如果没有你们所提供的实践经验，这本书也不可能问世。我很庆幸自己能加入特赞技术团队，也很感激你们对我一如既往的支持与鼓励。在微服务实践这条路上，你们才是专家，我只是把你们的宝贵经验整理出来，希望能让更多的人从中受益，我想这也是我们特赞技术团队的共同理想。

自从去年 9 月上册出版以后，很多读者都在问“下册什么时候出版？”，我原本以为是今年上半年就能完成的事情，没想到计划不如变化，公司有更加重要的工作需要我去完成，因此写书的计划也被频繁打断。直到一年后的今天，下册才能与你们见面。在此，我先对你们说声抱

歉，也感谢你们一直对我的关注与等待，希望下册能给你们带来更多的帮助。

最近这一年中，我在很多公开场合下分享过关于微服务的话题，在此向曾经帮助我的技术专家们致谢，和你们聊技术是一件兴奋的事情，让我更加深刻地认识到微服务的本质，也让我对我们技术团队现在搭建的微服务架构更有信心了。同时也感谢你们对这本书所做的推荐，希望我的全力以赴能够配得上你们的称赞。

如果说写完一本书全是作者的功劳，那就大错特错了。没有好的出版社，没有优秀的编辑，我想再好的书也会让大家失望。感谢本书编辑陈晓猛先生对本书的辛勤付出，我在晓猛身上学会了谨慎与专注，这是我和晓猛第三次合作，每次合作都能让我感到，写书其实是一件快乐的事情，我很享受这个过程。

最后我想把感谢的话留给我的妻子和女儿，感谢你们一路陪伴着我，我们共同见证着对方的成长。这本书能够顺利完成，绝对离不开你们努力，因为你们给了我一个幸福而美满的家庭，让我能够将心思沉醉在写作之中。虽然最后我才感谢你们，但你们在我心中永远是最重要的人，永远无法取代。

黄勇

2017年7月18日于上海

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/32447>



目录

第 1 章 轻量级的微服务	1
1.1 微服务将变得轻量级	2
1.1.1 架构与架构师	2
1.1.2 架构演进过程	4
1.1.3 微服务架构发展趋势	6
1.2 微服务架构前期准备	7
1.2.1 认识微服务架构冰山模型	7
1.2.2 冰山下的微服务基础设施	8
1.2.3 根据业务切分微服务边界	9
1.3 轻量级微服务架构图	10
1.3.1 轻量级微服务部署架构	10
1.3.2 轻量级微服务运行架构	11
1.3.3 轻量级微服务全局架构	12
1.4 本章小结	13
第 2 章 微服务日志	14
2.1 使用 Spring Boot 日志框架	15
2.1.1 使用 Spring Boot Logging 插件	15
2.1.2 集成 Log4J 日志框架	17
2.1.3 将日志输出到 Docker 容器外	19
2.2 使用 Docker 容器日志	19
2.2.1 Docker 日志驱动	20
2.2.2 Linux 日志系统：Syslog	23