



DevOps

软件架构师行动指南

[澳] 伦恩·拜斯 (Len Bass) 英戈·韦伯 (Ingo Weber) 朱黎明 (Liming Zhu) 著
胥峰 任发科 等译

DevOps
A Software Architect's Perspective

为软件架构师所写的第一本DevOps完全指南
通过3个经典案例研究，系统讲解在不同场景下应用DevOps实践的方法
盛大游戏和唯品会的资深DevOps专家联合翻译



机械工业出版社
China Machine Press



架构师书库

DevOps

A Software Architect's Perspective

DevOps

软件架构师行动指南

[澳] 伦恩·拜斯 (Len Bass) 英戈·韦伯 (Ingo Weber) 朱黎明 (Liming Zhu) 著

胥峰 任发科 等译



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

DevOps : 软件架构师行动指南 / (澳) 伦恩·拜斯 (Len Bass) 等著; 胥峰等译. —北京: 机械工业出版社, 2017.3 (2017.6 重印)

(架构师书库)

书名原文: DevOps: A Software Architect's Perspective

ISBN 978-7-111-56261-0

I. D… II. ① 伦… ② 胥… III. 软件工程 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2017) 第 039927 号

本书版权登记号: 图字: 01-2015-5197

Authorized translation from the English language edition, entitled DevOps: A Software Architect's Perspective, 9780134049847 by Len Bass, Ingo Weber, Liming Zhu, published by Pearson Education, Inc., Copyright © 2015.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

Chinese simplified language edition published by Pearson Education Asia Ltd., and China Machine Press Copyright © 2017.

本书中文简体字版由 Pearson Education (培生教育出版集团) 授权机械工业出版社在中华人民共和国境内 (不包括香港、澳门特别行政区及台湾地区) 独家出版发行。未经出版者书面许可, 不得以任何方式抄袭、复制或节录本书中的任何部分。

本书封底贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

DevOps: 软件架构师行动指南

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 盛思源

责任校对: 董纪丽

印刷: 中国电影出版社印刷厂

版次: 2017 年 6 月第 1 版第 3 次印刷

开本: 186mm × 240mm 1/16

印张: 16.5

书号: ISBN 978-7-111-56261-0

定价: 69.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

.. 译者序 ..

为什么要翻译这本书

2016年7月中旬，机械工业出版社华章公司王春华老师问我有没有兴趣翻译一本 DevOps 实践的书，在看过这本书的英文版之后，我欣然答应了。

因为，我知道这是一本有重要价值的书，我愿意花时间把它翻译出来，我想让它被更多人研读学习。

自从在 Agile 2008 大会上讨论 DevOps 一词以来，DevOps 理念得到了 IT 从业者的认可和追捧。数年之后，再次客观地审视 DevOps 运动时，我们发现从整个信息技术领域来看，DevOps 还没有成为开发和运维的事实标准。

原因何在？

我想，其中最重要的一个原因是，很多组织缺少在业务中引入 DevOps 实践的专家知识，也没有很多可以借鉴的成功案例。而这本书正好可以解决这个问题。

本书从软件架构师的视角讲解了引入 DevOps 实践所需要掌握的技术能力，涵盖了运维、部署流水线、监控、安全与审计以及质量关注。第四部分通过 3 个经典案例研究，讲解了在不同场景下应用 DevOps 实践的方法。这对于想应用 DevOps 实践的组织具有切实的指导意义。

初识本书，我有一种相见恨晚的感觉。研读之后，这种感觉尤甚。相信读者也一定会有这样的感觉。

衷心希望本书中文版能给读者带来有价值的信息，为推动 DevOps 进一步发展和应用做出贡献。

读者对象

本书适合以下几类读者阅读：

- 软件架构师
- 中高级运维工程师
- 计算机相关专业的学生
- 希望提高 IT 生产力的人员

勘误支持

虽然译者试图努力保证本书翻译中不出现错误，但鉴于译者的知识和视角，书中难免出现用词错误、技术适用性的问题。在此，译者恳请读者不吝指教，指出错误。请读者发送邮件到 xufengnju@163.com 或者加入 QQ 群 434242482，帮助译者修正错误。本书的勘误将在网站 <http://xufeng.info/> 刊载。

译者说明

本书由来自盛大游戏、唯品会等互联网企业的资深 DevOps 专家联合翻译完成。

除胥峰、任发科外，参与翻译的人员还有米全喜、吴昊、黄灵等。全书由胥峰完成统稿和技术审校。

致谢

感谢华章公司引进了本书的中译本版权，这是本中译本得以面市的最核心要素。

感谢华章公司盛思源、关敏和王春华老师，你们专业的编辑能力为本书提供了重要的质量保证。

感谢盛大游戏各位领导对我翻译本书的关心和支持。

感谢我的妻子吕宁和可爱的女儿胥欣，谢谢你们的支持和理解。

胥峰
2016年10月

.. 前 言 ..

多年以来，我们一直在探索研究运维中的问题。自然而然地，我们也一直在追踪 DevOps 运动。它正在沿着 Gartner 成熟度曲线向上发展。这种现象有着坚实的业务原因。我们能够找到从信息技术经理视角对 DevOps 的探讨（例如小说《凤凰项目：一个 IT 运维的传奇故事》），也能找到从项目经理视角对 DevOps 的探讨（例如《持续交付：发布可靠软件的系统方法》）。而且，关于文化变革和文化变革对于打破组织单元之间障碍的意义，也有大量的资料。

然而，令我们感到失望的是，鲜有从软件架构师角度对 DevOps 探讨的资料。把运维人员视为首要干系人并且聆听他们的需求当然是非常重要的。使用工具来支持运维和项目管理也是非常重要的。然而，我们强烈感觉到，DevOps 并不仅仅是关于干系人管理和工具使用的。

事实上，正是因为目前缺少这方面的资料，所以我们才决定撰写本书，把这些缺失的内容填充起来。DevOps 与设计、过程、工具化和组织结构之间的相互作用令人着迷。我们试图回答两个主要问题：作为软件架构师，我需要做出什么样的技术决策才能够实现 DevOps 目标？DevOps 领域的其他参与者会对我产生什么样的影响？

以上这些问题的答案是，实现 DevOps 目标可能包含着对你的系统架构和角色以及责任做出重大变更，在把系统部署到生产环境以及在生产环境中支持这些系统时，需要以上角色及责任。

正如软件架构师必须理解他们所设计与构建的系统的业务情境和目标一样，理解 DevOps 也要理解组织和业务情境，同时也要理解技术和运维情境。我们探索了所有这些内容。

本书的主要读者是那些曾经被问过或者将被问到“这个项目或者组织是否应该采用 DevOps 实践”这个问题的执业软件架构师。甚至他们不是被追问这个问题，而是他们有可能被告知是否应该采用 DevOps 实践。与所有书一样，我们期望还有其他类型的读者。那些有兴趣学习更多关于软件架构实践的学生应该可以在本书中找到一些感兴趣的资料。那些想探索研究 DevOps 课题的研究人员可以在本书中找到重要的背景资料。然而，我们的主要读者还是那些执业软件架构师。

概述

在本书的开始，我们讨论了 DevOps 的背景。在第一部分中，我们研究了 DevOps 的目标以及期望使用 DevOps 来解决的问题。我们会涉及组织和文化问题，也会涉及 DevOps 实践和敏捷方法学的关系。

在第 2 章中，我们研究了云。随着云即平台（Cloud as a Platform）的增长，DevOps 实践也得到了发展。理论上，DevOps 和云即平台是可分开的，但是事实上，虚拟化和云是推动 DevOps 实践的重要力量。

在介绍背景知识的最后一章（第 3 章）中，我们按照信息技术基础设施库（Information Technology Infrastructure Library, ITIL）的理论研究了运维。ITIL 是对运维组的最重要功能进行组织的体系。不是所有的运维工作都包含在 DevOps 实践中，但是理解运维组的职责却会提供重要的上下文信息，特别是当需要理解角色和职责时。

第二部分描述了部署流水线（Deployment Pipeline）。在这一部分中，我们首先探索了微服务架构风格，这出现在第 4 章中。为了应用 DevOps 实践而以微服务风格进行系统架构并不是必需的，但是，却能用微服务架构风格来解决 DevOps 所需要解决的一些问题。

在第 5 章中，我们简略介绍了构建和测试过程以及工具链。理解这些内容很重要，但它们不是我们的焦点。我们介绍了把系统部署到生产环境所关联的各种不同环境以及在环境上执行的不同类型的测试。因为在 DevOps 中用到的很多工具也在构建和测试过程中被用到了，所以我们提供了理解这些工具 and 如何控制它们的上下文信息。

我们以讨论部署结束第二部分。DevOps 的目标之一是加速部署。实现这个目标的一种技术是，使得每个开发团队可以在完成准备后独立地部署他们的代码。独立部署带来了许多一致性的问题。我们将讨论不同的部署模型；管理同时运行在生产环境中不同版本的系统；在发生错误时回滚；以及与实际把系统部署到生产环境中有关的其他主题。

第二部分提供了从功能视角观察部署实践的内容。然而，与其他任何系统一样，质量视角常常是控制系统设计和系统接受度的主要因素。在第三部分中，我们聚焦在横切关注点上。我们首先在第 7 章中讨论了监控和现场测试（live testing）。现代软件测试实践在将系统部署到生产环境中以后并没有结束。首先，系统被广泛地监控以检测问题；其次，在将系统部署到生产环境以后，测试正以各种形式继续进行。

另一个横切关注点是安全。我们在第 8 章中讨论了该内容。我们展示了环境中各种不同类型的安全控制，包括组织范围和特定系统范围的安全控制。我们讨论了与实现安全相关联的不同角色以及在安全审计案例中如何评估这些角色。

安全不是质量关注的全部。在第 9 章中，我们讨论了与 DevOps 实践有关的其他方面的质量。我们讨论的主题包括部署流水线的性能、可靠性和可修改性。

第三部分的最后，我们在第 10 章中讨论了业务关注。像 DevOps 这种涉及面广泛的

实践，如果得不到来自管理层的支持，是无法被采用的。业务计划是寻求这种支持的一种典型方法。因此，我们描述了为引进 DevOps 所需要准备的业务计划的组成元素，然后讨论了如何进行论证、推出和测量业务计划。

在第四部分中，我们描述了 3 个案例研究。这些实施了 DevOps 实践的组织给我们讲解了他们所采用的一些技巧。第 11 章讨论了为了实现业务连续性如何维护两个数据中心；第 12 章展示了一个持续部署流水线的细节；第 13 章描述了一个组织是如何迁移到微服务架构上的。

作为本书的结尾，我们在第五部分展望了 DevOps 的未来。第 14 章描述了我们的研究以及它是如何基于把运维视作一系列过程来进行的。第 15 章预测了未来 3 ~ 5 年 DevOps 将如何发展。

致谢

本书离不开众多人的帮助。在此，感谢 Chris Williams、John Painter、Daniel Hand 和 Sidney Shek，他们为案例研究做出了贡献。同时，感谢 Adnene Guabtni、Kanchana Wickremasinghe、Min Fu 和 Xiwei Xu 在有些章节中提供的帮助。

感谢 Manuel Pais 帮助我们编排了案例研究。感谢 Philippe Kruchten、Eoin Woods、Gregory Hartman、Sidney Shek、Michael Lorant、Wouter Geurts 和 Eltjo Poort 对本书做出了评论或者为本书的某些方面做出了贡献。

下列人士对第 13 章做出了评论，在此表示感谢：Jean-Michel Lemieux、Greg Warden、Robin Fernandes、Jerome Touffe-Blin、Felipe Cuozzo、Pramod Korathota、Nick Wright、Vitaly Osipov、Brad Baker 和 Jim Watts。

在出版阶段，Addison-Wesley 出版社的编辑用他们一贯的专业和高效的工作为本书顺利出版做出了贡献。他们的专业知识和工作经验提升了本书的质量。

最后，感谢 NICTA 和 NICTA 的管理层。NICTA 是由澳大利亚政府通过通信部 (Department of Communications) 和由澳大利亚研究委员会 (Australian Research Council) 通过 ICT 卓越中心 (Centre of Excellence) 项目组建的。如果没有他们的慷慨支持，本书将难以完成。

图例说明

对于图，我们使用了 4 种不同的图例。我们使用架构标注法来识别我们使用的关键架构概念；使用业务流程建模标注 (Business Process Model and Notation, BPMN) 来描述一些过程；使用波特价值链标注 (Porter's Value Notation) 来描述一些其他的过程；使

用 UML 时序图来组织活动的顺序。这里没有给出 UML 时序图标注。我们给出了通过其他方法进行的标注。

架构



图 P-1 人员，包括个体和组



图 P-2 组件（运行时实体）、模块（代码时实体集合）和数据流

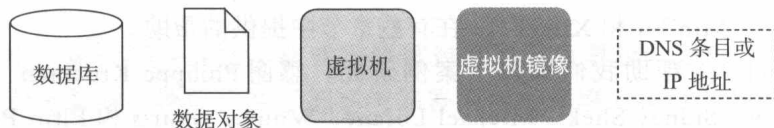


图 P-3 特殊化的实体

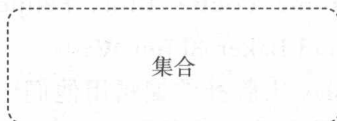


图 P-4 实体集合

业务流程建模标注

我们使用业务流程建模标注来描述事件和活动 [OMG 11]。



图 P-5 事件标识

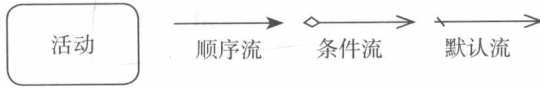


图 P-6 活动和活动序列

波特价值链

该标注用来描述过程（包含在业务流程建模标注中建模的活动）。

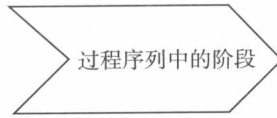


图 P-7 价值链中的条目

.. 目 录 ..

译者序
前言

第一部分 背 景

- 第 1 章 DevOps 是什么 2
 - 1.1 概述 2
 - 1.1.1 定义 DevOps 2
 - 1.1.2 DevOps 实践 3
 - 1.1.3 持续部署的例子：IMVU 5
 - 1.2 为什么是 DevOps 5
 - 1.2.1 发布过程 5
 - 1.2.2 配合不佳的原因 7
 - 1.2.3 运维人员能力有限 7
 - 1.3 DevOps 视角 8
 - 1.3.1 自动化 8
 - 1.3.2 开发团队的职责 9
 - 1.4 DevOps 与敏捷 9
 - 1.5 团队结构 10
 - 1.5.1 团队规模 10
 - 1.5.2 团队角色 10
 - 1.6 协作 13
 - 1.6.1 协作的形式 13
 - 1.6.2 团队协作 14
 - 1.6.3 跨团队协作 14
 - 1.7 障碍 15

- 1.7.1 文化及组织类型 15
- 1.7.2 部门类型 16
- 1.7.3 筒仓思维方式 (Silo Mentality) 17
- 1.7.4 工具支持 17
- 1.7.5 人员问题 17
- 1.8 小结 18
- 1.9 更多阅读材料 18

第 2 章 云即平台 20

- 2.1 概述 20
- 2.2 云的特性 21
 - 2.2.1 虚拟化 22
 - 2.2.2 IP 和域名系统管理 23
 - 2.2.3 平台即服务 25
 - 2.2.4 分布式环境 25
- 2.3 独特的云特性对 DevOps 的影响 30
 - 2.3.1 环境 30
 - 2.3.2 轻松创建虚拟机 31
 - 2.3.3 数据考量 31
- 2.4 小结 32
- 2.5 更多阅读材料 33

第 3 章 运维 34

- 3.1 概述 34
- 3.2 运维服务 34
 - 3.2.1 供给硬件 34
 - 3.2.2 供给软件 35
 - 3.2.3 IT 功能 36
 - 3.2.4 服务级别协议 36
 - 3.2.5 容量规划 36
 - 3.2.6 业务连续性和安全 37
 - 3.2.7 服务策略 38
 - 3.2.8 服务设计 39
 - 3.2.9 服务移交 39

- 3.2.10 服务运维 40
- 3.2.11 服务运维概念 40
- 3.3 服务运维功能 41
- 3.4 持续服务改进 42
- 3.5 运维和 DevOps 43
- 3.6 小结 44
- 3.7 更多阅读材料 44

第二部分 部署流水线

第 4 章 整体架构 48

- 4.1 DevOps 实践是否需要架构调整 48
- 4.2 架构结构总览 49
 - 4.2.1 协作模式 50
 - 4.2.2 资源管理 51
 - 4.2.3 架构元素之间的映射 52
- 4.3 微服务架构的质量 52
 - 4.3.1 可靠性 53
 - 4.3.2 可修改性 54
- 4.4 团队的亚马逊规则 55
- 4.5 现有系统的微服务方案 56
- 4.6 小结 56
- 4.7 更多阅读材料 57

第 5 章 构建与测试 58

- 5.1 概述 58
- 5.2 在部署流水线中移动系统 59
 - 5.2.1 可追溯性 59
 - 5.2.2 环境 60
- 5.3 横切关注点 61
- 5.4 开发及提交前测试 63
 - 5.4.1 版本控制与分支 63
 - 5.4.2 功能开关 65
 - 5.4.3 配置参数 66

- 5.4.4 在开发和提交前测试中的测试 67
- 5.5 构建与集成测试 67
 - 5.5.1 构建脚本 67
 - 5.5.2 打包 68
 - 5.5.3 持续集成与构建状态 69
 - 5.5.4 集成测试 70
- 5.6 用户验收测试 / 预发布 / 性能测试 70
- 5.7 生产环境 71
 - 5.7.1 早期发布测试 71
 - 5.7.2 错误检测 72
 - 5.7.3 现场测试 72
- 5.8 事件 73
- 5.9 小结 73
- 5.10 更多阅读材料 74

第 6 章 部署 75

- 6.1 概述 75
- 6.2 部署管理的策略 76
 - 6.2.1 蓝 / 绿部署 76
 - 6.2.2 滚动升级 77
- 6.3 逻辑一致性 78
 - 6.3.1 相同服务的多个版本同时存在 78
 - 6.3.2 兼容数据库中保存的数据 81
- 6.4 打包 82
- 6.5 多环境部署 84
- 6.6 部分部署 86
 - 6.6.1 金丝雀测试 86
 - 6.6.2 A/B 测试 87
- 6.7 回滚 87
- 6.8 工具 89
- 6.9 小结 90
- 6.10 更多阅读材料 90

第三部分 横切关注点

第 7 章 监控 94

7.1 概述 94

7.2 监控什么 95

7.2.1 故障检测 96

7.2.2 性能下降检测 96

7.2.3 容量规划 97

7.2.4 用户交互 98

7.2.5 入侵检测 99

7.3 如何监控 99

7.3.1 基于代理的监控和无代理的监控 101

7.3.2 监控运维活动 102

7.3.3 收集和存储 102

7.4 什么时候变更监控配置 103

7.5 解释监控数据 103

7.5.1 日志 104

7.5.2 绘图和展示 105

7.5.3 警报和警告 105

7.5.4 诊断和反应 106

7.5.5 监控 DevOps 过程 106

7.6 挑战 107

7.6.1 挑战 1: 持续变更下的监控 107

7.6.2 挑战 2: 自下向上与自上向下和在云中的监控 108

7.6.3 挑战 3: 监控微服务架构 109

7.6.4 挑战 4: 处理大容量的分布式 (日志) 数据 109

7.7 工具 109

7.8 从监控数据中诊断出异常——Platformer.com 的案例 110

7.8.1 背景 111

7.8.2 数据收集 112

7.8.3 检测异常 112

7.8.4 思考 113

7.9 小结 113

7.10 更多阅读材料 114

第 8 章 安全与安全审计 115

- 8.1 安全是什么 115
- 8.2 威胁 117
- 8.3 需要保护的资源 118
- 8.4 安全角色和活动 120
- 8.5 身份管理 122
 - 8.5.1 认证 123
 - 8.5.2 授权 125
- 8.6 访问控制 126
 - 8.6.1 阻止访问 127
 - 8.6.2 谁负责预防控制 129
- 8.7 检测、审计和拒绝服务 129
- 8.8 开发 130
- 8.9 审计者 130
- 8.10 应用设计考虑 131
- 8.11 部署流水线设计考虑 132
- 8.12 小结 133
- 8.13 更多阅读材料 134

第 9 章 其他非功能需求 135

- 9.1 概述 135
- 9.2 可重复性 136
 - 9.2.1 在恰当的层级上定义和执行过程 136
 - 9.2.2 版本控制所有事物 138
- 9.3 性能 139
 - 9.3.1 测量重要的事物 139
 - 9.3.2 提高资源使用率 140
- 9.4 可靠性 141
 - 9.4.1 理解不同服务的可靠性特性 141
 - 9.4.2 早期检测和修复错误 142
- 9.5 可恢复性 142
- 9.6 互操作性 143
 - 9.6.1 注意接口的互操作性 143
 - 9.6.2 理解现有的数据模型 143

- 9.7 可测试性 144
- 9.8 可修改性 145
 - 9.8.1 一个工具内的修改 145
 - 9.8.2 工具之间交互行为的修改 146
- 9.9 小结 146
- 9.10 更多阅读材料 147

第 10 章 业务关注点 148

- 10.1 概述 148
- 10.2 业务案例 148
 - 10.2.1 问题和解决问题所带来的好处 149
 - 10.2.2 成本 149
 - 10.2.3 干系人影响 150
 - 10.2.4 风险及其减缓 151
 - 10.2.5 推出计划 153
 - 10.2.6 成功标准 154
- 10.3 度量和对 DevOps 实践的合规性 155
 - 10.3.1 测量 DevOps 实践的成功度 155
 - 10.3.2 测量对 DevOps 实践的合规性 156
 - 10.3.3 测量干系人的满意度 157
- 10.4 Dev 和 Ops 之间的交互点 157
 - 10.4.1 许可 157
 - 10.4.2 事故处理 158
- 10.5 小结 159
- 10.6 更多阅读材料 159

第四部分 案例研究

第 11 章 支持多数据中心 162

- 11.1 概述 162
- 11.2 当前的情况 163
- 11.3 业务逻辑和 Web 层 163
 - 11.3.1 应用逻辑 163
 - 11.3.2 基础设施 164