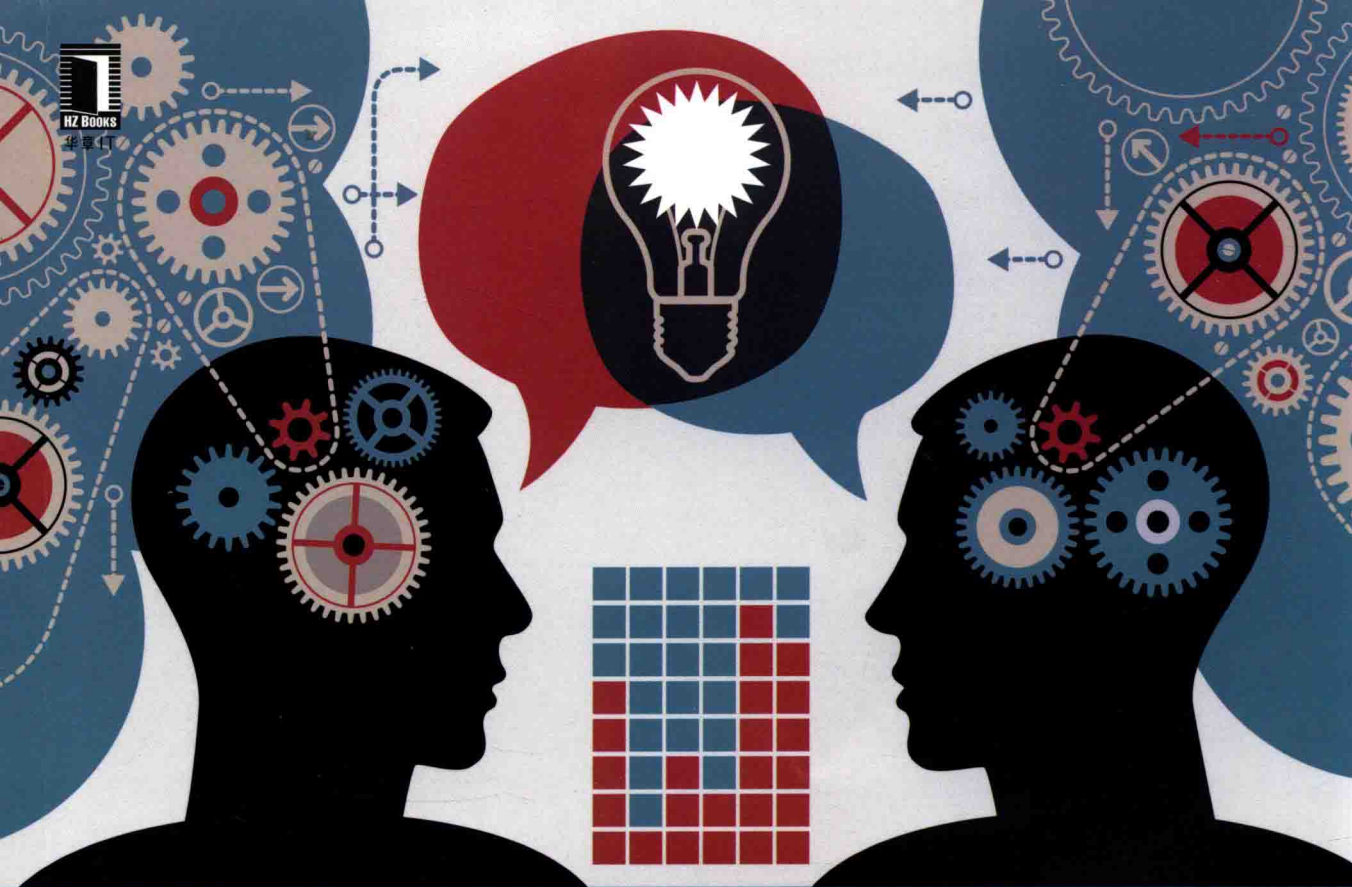


HZ BOOKS
华章



DevOps

原理、方法与实践

荣国平 张贺 邵栋 等编著

DevOps

Theory, Method and Practice

深度解析互联网时代开发运维模式DevOps
每一个想学习DevOps理论与实践的人都应该研读
工业界和学术界的一次思想碰撞——实用具体和严谨全面的有机融合

李兵 刘超 王青
汪浩 洪绯 王润
郭峰 徐峰 郭蕾
吴超英 沈海峰 黄丽果

联袂力荐



机械工业出版社
China Machine Press

非虚构

DevOps
Theory, Method and Practice

DevOps

原理、方法与amp;实践

荣国平 张贺 邵栋 王天青 任群 腾灵灵 宋骏 蒋孟杰 何勉 吴昊 编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

DevOps: 原理、方法与实践 / 荣国平等编著. —北京: 机械工业出版社, 2017.10

ISBN 978-7-111-58191-8

I. D… II. 荣… III. 软件工程—教材 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2017) 第 248508 号

本书是第一本全面系统介绍 DevOps 方法和实践的教材, 尽可能覆盖 DevOps 这种互联网时代新型开发模式的理论、方法、实践以及工具等多个方面。全书第一部分首先从时代背景出发, 介绍 DevOps 模式的渊源, 然后结合云时代运维的特征, 进一步阐明 DevOps 模式是适应当前软件系统开发、部署和维护的必然选择; 第二部分介绍了主流软件开发方法及其流程, 重点关注精益生产和看板方法; 第三部分则阐述了 DevOps 模式下的典型实践, 例如微服务架构、持续集成、持续交付(部署)、虚拟化、Docker 容器、自动化等。

本书可以作为高等院校软件工程或者计算机专业高年级本科生及研究生的教材使用, 也适合软件产业的研发和运维人员参考阅读。

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: 和 静

责任校对: 殷 虹

印 刷: 北京市荣盛彩色印刷有限公司

版 次: 2017 年 11 月第 1 版第 1 次印刷

开 本: 186mm×240mm 1/16

印 张: 15.75

书 号: ISBN 978-7-111-58191-8

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) 88378991 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzjsj@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

云计算时代，软件向网路化、服务化、平台化、生态化以及智能化方向发展，持续集成、持续交付、持续部署成为常态，软件的开发、运营和质量保证之间的协作与整合成为亟待解决的难题。

本书是目前国内（也可能是国际范围内）第一本 DevOps 的系统性教材，从价值观、方法论等角度对上诉问题进行剖析并提供技术指导。作者以云计算、微服务等新兴领域为研究载体，聚焦其中的前沿问题与关键技术，结合 DevOps 的应用开发案例与工具集，深入浅出、娓娓道来，既引人入胜，又发人深省，是一本不可多得的软件工程教材，既适合软件工程以及相关专业的学生学习，也有助于软件企业技术人员阅读。书中配设的大量图片与文字相得益彰，将会受到在“读图时代”成长起来的青年学生欢迎。

——李兵，武汉大学国际软件学院副院长

开发、技术运维及质量管理部门协作的系统化管理日益成为高速运营的企业关注的焦点，需要运用有效的最佳实践以支持敏捷、有序且高效地交付产品的需求，从而达成业务目标。管理与技术相结合，通过持续学习与回顾、持续过程改进来确保公司竞争力。本书将高等教育与社会需求对接，全面介绍了支持 DevOps 落地的基本方法，聚集了既规范又敏捷的最佳实践，供读者参考选用，不愧是及时雨。

——吴超英，北京航空航天大学软件工程研究所副教授，软件工程与过程管理顾问

应用需求的日益翻新和快速响应要求，促使各型企业加速整合其面向各种服务的敏捷开发、快速上线运营、持续维护和升级的全流程工作模式。为此，来自不同国度的企业精英们在凝练成功实践基础上，发展出了近年来广受企业关注的高效工作方式 DevOps，以支持敏捷和持续的应用开发与运维。这本书全面地介绍了 DevOps 的由来和发展，遵循的基本原则，相关的理论基础，推荐的有效方法和实践，以及支持平

台和工具，为在校学生和企业开发与运维相关人员提供了一本内容丰富、易读易懂的教材。

——刘超，北京航空航天大学软件工程研究所所长

本书从理论、技术、工具、方法及实践等多个角度对 DevOps 进行了全面诠释，在内容上引入了多个组织采用 DevOps 的成功案例，凝聚了作者和众多行业专家多年的实践积累与思想结晶。对于众多正在计划和已经采用 DevOps 开发模式的组织来说，本书是一本不可多得的权威参考。

——汪浩，中金云金融（北京）大数据科技股份有限公司研发总监

近些年，大型互联网应用如雨后春笋般地进入市场，它们的用户增长也是一样的惊人。工业界一直在探索新的应用软件开发方法使得它们的产品能更快地进入市场，更加可靠，开发及运维成本更低。DevOps 因为正好符合这些要求而在工业界越来越受推崇。学术界也展现出浓厚的兴趣去探索研究 DevOps 的理论、实践及工具。因此，让学生有机会系统地学习 DevOps 是一件非常有意义的事。虽然市面上有关 DevOps 的书籍不少，但大都是英文版的，而且不够系统全面，不适合作为教材。本书弥补了这个空白。它系统地阐述了 DevOps 的历史演变以及与之相关的技术、理论、实践及工具。虽然云计算、软件架构、软件开发过程、容器技术等都有专门的书籍，但这些概念围绕 DevOps 出现在同一本书是一件很有挑战性的尝试。本书值得推荐给软件工程及相关专业的学生以及对 DevOps 感兴趣的业界人士。

——沈海峰，澳大利亚弗林德斯大学高级讲师

在移动互联网高度竞争的今天，创新、速度、质量决定了一款产品的成败。本书深入浅出地剖析了 DevOps 模式如何具体应用到产品研发中，当我们面对日益复杂的业务场景和技术发展时，能提供更为高效的方法和策略以提升综合竞争力。

——洪绯，阿里巴巴土豆视频产品研发副总裁

DevOps 概念的提出到现在已经有近 10 年时间，随着云计算、容器、微服务等技术的发展，DevOps 也逐步开始在企业落地生根。InfoQ 在很早之前就向社区普及推广 DevOps 相关的实践，我们坚信 DevOps 将会对软件开发产生深远影响。本书是国内为数不多的系统讲解 DevOps 技术的书籍，推荐阅读。

——郭蕾，InfoQ 总编辑

在这个充满变革的时代，信息科技及系统在社会经济发展中扮演的角色越来越重要。DevOps 作为将软件开发、质量保证和技术操作集成的模式，在 IT 开发运维管理实践中日益广泛地得到接受和认可。本书全面且深入浅出地介绍、阐述并探讨了 DevOps 本身及其运用的重要维度和方面，特别就 DevOps 的初学者学习理论基础而言大有裨益。

——王润，普华永道合伙人

在当前数字化、网络化的大背景下，企业的商业成功越来越多地依赖于其交付软件业务的速度，DevOps 是对传统软件交付方式的一次革新，能够显著缩短交付周期，提高交付效率。但是，DevOps 并不是一种最佳实践或者工具集，它是一套全新的交付管理体系和方法论，如果缺乏体系化的学习和指导，很容易在实践中走弯路。在本书中，作者详细阐述了 DevOps 背后的软件交付理论体系，并结合当下热门的微服务架构设计和容器技术，使读者领略了前沿的交付设计和实践探索，相信读者读完这本书一定会受益匪浅。

——郭峰，DaoCloud 联合创始人兼首席技术官

DevOps 是敏捷和精益理念的进一步实践与延伸。从敏捷的角度看，DevOps 从开发的敏捷延伸到了运维的敏捷，我们通过一系列自动化的质量保障服务结合自动化流水线实现了 Dev 阶段与 Ops 阶段的无缝衔接；而从精益的角度看，DevOps 是组织和价值链的优化，组织上 DevOps 将开发工程师和运维工程师的界限逐步模糊，进而走向了全功能型的团队，价值链上产品、服务以及特性的 TTM 进一步缩减，使得企业的价值变现更加迅捷。

本书为大家清晰讲述了 DevOps 理念发展的历程，以及 DevOps 和敏捷、精益之间的联系，以实践项目的方式将软件架构、研发模式和研发工具结合起来，是一部非常值得大家去深入学习、探讨的 DevOps 教程。

——徐峰，华为软件开发云 CTO

我非常高兴看到这样一本介绍 DevOps 的教材出现，随着互联网技术的发展，软件的开发方法受到革命性的冲击，新的概念、方法和技术日新月异，同时也给学习和实践带来很多困惑：我们该遵循什么基本原则？我们可以获得什么价值？我们应该采取什么技术手段和方法？本书梳理了 DevOps 的发展历史和基本概念，并给出了一些技术和实践的方法，为初学者提供了很好的指导，衷心希望本书可以得到读者们的喜爱！

——王青，中国科学院软件研究所互联网软件技术实验室主任

软件开发者和其他相关干系人之间持续有力的协作有助于加快新的 IT 服务的交付，同时提升交付质量，更好地满足稳定性、性能以及安全性等质量目标。尽管 DevOps 运动早在 2009 年就开始了，然而，究竟什么是 DevOps？如何实施 DevOps？拥护者所宣传的 DevOps 的优点是否真实？这些问题一直都困扰着我们。本书是迄今为止我看到的最完整介绍 DevOps 概念、理论和实践的教材，很大程度上给出了上述问题的答案。基于最新的软件系统平台和过程方法，本书为 DevOps 的应用和实施提供了详细指南，具体包括云计算、微服务以及敏捷开发等。特别是书中的案例以及工具的详细介绍，对于在校学生以及工业界的实践者提炼 DevOps 知识要点，投入 DevOps 学习过程都有裨益。

——黄丽果，南卫理公会大学副教授

两年前，我出版了一本关于 DevOps 的著作。从那以后，我一直在留意各类关于 DevOps 的著述，关注着该领域的进展。但让我略感失望和不足的是，作为一位软件工程的研究者，我清楚地认识到，对于各种流行的 DevOps 实践、方法和技术背后的软件工程理论，我们缺少深入的理解和探索。这种基于软件工程视角对 DevOps 进行系统梳理的缺失，使得我们对软件工程本科生和研究生的教育变得更加困难。这是因为，我们的教育目的更多应着力于教导学生和实践者去理解 DevOps 背后的理论，并培养他们应用 DevOps 解决实际问题的能力，而不是去简单追逐最新的热门技术或者流行术语。

当我仔细通读本书时，立即感到了一种缺憾获得填补的兴奋。本书清晰、中肯地解释了 DevOps 背后的理论，建立了 DevOps 与软件工程中一系列子领域现有知识的关联，从需求到架构、测试、部署，还有开发过程。这一切并未让我感觉到意外，因为我和作者之一张贺教授已经相识并一起工作了十余年。张教授对软件开发过程的仿真和实证方面的开创性工作，使得他对软件开发过程中各个阶段都有着深入的理解，而这一切都在本书的结构、流程以及深度上得到了体现。

作为澳大利亚联邦科学院（CSIRO）的一员，我重点专注于工业界需求和问题的解决。作为许多工业咨询委员会的一员，我也参与了很多大学软件工程课程体系的构建。看到本书中阐述的一些具体知识和技术，例如微服务、轻量级容器以及 DevOps 工具包，包括 Chef/Puppet、Cucumber/Selenium 以及 Jenkins 等，我立刻有一种强烈的愿望，希望能将本书的大纲介绍给那些大学，让这些内容充实他们的软件工程教育课程。同时，我也坚信那些可能已经熟悉 DevOps 工具的工业界实践者可以从本书中获益良多。因为本书阐述了必要的上下文，能够帮助实践者对组织中使用这些 DevOps 工具和方法的原因有更深入的理解。对于这样一本系统介绍 DevOps 的书籍能够在中国出版，我由衷感

到高兴，这表明在 DevOps 运动和相关软件工程研究中，中国不仅是跟随者，也是引领者。

祝立明

Data61 研究所所长

澳大利亚联邦科学院 (CSIRO) 院士

《DevOps: A Software Architect's Perspective》作者

2017 年 8 月于悉尼

近年来 DevOps 开发模式对软件产业产生了深远影响，相当多的软件企业开始采用这种新的模式。来自权威机构的预测报告甚至认为，未来全球排名前 2000 的软件企业中，超过 80% 都将转向 DevOps 模式。事实上，DevOps 发展速度之快和影响范围之广都大大超出了人们的预期。

DevOps 之所以会产生如此巨大的影响，我们认为这不是偶然的。这种方法本身具有的特性非常适合在需求很难确定、快速响应变更、快速提供价值和高可靠性要求这样的所谓互联网时代软件环境中得到应用。由此，作为软件工程教育者，我们不得不思考 DevOps 给现代软件工程教育所带来的影响。一方面，我们的教育本身就需要教会学生把经过实践检验的优秀管理方法和适用的具体开发技术相结合，应用过程化思想和系统化的方法去开发和维护各类软件系统。从这个意义上说，DevOps 是满足上述目标的极好载体。因此，忽视 DevOps，不仅仅会错过一个实现软件工程教育目的的好机会，更为糟糕的是，可能会由此扩大学校教育和业界实际实践之间的差距。另一方面，将 DevOps 引入大学课堂，也面临着诸多挑战。首要的就是目前尚缺一本专门以 DevOps 为主题，全面系统地涵盖 DevOps 各个方面的教材。有鉴于此，我们编著了这样一本教材，试图弥补这一缺憾。

考虑到本书主要面向 DevOps 的初学者，因此，在内容的选择和组织上，我们并没有一味罗列 DevOps 的流行词汇和工具，而是尽可能把 DevOps 方法背后的理论基础解释清楚。对于具有典型 DevOps 特征的软件工程技术和实践，例如微服务架构演进、精益管理、容器技术等，我们则不吝笔墨地大篇幅介绍。由此，我们试图传递出一个概念，即 DevOps 作为一种方法学，不能简单等同于某类实践或者工具，而是涵盖管理的基础理论、技术以及工具的有机整体。综合起来，本书具有如下特点：

- 全面系统地覆盖 DevOps 的各个方面，便于读者以此书作为 DevOps 的基础（尽

管内容并不基础!)入门书籍。

- 保持客观、中立和审慎的态度。尽管我们推崇 DevOps, 但绝不盲从。在材料的组织和内容表述方面, 我们以解决实际问题为导向来介绍 DevOps。同时, 我们也清晰表达观点——DevOps 并不抗拒其他方法学。
- 部分知识点和相应的案例直接来源于一线业界专家的工作经历, 可以增强读者的代入感, 同时, 也有助于读者更好地理解 DevOps。

本书具体分工如下: 第 1 章由王天青、邵栋、张贺以及任群合作编写; 第 2 章由腾灵灵和宋骏合作编写; 第 3 章由蒋孟杰编写; 第 4 章由荣国平编写; 第 5 章由何勉编写; 第 6~8 章由王天青编写; 第 9 章由吴昊编写。此外, 荣国平、张贺和邵栋对全书进行统稿和润色。由于时间仓促, 再加上编者的水平所限, 书中难免有错误与不妥之处, 恳请读者指正和赐教。我们的电子邮件地址为: ronggp@nju.edu.cn、hezhang@nju.edu.cn 和 dongshao@nju.edu.cn。

荣国平

2017 年 7 月于南大北园

赞誉
序
前言

第 1 章 DevOps 概述 1

1.1 互联网时代的转型挑战 1

1.2 独角兽公司 4

1.2.1 Netflix 公司 4

1.2.2 Instagram 8

1.2.3 成功秘诀 9

1.3 什么是 DevOps 10

1.3.1 发展渊源 10

1.3.2 价值观 12

1.3.3 原则 13

1.3.4 方法 15

1.3.5 实践 16

1.3.6 工具 16

1.4 DevOps 应用与研究现状 17

1.4.1 微服务 17

1.4.2 持续集成和持续交付 19

1.4.3 工具研究和开发 21

本章小结 23

思考题 23

参考文献 23

第2章 云时代的运维 …… 26

2.1 云计算概述 …… 26

2.1.1 IaaS …… 26

2.1.2 PaaS …… 28

2.1.3 SaaS …… 29

2.1.4 XaaS …… 30

2.2 IT 服务标准介绍 …… 30

2.2.1 CMMI-SVC …… 30

2.2.2 ITIL …… 33

2.2.3 ISO20000 …… 33

2.2.4 ITSS …… 36

2.3 什么是运维 …… 39

2.3.1 运维的价值 …… 39

2.3.2 运维的技术与技能 …… 40

2.3.3 传统运维的转型之路 …… 40

本章小结 …… 41

思考题 …… 41

参考文献 …… 42

第3章 软件架构演进 …… 44

3.1 软件架构概述 …… 44

3.1.1 什么是软件架构 …… 44

3.1.2 软件架构的目标 …… 46

3.1.3 软件架构的不同视角 …… 46

3.2 软件架构的演进 …… 51

3.2.1 传统软件架构的演进 …… 51

3.2.2 流量爆炸时代的大型互联网软件架构 …… 54

3.2.3 互联网软件架构演进实例 …… 60

本章小结 …… 66

思考题 …… 67

参考文献 …… 67

更多阅读 …… 67

第 4 章 软件开发过程和方法	68
4.1 软件过程概述	68
4.1.1 软件开发方法发展历史	69
4.1.2 软件过程的多维视角	77
4.2 个体过程和实践	79
4.2.1 PSP 过程基本原则	80
4.2.2 PSP 过程度量	81
4.2.3 PROBE 估算原理	82
4.2.4 PROBE 估算流程	83
4.2.5 通用计划框架	85
4.2.6 PSP 质量与质量策略	87
4.2.7 评审与测试	88
4.2.8 评审过程质量	89
4.2.9 设计与质量	98
4.2.10 设计过程	100
4.2.11 设计的层次	101
4.3 小组过程和实践	102
4.3.1 XP 实践	102
4.3.2 Scrum 方法	105
4.3.3 TSP 过程	106
4.4 软件过程改进	108
4.4.1 元模型	108
4.4.2 过程改进参考模型与标准	112
4.5 DevOps 中的开发过程和方法	121
本章小结	121
思考题	121
参考文献	122

第 5 章 精益思想和看板方法 124

5.1 从精益思想说起	124
5.1.1 精益起源于丰田	124
5.1.2 精益实践的传播	124

5.1.3	精益作为方法学开始超越生产制造	126
5.1.4	上升至精益的价值观	126
5.2	精益的三个层面	128
5.3	精益产品开发实践体系	129
5.3.1	精益产品开发的目标	129
5.3.2	精益产品开发的原则	130
5.3.3	精益产品开发的运作实践	131
5.4	看板方法的起源	132
5.4.1	看板的中文意思带来误解	132
5.4.2	看板是精益制造系统的核心工具	133
5.4.3	看板形成拉式生产方式	135
5.5	什么是产品开发中的看板方法	136
5.5.1	产品开发中的看板方法的诞生	136
5.5.2	看板方法的第一组实践——建立看板系统的3个实践	136
5.5.3	看板方法的第二组实践——运作看板系统的2个实践	140
	本章小结	142
	思考题	142
	参考文献	143

第6章 微服务软件架构 144

6.1	软件架构的发展	144
6.1.1	单体架构	144
6.1.2	分层架构	144
6.1.3	SOA 架构	146
6.1.4	分布式架构	148
6.2	现代应用的12范式	150
6.3	什么是微服务架构	150
6.4	微服务架构的特征	151
6.4.1	通过服务组件化	152
6.4.2	围绕业务能力组织	152
6.4.3	是产品不是项目	153
6.4.4	智能端点和哑管道	153

6.4.5	去中心化治理	154
6.4.6	去中心化数据管理	154
6.4.7	基础设施自动化	155
6.4.8	为失效设计	155
6.4.9	进化式设计	155
6.5	微服务核心模式	156
6.5.1	服务注册与发现	156
6.5.2	配置中心	157
6.5.3	API 网关	157
6.5.4	熔断器	158
6.5.5	分布式追踪	160
	本章小结	161
	思考题	162
	参考文献	162

第 7 章 容器技术基础 163

7.1	内核基础	163
7.1.1	Linux namespace	164
7.1.2	Linux CGroup	168
7.2	Docker 架构概览	169
7.2.1	Client	170
7.2.2	Docker Daemon	171
7.2.3	Docker Registry	172
7.2.4	Graph	173
7.2.5	Driver	173
7.2.6	libcontainer	173
7.3	镜像管理	174
7.3.1	什么是 Docker 镜像	174
7.3.2	Dockerfile、Docker 镜像和 Docker 容器的关系	174
7.3.3	Dockerfile	176
7.4	Docker 网络管理	178
7.4.1	Docker 网络模式	178

7.4.2	libnetwork 和 Docker 网络	179
7.4.3	Docker 的内置 Overlay 网络	180
7.5	Docker 存储	181
7.5.1	Docker 存储驱动	181
7.5.2	Docker 驱动比较	182
7.6	Docker 编排	183
7.6.1	Docker Swarm	183
7.6.2	Kubernetes	185
	本章小结	186
	思考题	186
	参考文献	187

第 8 章 基于容器技术的 DevOps 实践 188

8.1	概述	188
8.2	代码管理	188
8.2.1	Git 介绍	190
8.2.2	Git 工作流程	191
8.3	持续交付流水线	194
8.3.1	预备步骤	194
8.3.2	实现持续交付流水线	199
8.3.3	持续交付最佳实践	201
8.3.4	检查列表	204
8.4	持续集成工具	205
8.4.1	传统的 CI 工具	205
8.4.2	云计算环境中的 CI 工具	206
8.4.3	用于移动应用的 CI 工具	206
8.4.4	使用 Docker 的 CI 工具	207
8.5	Java 应用持续交付实践举例	207
8.5.1	持续集成	207
8.5.2	持续部署	209
8.5.3	版本管理	211
	本章小结	212