

- 腾讯云Serverless实战经验总结
- 从原理到应用场景，再到项目实战
- 让从业人员快速通往Serverless殿堂

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

# Serverless架构

## 从原理、设计到项目实战

刘宇◎著

 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>

内容简介

本书以 Serverless 架构为切入点，深入探讨了 Serverless 架构的核心理念、设计原则、最佳实践以及其在不同场景下的应用。本书不仅适合初学者入门，也适合有一定经验的开发者深入学习和实践。本书将带你了解 Serverless 架构的各个方面，包括其优势、挑战以及如何利用 Serverless 架构构建高性能、可扩展的应用程序。

本书详细介绍了 Serverless 架构的核心理念、设计原则、最佳实践以及其在不同场景下的应用。本书不仅适合初学者入门，也适合有一定经验的开发者深入学习和实践。本书将带你了解 Serverless 架构的各个方面，包括其优势、挑战以及如何利用 Serverless 架构构建高性能、可扩展的应用程序。

# Serverless架构

## 从原理、设计到项目实战

刘宇◎著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

## 内 容 简 介

本书面向有兴趣使用或者正在使用 Serverless 架构部署相关应用与服务的技术人员,通过对 Serverless 基础的介绍、对相关规范和标准的描述,对架构、应用的讲解与阐述,为读者提供相对客观和完整的学习资料及文档资料;通过网站后端、监控告警、大数据与人工智能、物联网与高可用服务等不同场景的应用实例,以及简单博客系统的建设与基于人工智能的相册小程序系统的建设等完整项目,为读者提供更多的灵感与启发。基于本书所提供的十余个场景实践及两个完整项目,读者不仅可以快速入门 Serverless 架构,也可以将这些应用场景的代码直接应用到测试项目中,快速体验 Serverless 架构带来的便利与变化。

本书不仅包括 Serverless 架构的相关知识、规范标准、应用实践,也包括工业化产品和开源产品的相关探究和分析,还包括开源工具的介绍和使用方法,可以让读者对目前的 Serverless 相关产品有更深入的了解,甚至可以自行搭建属于自己的 FaaS 平台。本书的特色是学习与实践结合,希望通过广泛的场景抛砖引玉,让读者读完本书之后不仅可以知道什么是 Serverless 架构,更可以知道 Serverless 架构为我们带来了哪些变化、可以做哪些事情,最主要的是,可以知道 Serverless 架构能为自己做哪些事情,并进行相关实践。

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

Serverless 架构:从原理、设计到项目实战 / 刘宇著. —北京:电子工业出版社, 2019.11  
ISBN 978-7-121-37628-3

I. ①S… II. ①刘… III. ①移动终端—应用程序—程序设计 IV. ①TN929.53

中国版本图书馆 CIP 数据核字(2019)第 220461 号

责任编辑:董 英

印 刷:天津千鹤文化传播有限公司

装 订:天津千鹤文化传播有限公司

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:19.75

字数:455 千字

版 次:2019 年 11 月第 1 版

印 次:2019 年 11 月第 1 次印刷

印 数:3500 册 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 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。

# 推荐序 1

Fourteen years ago, in 2005, it was unfathomable to imagine building an internet application without spending millions of dollars in upfront capital and tens of thousands of working hours building one's own data center and developing the requisite software stack — constraining this domain to a small set of companies and stymying product innovation.

The launch of Amazon Web Services' Elastic Compute Cloud in 2006 enabled customers to skip the capital intensive process of building data centers and purchasing hardware. This model, known as Infrastructure as a Service(IaaS), allowed customers to rent compute capacity in a pay as you go model — opening up internet application development to a broader audience. However, applications still required building and operating associated backend systems on top of hardware — requiring large engineering teams whose sole focus is on backend infrastructure, not product innovation. Cloud computing remained a product for operators, not for product innovators or individual developers.

Today, we are at a turning point in cloud computing. Serverless computing — the idea that customers only write the required business logic while the cloud provides everything else — represents a quantum leap in cloud computing. Finally, it's possible for hardware, infrastructure, application platforms, and operations to be abstracted away, freeing developers to focus solely on product innovation. Industry trendsetters such as Netflix have already shifted parts of their core business logic over to serverless — improving product innovation, reducing cost, and enhancing reliability. Application platforms built on serverless are emerging, such as WeChat mini-programs, allowing individual developers to productize and scale them for millions of users quickly.

This book serves as an excellent introduction to readers interested in serverless architectures — striking a balance between theory and application — while also introducing the history and background.

The serverless revolution will bring cloud computing to the masses. I eagerly await the products this revolution will unlock!

Yunong Xiao

腾讯云中间件总经理、首席架构师

2019年10月18日

推荐序 1 大意如下：

14 年前，也就是 2005 年，那时我们无法想象在不花费数百万美元和数万小时构建自己的数据中心的情况下，能够开发一款软件或者构建一个网络应用，当时，这也是阻碍一些小型团队持续创新与发展的重要原因。

2006 年，Amazon Web Services 推出弹性计算云服务，该项服务使客户可以跳过建设数据中心和购买硬件的资本密集型流程，被称为 Infrastructure as a Service，即 IaaS。IaaS 允许客户以“即用即付”的模式租用计算能力，从而可以面向更广泛的客户开放应用开发所必需的基础设施。然而，应用程序仍然需要在硬件上构建，同时需要运行相关的后端系统，这同样需要大型工程团队全身心投入和关注后端基础设施，而不是只关注产品创新等。云计算仍然是云厂商的产品，而不是个人开发者的产品。

今天，我们正处于云计算发展的转折点。Serverless——一个客户仅编写所需的业务逻辑代码而云厂商提供其他所有功能的架构，代表了云计算的一种巨大进步与飞跃。这个架构有可能抽象出硬件、基础架构、应用程序平台和操作等，这个架构可以使开发人员将精力完全放在产品创新上。像 Netflix 这样的行业引领者已经将其部分核心业务逻辑实现转移到了 Serverless 架构上，从而促进了产品创新，降低了成本并增强了可靠性。建立在 Serverless 架构上的应用平台正在兴起，例如微信小程序，它使个人开发者可以快速为数百万用户提供稳定的产品和服务。

该书围绕 Serverless 架构，不仅有历史和背景的相关描述，还有理论基础的总结和应用实践的经验分享，对于对 Serverless 架构感兴趣的读者来说，是一本不错的书籍。

Serverless 革命将把云计算带给大众。我热切期待这场革命将解锁的产品！

# 推荐序 2

从简单的 C/S、B/S 到云时代，是系统架构的阶段性的跨越。云是目前所有用户服务、客户服务不可忽视的架构基础，Serverless 是云架构发展趋势中一股强劲潮流。

Serverless 的出现使得单一职责和关注分离的思想有了更进一步的落地。从业务服务及应用开发的角度看，Serverless 带来的优势非常明显：高扩展性、低运维成本、低开发成本。让更合适的“人”做更合适的事，业务方可以将更多的精力放到业务创新上。

Serverless 还是较新的概念，Serverless 本身及 Serverless 相关生态还需要我们共同建设。我们相信，在不久的将来 Serverless 会成为中小企业的首选系统架构，Serverless 架构落地应用也会催生更多的业务创新，我们最终服务的用户及客户也会从技术发展中获得更多的收益，包括用户体验及经济收益等。

本书从全局考虑，较为全面地介绍了 Serverless 知识体系，内容涵盖 Serverless 技术入门、架构到复杂应用构建，既有基础理论，也有方法指导，更包含丰富的实战案例，是一本 Serverless 领域必备的技术丛书。

高磊

阅文集团技术部总经理

# 专家力荐

在数字经济的时代背景下，传统行业的数字化转型将成为未来经济增长的重要支柱，而云原生技术作为云计算领域的新势力，得到了业界的广泛关注。从虚拟化到容器化，再到 Serverless 化，这些新技术的应用，打破了传统 IT 资源的固有边界并逐步降低颗粒度，改变了开发、运维原有的生产方式，凸显了业务应用价值并最终实现效益最大化。Serverless 的诞生进一步降低了企业信息化的门槛，让计算资源无处不在却又平滑无感，降低了开发的复杂度并能快速交付。Serverless 本身具有自动伸缩、低成本、免运维、按需付费等特性，未来将在行业应用实践中大放异彩。本书从概念到实战，再到典型应用场景分析，较为全面地展现了 Serverless 的独特魅力，为从业人员通往 Serverless 殿堂指明了方向。

——陈屹力 云原生产业联盟秘书长、中国信息通信研究院云大所云计算部副主任

Serverless 作为一种新型的互联网架构，是当前很火的技术之一，它直接或间接推动了云计算的发展。本书从一线互联网企业技术人员的角度，对目前热门的 Serverless 架构进行了全面系统的解读，不仅讲述了基本原理，还介绍了大量实战案例和项目，使读者能够清晰直观地了解 Serverless 是什么、能做什么、如何应用等。Serverless 的底层原理较复杂，本书从原理、设计到实战，深入浅出地进行了介绍，适合不同知识层次的读者阅读。

——卜佳俊 浙江大学软件学院常务副院长

云计算技术可能从概念产生到实际产业化最快的技术。在过去 10 年里云计算技术不断发展，从 IaaS、PaaS、SaaS 到 FaaS、BaaS，向着理想目标逐渐逼近。Serverless 技术是其最新进展，它让开发人员无须关注云服务的部署和管理，显著降低了开发云计算程序的难度。本书从原理出发，以当前典型的应用为案例，具体介绍了 Serverless 技术的使用方法，突出实用性，具有很强的实战特点。

——窦勇 国防科技大学教授、中国计算机学会体系结构专委会主任

Serverless 是云计算发展到一定阶段的必然产物，未来会越来越流行，开发者将不再需要参与基础设施及软件的维护，尤其是大规模的集群运营，成本将大幅度降低。本书从 Serverless 入门到它在人工智能、物联网领域的应用都有详细介绍，是一本值得一读的 Serverless 书籍。

——于涛 腾讯前端技术通道负责人

Serverless，无服务器架构，是云计算的一种计算资源组织和分配形式。Serverless 相对于传统算力，就好比高级语言相对于汇编语言，它使开发者能够将绝大部分时间用于思考和实现核心业务逻辑代码，而不耗费过多精力在系统层面的资源调度、资源分配、健壮性及扩展性方面。在 Serverless 架构下，开发的本质是将有限资源更多地投入到核心业务中，减少“Time to Market”，进一步提升 ROI。本书来自我的好友刘宇，他细致思考，潜心写作，内容由浅入深，值得阅读。

——王俊杰 腾讯云 Serverless 技术专家

非常高兴看到一本介绍 Serverless 架构的好书面世。我在 20 年的软件开发生涯中反复思考过，计算能力如何才能像水一样，一拧水龙头就有，要多少有多少，要多少支付多少费用。网络计算、云主机、容器、BaaS 都让我惊喜过，但是直到 Serverless 相关技术出现后，我才终于找到了一个兼顾灵活性、通用性、开发便利性、规模弹性、经济性的好答案。有了 Serverless，我们能超低成本地快速实践创意、试错、发布，不再受限于我们的后台架构能力。希望更多的开发者拥抱 Serverless，用更简单高效的方式来面对更复杂的互联网未来。

——黄希彤 前端专家、青宁信安科技 CTO

在云原生和 DevOps 的技术背景下，Serverless 的出现让开发者可以更加专注于业务逻辑和应用的实现。感谢本书作者在中国 DevOps 社区所做过的分享，相信本书能够使更多对 Serverless 感兴趣的朋友们获益。本书从实战出发，理论结合实践，可以指导那些新技术的探索者们获得第一手实战经验。

——刘征 Elastic 技术布道师、DevOps 社区组织者

前端行业经过十多年的刀耕火种，逐渐实现了工程化、开发模式规范化，进入深水区。Serverless 特别是 FaaS 的逐步广泛应用，给前端带来了从纯前端研发到应用研发转变的可能，这不仅会涉及新领域带来的工作职责的变化，还可能带来新的高效的研发模式。本书能够从概念到原理清楚地讲解 Serverless 的本质及应用，建议想更加深入理解 Serverless 的前端同学都阅读一下。

——侯鹏 阿里巴巴神马搜索前端技术负责人

Serverless 技术正在成为微服务、容器云之后的另外一个架构热点。这一技术进一步淡化了应用生命周期中的资源运维环节，以业务支撑为核心，为应用提供完整的构建和运行支持。本书作者是腾讯云 Serverless 团队的主力成员，由浅入深地对 Serverless 技术进行了系统全面的讲解，并给出了大量实际应用方法。相信各个技能水平的架构师和开发者，以及不同 Serverless 技术流派的用户都能从本书获益。

——崔秀龙 知名 YAML 架构师、腾讯资深技术专家

# 前言

云计算的不断发展给社会带来了诸多便利，云计算与大数据、人工智能的结合，更让我们生活悄然发生着翻天覆地的变化。从 IaaS 到 PaaS，再到如今的 FaaS，每一个阶段都是人类文明前行的一小步，但却是当前技术发展的一大步。Serverless 自概念被提出到现在，搜索热度与关注度、相关开源项目的增长速度与社区活跃度，都充分显示了广大云厂商、开发者对这个架构的支持和期待。甚至有人说 Serverless 才是真正的云计算，这种说法可能有些夸张，但是不可否认，Serverless 技术确实在逐渐融入人们的生活并改变世界，无时无刻不在向我们阐述“云时代”真的要到来了。

本书将通过 Serverless 架构的基本描述，介绍其发展历史、应用场景及未来展望，让读者可以更好地了解什么是 Serverless 及 Serverless 的过往、未来。本书也对 Serverless 架构的规范、标准等进行了描述，对开源框架、开源工具、工业化产品进行了相关分析，让读者在明白如何使用 Serverless 或者 Serverless 有哪些特性的同时，可以真正地使用 Serverless 相关产品，甚至可以通过开源项目搭建一套自己的 Serverless 服务。最后，本书通过监控告警、网站后端、人工智能、大数据、物联网、高可用等多个领域的场景实践，以及两个完整的项目实战，并且与 Web 应用、微信小程序、快应用等结合，为读者进行了更加细致的分析和讲解，希望通过这种方式使读者不仅知道 Serverless 是什么，Serverless 能做什么，还可以获得更多的启发，并将 Serverless 架构应用于实战，通过 Serverless 架构创造更多的价值，实现自身业务的快速上云。

本书共分为 7 章，详细介绍了 Serverless 的概念、标准规范、场景实践及项目实战。

第 1 章，主要介绍什么是 Serverless、Serverless 架构的优势和劣势、影响和价值、典型应用场景及对未来的展望，这一章是 Serverless 架构的介绍部分，读者可以对 Serverless 架构有一个全面深刻的认识，甚至可以评估出自己目前的产品是否适合接入 Serverless 架构、如果接入会有哪些好处和变化等。

第 2 章，主要介绍 Serverless 相关的标准和规范、开源产品发展情况、工业化产品发展情况及开发工具建设情况等。读者可以明确地知道 Serverless 架构的组成及各种相关产品的关系等，同时通过对开源产品的分析和探讨，深入了解 OpenWhisk、Fission、Kubeless 等相关开源项目，甚至可以动手搭建一个属于自己的 Serverless 平台。通过对工业化产品的分析，读者可以知道 AWS、Google、Azure、腾讯云等云厂商在 Serverless 架构方面的发展情况。通过开发者工具部

分，读者可以明白自己如果使用 Serverless 架构，应该如何进行开发。

第 3、4、5 章，结合 Web 服务、监报告警、大数据、人工智能及物联网等为读者带来十余个场景实践，通过这些场景实践希望读者对 Serverless 架构的作用及应用场景有更多的思考、获得更多启发，并深度认识 Serverless 架构给开发者、运维者等带来的变化，进而将 Serverless 架构与自身的项目和业务结合，并拓展思路探索更多的应有场景，创造更多的价值。

第 6 章和第 7 章，是两个相对完整的实战项目，从需求分析开始，到功能设计、数据库设计、接口设计，再到后端功能实现、前端功能实现，最后部署上线，通过将对象存储、云数据库及云函数、API 网关等产品进行融合，构建出完整的 Web 项目和微信小程序项目。在这两个项目中有大量的人工智能技术的应用，包括自然语言处理、推荐系统及计算机视觉等技术，不仅为学习增加了趣味性，也提高了项目的应用价值。

本书在编写过程中力求深入浅出、重点突出、简明扼要，尽可能方便不同专业背景和知识层次的读者阅读，同时通过基础知识与实践项目结合，力求为读者打造一种可学习、可实践、可思考、可探索的阅读体验。

在本书的编写过程中，腾讯云 Serverless 团队给予了大力支持，包括 Yunong、Cliff、Scott、Horace 等在内的前辈们都提供了大量帮助。Tina、Liujiang 等也对本书的编写及相关文章的发表等提供了重要帮助和保障。在此特别感谢大家。另外，本书部分内容参考了公开资料和网络上的资源，在此也对相关作者致以深深的感谢。

最后需要说明的是，Serverless 是目前非常火的一种技术，其底层原理相对复杂，应用领域相对广泛，因此编写一本完美的 Serverless 图书绝非易事，书中难免存在疏漏或者错误，希望广大读者不吝赐教。如有任何建议、意见或者疑问，请及时联系作者，以便在后续版本中改进和完善。

## 读者服务



- 获取本书配套素材
- 获取更多技术专家分享视频与学习资源
- 加入读者交流群，与更多读者互动

微信扫码回复：37628

# 目录

<b>第 1 章 初识 Serverless</b> .....	<b>1</b>
1.1 什么是 Serverless.....	2
1.2 优势与劣势.....	6
1.3 影响力和价值.....	11
1.4 技术难点与挑战.....	14
1.5 典型应用场景.....	18
1.5.1 视频转码服务.....	18
1.5.2 数据 ETL 处理服务.....	19
1.5.3 网站（服务）后端.....	20
1.5.4 人工智能推理预测.....	20
1.5.5 物联网相关应用.....	21
1.6 发展与展望.....	21
<b>第 2 章 Serverless 入门</b> .....	<b>24</b>
2.1 Serverless 基础.....	25
2.2 Serverless 架构规范.....	28
2.2.1 处理模型.....	28
2.2.2 函数相关规范.....	29
2.2.3 函数生命周期.....	32
2.2.4 工作流程.....	37
2.3 事件与规范.....	39
2.3.1 相关术语.....	39

2.3.2	上下文属性.....	40
2.4	开源项目与技术.....	44
2.4.1	OpenWhisk 项目.....	44
2.4.2	Fission 项目.....	54
2.4.3	Kubeless 项目.....	70
2.5	工业化产品.....	78
2.5.1	AWS Lambda.....	78
2.5.2	Google Cloud Function.....	80
2.5.3	Azure Functions.....	80
2.5.4	腾讯云 SCF.....	82
2.6	开发运维工具.....	83
2.6.1	CLI 命令行工具.....	83
2.6.2	插件.....	91
<b>第 3 章</b>	<b>Web 与监控告警场景实践.....</b>	<b>94</b>
3.1	运维、监控与告警.....	95
3.1.1	网站状态监控告警.....	95
3.1.2	服务状态监控告警.....	107
3.2	Web 后端与应用服务端.....	114
3.2.1	上传图片加水印.....	114
3.2.2	用 IP 地址查天气并在小程序中应用.....	122
3.2.3	简易用户反馈功能与快应用实践.....	136
<b>第 4 章</b>	<b>大数据与人工智能场景实践.....</b>	<b>147</b>
4.1	大数据技术之 MapReduce 的简单实现.....	148
4.2	人工智能领域的实践.....	158
4.2.1	自然语言处理为你的网站赋能.....	158
4.2.2	智能客服与微信小程序.....	163
4.2.3	简单的车牌识别系统.....	174
4.2.4	视觉：对象检测系统.....	178

<b>第 5 章 物联网和其他场景实践</b> .....	190
5.1 物联网相关——定制智能音箱小工具.....	191
5.2 其他场景实践.....	197
5.2.1 OJ 判题机/小程序在线编程系统实现.....	197
5.2.2 Serverless 与高可用服务——使用云函数实现双机存储.....	202
5.2.3 Serverless 与高可用服务——基于 Serverless 架构的高可用模型.....	206
5.2.4 微信公众号机器人.....	215
<b>第 6 章 实战：Serverless 与 CMS</b> .....	224
6.1 项目背景.....	225
6.2 项目设计.....	225
6.2.1 功能设计.....	225
6.2.2 数据库设计.....	229
6.2.3 接口设计.....	231
6.3 数据库开发.....	239
6.4 后端开发.....	241
6.5 前端开发.....	253
6.6 功能演示.....	257
6.7 项目总结.....	260
<b>第 7 章 实战：小程序图片管理系统</b> .....	261
7.1 项目背景.....	262
7.2 项目设计.....	262
7.2.1 功能设计.....	262
7.2.2 数据库设计.....	266
7.2.3 接口设计.....	268
7.3 数据库开发.....	275
7.4 后端开发.....	277
7.5 前端开发.....	298
7.6 效果展示.....	301
7.7 项目总结.....	304



## 1.1 什么是 Serverless

近年来，IT 技术的更新迭代速度非常快，每个时间点都有典型的代表名词及概念，就目前而言，人工智能领域中的机器学习、深度学习、强化学习等，还有区块链、物联网等非常火热。在云计算领域，有一种技术被众多云厂商认为是“风口项目”，甚至可以颠覆现有云计算中的某些格局，为此 AWS、Google 腾讯云、阿里云等云厂商都投入了大量精力进行相关产品建设，这种技术就是 Serverless。

Serverless 又名无服务器，所谓无服务器并非是不需要依赖和依靠服务器等资源，而是开发者再也不用过多考虑服务器的问题，可以更专注在产品代码上，并且无须管理和操作云端或本地的服务器，同时计算资源也开始作为服务出现，而并非作为服务器的概念出现。Serverless 是一种构建和管理基于微服务的架构的完整流程，允许你在服务部署级别而不是在服务器部署级别来管理你的应用部署。与传统架构的不同之处在于，它完全由第三方管理，由事件触发，存在于无状态（Stateless），暂存（可能只存在于一次调用的过程中）在计算容器内。Serverless 真正做到了部署应用时无须涉及更多的基础设施建设，就可以基本实现自动构建、部署和启动服务。

众所周知，云计算的发展历程如图 1.1 所示，从 IaaS、PaaS、SaaS 到最新的 BaaS、FaaS 等。

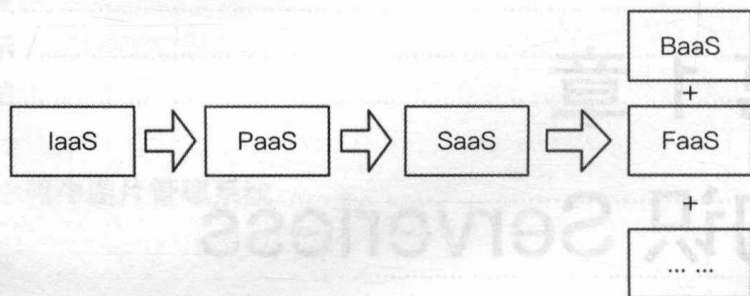


图 1.1 云计算的发展历程

在云计算的发展过程中，从 IaaS 到 PaaS 再到 SaaS 的过程中，去服务器化越来越明显，如图 1.2 所示。

目前的 Serverless 去服务器化到了什么程度呢？有人这样说，无服务器架构是传统的云计算平台的延伸，是 PaaS 向更细粒度的 BaaS 和 FaaS 的发展，那么 Serverless 与 FaaS 有什么关系？FaaS 与 PaaS 又有什么区别呢？

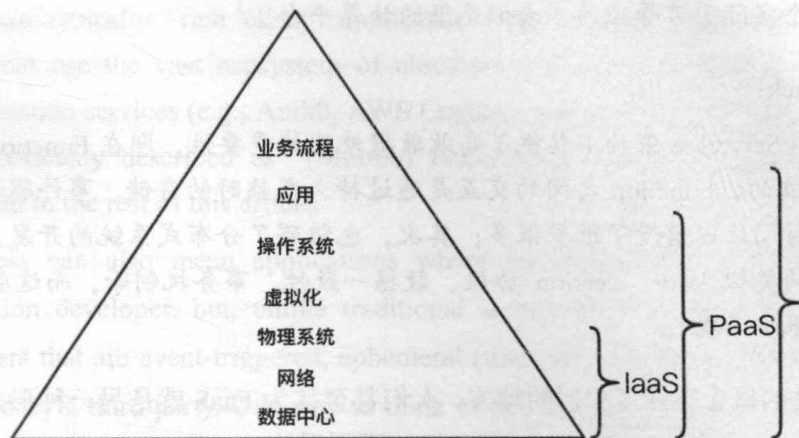


图 1.2 IaaS 和 PaaS、SaaS 的区别

微服务 (Micro Service) 是软件架构领域另一个热门的话题。如果说微服务以专注于单一责任与功能的小型功能块为基础，利用模组化的方式组合出复杂的大型应用程序，那么我们可以进一步认为 Serverless 架构可以提供一种更加“代码碎片化”的软件架构范式，我们称之为 Function as a Services (FaaS)。而所谓的“函数”(Function) 提供的是相比微服务更加细小的程序单元。例如，微服务可以代表为某个客户执行所有 CRUD 操作所需的代码，而 FaaS 中的函数可以代表客户所要执行的每个操作：创建、读取、更新及删除。当触发“创建账户”事件后，将通过函数的方式执行相应的“函数”。单就这一层来说，可以简单地将 Serverless 架构与 FaaS 概念等同起来。但是如果就具体的概念深刻探索，Serverless 和 FaaS 还是不同的，UC Berkley 发表过一篇关于 Serverless 观点的文章：*Serverless Computing: One Step Forward, Two Steps Back*<sup>1</sup>：

因此，云提供商很快就会强调 Serverless 并非不存在服务器，也并不仅仅是 FaaS。它是由“Standard library”支持的 FaaS：即由供应商（例如 AWS、阿里云、腾讯云等）提供各种底层能力。对于 AWS 而言，这种底层能力包括 S3 (Large object storage)、DynamoDB (Key-value storage)、SQS (Queuing services)、SNS (Noti-fication services) 等，这些能力也可以称为 BaaS。这些基础设施或者说底层能力由 AWS 管理和运营，开发人员只需注册使用相应服务即可，开发人员通过编写函数代码，使其运行在 FaaS 平台上，在适当的时候，可以使用这些底层能力，并且“即付即用”，即按量付费。

其中关于 One Step Forward 及 Two Steps Back 的说法大概如下：

#### One Step Forward:

通过提供 Autoscaling (自动伸缩，也称为自动扩缩容)，今天的 FaaS 产品向云编程迈出了一

1 参考地址[1]。全书参考地址列表可到“博文视点”官网获取。

大步，提供了一个实际上可管理的、看似无限的计算平台。<sup>1</sup>

### Two Steps Back:

首先，它们（Serverless 架构）忽视了高效数据处理的重要性，即在 Function 的情况下，每个 Function 是独立的，Function 之间的交互是通过持久或临时的存储、事件驱动来完成的，导致了完成交互的时间比以前慢了很多很多；其次，也阻碍了分布式系统的开发，即通常分布式系统会依赖很多的类似 leader election 协议、数据一致性、事务机制等，而这些在目前的 FaaS 类型的平台里是很难实现的。

FaaS 与 PaaS 的概念有许多相似的地方，人们甚至认为 FaaS 就是另一种形式的 PaaS，但是 Intent Media 的工程副总裁 Mike Roberts 有自己的不同看法：

Most PaaS applications are not geared towards bringing entire applications up and down in response to an event, whereas FaaS platforms do exactly this.（大多数 PaaS 应用程序都不适合响应事件而对整个应用程序进行调整，而 FaaS 平台是可以这样做的。）

FaaS 和 PaaS 在运维方面最大的差异在于缩放能力。对于大部分 PaaS 平台，用户依然需要考虑缩放。但是对于 FaaS 应用，这种问题完全不存在。就算将 PaaS 应用设置为自动缩放，依然无法在具体请求的层面上进行缩放，而 FaaS 应用在成本方面少很多。AWS 云架构战略副总裁 Adrian Cockcroft 曾经针对两者的界定给出了一个简单的方法：

If your PaaS can efficiently start instances in 20ms that run for half a second, then call it serverless.（如果 PaaS 可以在 20ms 内有效地启动运行半秒的实例，那么它就可以被称为 Serverless 架构。）

正因为 Serverless 与 PaaS 的不同，以及 Serverless 与 FaaS 和 BaaS 的关系，所以有人说 Serverless 真正实现了当初云计算的目标，更有人这样定义 Serverless：

Serverless = BaaS + FaaS + ...

诚然，这种说法未必是百分之百准确的，就像 Martin Fowler 的一句话：Like many trends in software, there's no one clear view of what Serverless is. 但是不可否认的一点是 Serverless 在很多层面、很多领域，越来越显现出“期望中云计算”的样子了。Martin Fowler 在 *Serverless Architectures* 中对 Serverless 这样定义：

- Serverless was first used to describe applications that significantly or fully incorporate third-party, cloud-hosted applications and services, to manage server-side logic and state.

<sup>1</sup> 参考地址[2]。