Broadview®
www.broadview.com.cn

总有人为你制定规则，总有人试图给你答案，但没有人可以代替你思考。
——韩寒

持续演进的
Cloud Native
云原生架构下微服务最佳实践

王启军 / 著

中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

持续演进的
Cloud Native
云原生架构下微服务最佳实践

王启军 / 著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

本书从架构、研发流程、团队文化三个角度详细介绍了如何构建 Cloud Native。作者长期活跃在研发一线，具有丰富的架构设计经验，也曾亲身经历过很多失败的架构设计，如很多团队在实施微服务架构的时候，只强调拆分服务，根本没有理解微服务架构应该怎么做。本书就是想告诉读者，除了拆分服务，还要把哪些事做好，例如基础设施、一致性、性能、研发流程、团队文化等。

本书共分为 10 章，第 1 章从整体上描述了 Cloud Native 的起源、组成及原则等；从第 2 章到第 7 章重点描述了微服务架构、敏捷基础设施及公共基础服务、可用性、可扩展性、性能、一致性等方面的设计实践；第 8 章介绍了 Serverless 和 Service Mesh；第 9 章介绍了如何构建研发流程；第 10 章介绍了如何建设团队文化。

本书希望给技术管理者、架构师和有一定基础的技术人员提供帮助，特别是希望改变研发模式，从交付型软件过渡到云服务的传统软件企业开发者，此书将帮助你少走弯路。

图书在版编目（CIP）数据

持续演进的 Cloud Native：云原生架构下微服务最佳实践 / 王启军著. —北京：电子工业出版社，2018.10
ISBN 978-7-121-35120-4

Ⅰ. ①持… Ⅱ. ①王… Ⅲ. ①程序语言－程序设计 Ⅳ. ①TP312

中国版本图书馆 CIP 数据核字（2018）第 222022 号

责任编辑：汪达文
印　　刷：三河市华成印务有限公司
装　　订：三河市华成印务有限公司
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：19.5　字数：410 千字
版　　次：2018 年 10 月第 1 版
印　　次：2018 年 10 月第 1 次印刷
印　　数：2500 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 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。

推荐序一
云端应用时代之光

写书殊为不易，分享精神更是难能可贵。我一直坚定地认为，能把自己的技术经验写成一本公开出版的技术书籍是件非常了不起的事，因此对写书的朋友总会肃然起敬，对创作了多本技术书籍的朋友更是高山仰止。启军是我在当当时的同事，我对他的独到思维和行动力印象深刻，作为架构师他推动了产品线架构的演化升级，这样的同事令人兴奋。如今他从一线实践经验出发，结合行业主流技术发展趋势总结出自己的系统思考。具体而言，启军为云原生技术架构体系著书立说，在 IT 技术的浪潮之巅做布道师，给了我一个大大的惊喜。与这样努力的朋友为伍，我哪敢不持续学习和成长呢？

云原生架构是 IT 技术在云计算时代的进化升级，标志着云端应用进入成熟阶段。技术的价值是高效稳定、快速响应、驱动甚至引领业务发展，避免叠见层出，以及减少工作量。成规模的系统和团队需要与之匹配的技术体系。云计算兴起之时，有人说：“未来技术人员会分成两种，一种是构建云的，另一种是基于云构建应用的”。那时还没有成熟的云解决方案，对云计算的畅想也只能局限于原有的技术产品。如今云计算时代已经到来，应运而生并经过时间锤炼的云原生技术是这个时代的热点，因此技术人员只有与时俱进、更新技能，才能走向未来。

架构持续演进，技术也会更新换代，也许几年后书中的许多名词都会成为历史，而书

中所介绍的架构处理方法和思考问题的角度依然有借鉴意义。互联网时代的 IT 技术一直在加速演进，多种产品协同形成体系，技术人员进阶需要掌握整套技术栈。通过本书，我们不仅能够掌握云原生的各个关键组件，更能理解架构设计的思想，掌握技术架构持续演进的过程，以及其与团队文化、研发流程的共存互生的关系。从开发编码晋级到架构设计，需要提升认知、开阔视野，相信会有很多同道中人，在多年后回首来时的路，会想起这本书对自己的帮助，想起那些在字里行间有所得的刹那，心中暗喜，转身昂首阔步向前！

史海峰

公众号“IT 民工闲话”作者

推荐序二

有幸和启军共事一年多，我总是被他执着的工作态度和敏锐的技术洞察力深深震撼。启军在技术分享时说过一句话“有人告诉你答案，但没人代替你思考”，这句话引发了大家的共鸣，而他对业务的思考十分全面，推动了公司整体架构水平的持续提升。本书中提到的分布式 UUID、JOB 调度、消息队列和集群缓存，不仅依然在高效平稳地运行，而且帮助公司架构从基础组件演进到了云平台，并对外提供平台级服务。这充分展现了启军对架构持续演进的前瞻性。启军对业务端服务的剥离和抽象使得上百万数量级的商品变价可以实时实现，时效性和高可用在启军的设计中体现得淋漓尽致，这些在书中都有介绍。

启军耿直的工作作风和开放的心态，使得大家与他的合作都非常高效。各服务场景自己验证压力情况，破坏性测试恢复机制，如何做平衡方案，本书也都有详细介绍。这本书真正体现了实践的价值，为架构的持续演进做了铺路石，更为依然奋斗在云原生路上的朋友们提供了实践指南。不论是刚开始接触业务开发的同学，还是已经有多年实战经验的专家，阅读本书都会有醍醐灌顶之感。

刘利川

当当高级技术总监

序

架构没有绝对的对与错

在技术的领域里，并不存在“上帝”。没有人的每句话都是对的，没有人的所有思想都能被别人所接受。

我经常在公司范围内培训，首先是灌输架构思想和解决方案，然后会在实战演练中模拟一个比较简单的业务场景，把所有人分成 4 个团队，每个团队大概有 10 个人。结果发现，每个团队最终形成的架构图总有很大差异，很难评价一个团队的做法是对是错。例如，是要拆分为 3 个服务，还是 5 个服务，他们有各自的理由，除非比较明显的问题，否则你很难以一个理由去否定另一个理由。原因只是各个团队站在了不同的维度综合判断、权衡，形成了自己认为满意的架构方案。因此，架构没有绝对的对与错，只是在不同的角度做出的决定而已。

架构很难被衡量

每个公司的管理层都希望尽可能地去衡量架构的先进性，希望认清差距，向着好的架构方向不断演进。然而架构很难被衡量，须同时具备差距特别明显、制定指标的人能力达到一定高度、业务场景比较接近这三条才有可能衡量。当然我们可以去制定一些指标，这些指标应该是参考性的，作为一个自检项，而不是评价标准。从这个角度看，并不是符合 Cloud Native 就是好的，不符合就是差的，当不符合时，你的理由是什么？你站在问题的哪个角度？

Martin Fowler 曾说：“优秀的技术人员的观点胜过任何度量，尽管它是主观的。”

因为你无法统一每个人关注的点，以及对各自关注的点的重视程度，所以架构很难被衡量。

架构需要持续演进

在传统企业中，架构设计是一个很重要且很耗费时间的过程，需要经过很多轮审核，架构文档动辄几百页，而且这个文档绝对不能有没考虑到的问题，必须面面俱到、接近完美。例如，目前系统还没有用户，就要为未来 1 千万的用户耗费精力解决性能问题，而且软件永远有你想象不到的问题发生。实际上我们描述的是一种静止的架构，这种架构每次变更都需要耗费巨大的成本。如果此时恰好出现了一个基于敏捷思想的竞争对手，则会形成一种鲜明的对比，他们不去考虑太长时间之后的事，出现什么问题就解决什么问题，因为有可能一年以后这个项目死了，也有可能用户人数突破 1 亿，系统需要进行大规模重构。总之，未来是不确定的。可见，架构是锤炼出来的，而不仅是设计出来的。

反应速度是传统企业的硬伤，这不是通过加班就能解决的。可以看一下互联网巨头们每年的发布次数，动辄每年发布几百万乃至上千万次，每个服务每天都在发生变化，每周可能都会上线。在如今这个快速发展的世界里，你无法依赖一个人去做所有的决策。这就需要发挥所有成员的主观能动性，也就是说，架构应该交给一线决策。回到前面提到的问题，服务怎么拆分更好？我想只有深入了解需求、场景、目标甚至自身条件之后才能做出决策。并且，架构的演进不是一蹴而就的，而是一个长期发展的过程。

变革需要坚决

历史上的变革大多阻力重重，因为一旦变革就意味着打破原有的“默契”，打破原有的“潜规则”，而“顽固派”通常是原有文化的受益者，他们通常不会反对变革，而是通过“我们不能完全照抄，要走出适合我们的路”来促成妥协。如果变革过程中遇到任何风吹草动，就更会给“顽固派”各种理由“走自己的路”。这也就是为什么我们熟知世界领先 IT 企业的技术、研发流程和企业文化，而就是学不会的原因。

这时候需要的是企业领导者的果断、坚决。只要方向没错，就要坚持，决不动摇。下面这段话是马云对刘振飞（阿里技术保障部负责人）关于阿里云内部争议的回复，反映了一个领导者在企业变革过程中起到的作用。

> 在王坚加入阿里之前，我跟教授（指曾鸣）讨论公司的未来，觉得云计算和大数据代表未来，对国家和社会的发展有长远的意义，所以我们要干，这是第一点。但是怎么做云计算、大数据？我们谁也不知道。现在来了个人叫王坚，他说：“我知道怎么做”，为什么不支持呢？这是第二点。第三点，即使万一做失败了，那也没关系，咱们的人倒下 70%，还有 30%活着，咱们活下来的人打扫战场，换个方向继续干，总要把它做出来。

写代码不同于搬砖

如果是搬砖，那么效率高的人和效率低的人之间的差距不会太大，因此每个人每天的工资都是相对固定的。但是在如今这个知识爆炸的时代，对于从事软件行业的群体来说，效率高者的工作效率比效率低者的可能高出几十倍、几百倍，优秀的人能写出更高质量的代码，能够预测问题。而在这个行业越是优秀的人才越是稀缺，因此很多互联网公司都愿意花大价钱去招一些更优秀的人。

优秀的人不愿意来，不一定是因为钱。花钱雇佣优秀的人是一方面，怎样管理这些人又是另外一方面，用管理搬砖者的方式来管理他们是不行的，管理优秀的人需要给予他们更多的信任，需要营造一种公开透明、自由高效的环境。

关于本书

为什么会出现 Cloud Native 这个概念呢？无论是云化、平台化，还是微服务架构，又或者是敏捷开发、自动化，都只是描述了几个点，而 Cloud Native 更像是一个面，通过它把这些点都关联起来了。某几个点做得很好而忽略了其他点通常会走入误区。例如，某些团队只关注服务拆分，而忽略了工具、组织对微服务的影响，最终效果并不理想。又如，要提升系统的可用性，只是从技术的角度去考虑是不够的，还要考虑如何通过自动化测试提升可用性，如何通过 Code Review 提升可用性，以及当故障发生时如何快速修复。我希望通过个人的工作经历以书的方式传递一些这方面的经验教训。

本书分别从架构、研发流程、团队文化三个角度全面论述 Cloud Native，因为只有三方面配合才能达到理想的效果。我见到过无数失败的案例，绝大多数都是因为考虑得比较片面，例如单纯从架构角度进行变革，或者单纯从研发流程角度变革。我们希望模仿 Google、Facebook、Amazon、Netflix 等领先企业，但是往往高估了架构的影响力，而低估了研发流程和团队文化的影响力。实际上，研发流程和团队文化对架构有着非常重要的影响。本书以 Cloud Native 的起源、诉求及组成开始，全面描述了 Cloud Native 的各个方面。从架构角度阐述了如何实施微服务架构，如何构建敏捷基础设施及平台服务。同时，从可用性、可扩展性、性能、一致性等角度描述了微服务架构中产生的问题及解决方案。最后，分别描述了 Cloud Native 下的研发流程和团队文化。

本书比较适合技术管理者、架构师和有一定基础的技术人员阅读，特别是传统软件企业的技术领导者，以及希望向互联网公司转型的或者转型失败的企业技术领导者。此书将帮助这些人少走弯路。还有一些比较有经验的高级研发人员，阅读此书也利于系统掌握 Cloud Native 的关键技能。无论如何，都希望此书能给你带来较好的体验，使你获得启发。

书中的内容大多来自我的工作经验，不免存在遗漏及错误，欢迎指正。读者可以直接

发送邮件到我的邮箱（41309975@qq.com），在此提前表示感谢。

感谢工作中的各位同事、生活中的各位好友，正是他们的支持和鼓励，才让本书完成。更要感谢我的家人，特别是我的妻子和女儿，是她们的拥抱和灿烂的笑容让我坚定了信念。

王启军

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。

- **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/35120*

目　录