

O'REILLY®

TURING

图灵程序设计丛书



# Kafka权威指南

Kafka: The Definitive Guide

Kafka核心作者和业界一流一线人员共同执笔  
全面介绍Kafka设计原理和架构细节

[美] Neha Narkhede Gwen Shapira Todd Palino 著

薛命灯 译



中国工信出版集团



人民邮电出版社  
POSTS & TELECOM PRESS

TURING

图灵程序设计丛书

# Kafka权威指南

Kafka: The Definitive Guide

[美] Neha Narkhede Gwen Shapira Todd Palino 著  
薛命灯 译

Beijing • Boston • Farnham • Sebastopol • Tokyo

O'REILLY®

O'Reilly Media, Inc. 授权人民邮电出版社出版

人民邮电出版社  
北京

## 图书在版编目 (C I P) 数据

Kafka权威指南 / (美) 妮哈·纳克海德  
(Neha Narkhede), (美) 格温·沙皮拉 (Gwen Shapira),  
(美) 托德·帕利诺 (Todd Palino) 著; 薛命灯译. --  
北京: 人民邮电出版社, 2018. 1  
(图灵程序设计丛书)  
ISBN 978-7-115-47327-1

I. ①K… II. ①妮… ②格… ③托… ④薛… III. ①  
分布式操作系统—指南 IV. ①TP316.4-62

中国版本图书馆CIP数据核字(2017)第290682号

## 内 容 提 要

本书是关于 Kafka 的全面教程, 主要内容包括: Kafka 相对于其他消息队列系统的优点, 主要是它如何完美匹配大数据平台开发; 详解 Kafka 内部设计; 用 Kafka 构建应用的最佳实践; 理解在生产中部署 Kafka 的最佳方式; 如何确保 Kafka 集群的安全。

本书适合 Java 开发人员、大数据平台开发人员以及对分布式系统感兴趣的读者阅读。

- 
- ◆ 著 [美] Neha Narkhede Gwen Shapira Todd Palino  
译 薛命灯  
责任编辑 朱 巍  
执行编辑 张海艳  
责任印制 彭志环
  - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号  
邮编 100164 电子邮件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京鑫正大印刷有限公司印刷
  - ◆ 开本: 800×1000 1/16  
印张: 14.5  
字数: 343千字 2018年1月第1版  
印数: 1-4000册 2018年1月北京第1次印刷  
著作权合同登记号 图字: 01-2017-6476号
- 

定价: 69.00元

读者服务热线: (010)51095186转600 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147号

---

# 版权声明

© 2017 by Neha Narkhede, Gwen Shapira, Todd Palino.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and Posts & Telecom Press, 2018. Authorized translation of the English edition, 2017 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版，2017。

简体中文版由人民邮电出版社出版，2018。英文原版的翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有，未得书面许可，本书的任何部分和全部不得以任何形式重制。

---

# O'Reilly Media, Inc.介绍

O'Reilly Media 通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自 1978 年开始，O'Reilly 一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly 的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly 为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了 *Make* 杂志，从而成为 DIY 革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly 的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly 现在还将先锋专家的知识传递给普通的计算机用户。无论是通过图书出版、在线服务或者面授课程，每一项 O'Reilly 的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

## 业界评论

“O'Reilly Radar 博客有口皆碑。”

——*Wired*

“O'Reilly 凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——*Business 2.0*

“O'Reilly Conference 是聚集关键思想领袖的绝对典范。”

——*CRN*

“一本 O'Reilly 的书就代表一个有用、有前途、需要学习的主题。”

——*Irish Times*

“Tim 是位特立独行的商人，他不光放眼于最长远、最广阔的视野，并且切实地按照 Yogi Berra 的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去，Tim 似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——*Linux Journal*

---

# 序

这是一个激动人心的时刻，成千上万的企业在使用 Kafka，三分之一多的世界 500 强公司也在其中。Kafka 是成长最快的开源项目之一，它的生态系统也在蓬勃发展。Kafka 正在成为管理和处理流式数据的利器。

Kafka 从何而来？我们为什么要开发 Kafka？Kafka 到底是什么？

Kafka 最初是 LinkedIn 的一个内部基础设施系统。我们发现，虽然有很多数据库和系统可以用来存储数据，但在我们的架构里，刚好缺一个可以帮助处理持续数据流的组件。在开发 Kafka 之前，我们实验了各种现成的解决方案，从消息系统到日志聚合系统，再到 ETL 工具，它们都无法满足我们的需求。

最后，我们决定从头开发一个系统。我们不想只是开发一个能够存储数据的系统，比如传统的关系型数据库、键值存储引擎、搜索引擎或缓存系统，我们希望能够把数据看成是持续变化和不断增长的流，并基于这样的想法构建出一个数据系统；事实上，是一个数据架构。

这个想法实现后比我们最初预想的适用性更广。Kafka 一开始被用在社交网络的实时应用和数据流当中，而现在已经成为下一代数据架构的基础。大型零售商正在基于持续数据流改造他们的基础业务流程，汽车公司正在从互联网汽车那里收集和处理实时数据流，银行也在重新思考基于 Kafka 改造他们的基础流程和系统。

那么 Kafka 在这当中充当了怎样的角色？它与现有的系统有什么区别？

我们认为 Kafka 是一个流平台：在这个平台上可以发布和订阅数据流，并把它们保存起来、进行处理，这就是构建 Kafka 的初衷。以这种方式来看待数据确实与人们习惯的想法有所不同，但它确实在构建应用和架构方面表现出了强大的抽象能力。Kafka 经常会被拿来与现有的技术作比较：企业级消息系统、大数据系统（如 Hadoop）和数据集成或 ETL 工具。这里的每一项比较都有一定的道理，但也有失偏颇。

Kafka 有点像消息系统，允许发布和订阅消息流。从这点来看，它类似于 ActiveMQ、RabbitMQ 或 IBM 的 MQSeries 等产品。尽管看上去有些相似，但 Kafka 与这些传统的消息系统仍然存在很多重要的不同点，这些差异使它完全不同于消息系统。首先，作为一

个现代的分布式系统，Kafka 以集群的方式运行，可以自由伸缩，处理公司的所有应用程序。Kafka 集群并不是一组独立运行的 broker，而是一个可以灵活伸缩的中心平台，可以处理整个公司所有的数据流。其次，Kafka 可以按照你的要求存储数据，保存多久都可以。作为数据连接层，Kafka 提供了数据传递保证——可复制、持久化，保留多长时间完全可以由你来决定。最后，流式处理将数据处理的层次提升到了新高度。消息系统只会传递消息，而 Kafka 的流式处理能力让你只用很少的代码就能够动态地处理派生流和数据集。Kafka 的这些独到之处足以让你刮目相看，它不只是“另一个消息队列”。

从另一个角度来看 Kafka，我们会把它看成实时版的 Hadoop——这也是我们设计和构建 Kafka 的原始动机之一。Hadoop 可以存储和定期处理大量的数据文件，而 Kafka 可以存储和持续处理大型的数据流。从技术角度来看，它们有着惊人的相似之处，很多人将新兴的流式处理看成批处理的超集。它们之间的最大不同体现在持续的低延迟处理和批处理之间的差异上。Hadoop 和大数据主要应用在数据分析上，而 Kafka 因其低延迟的特点更适合用在核心的业务应用上。业务事件时刻在发生，Kafka 能够及时对这些事件作出响应，基于 Kafka 构建的服务直接为业务运营提供支撑，提升用户体验。

Kafka 与 ETL 工具或其他数据集成工具之间也可以进行一番比较。Kafka 和这些工具都擅长移动数据，但我想它们最大的不同在于 Kafka 颠覆了传统的思维。Kafka 并非只是把数据从一个系统拆解出来再塞进另一个系统，它其实是一个面向实时数据流的平台。也就是说，它不仅可以将现有的应用程序和数据系统连接起来，它还能用于加强这些触发相同数据流的应用。我们认为这种以数据流为中心的架构是非常重要的。在某种程度上说，这些数据流是现代数字科技公司的核心，与他们的现金流一样重要。

将上述的三个领域聚合在一起，将所有的数据流整合到一起，流平台因此变得极具吸引力。

当然，除了这些不同点之外，对于那些习惯了开发请求与响应风格应用和关系型数据库的人来说，要学会基于持续数据流构建应用程序也着实是一个巨大的思维转变。借助这本书来学习 Kafka 再好不过了，从内部架构到 API，都是由对 Kafka 最了解的人亲手呈现的。我希望你们能够像我一样喜欢这本书！

—— Jay Kreps, Confluent 联合创始人兼 CEO

---

# 前言

给予一个技术书籍作者最好的赞赏莫过于这句话——“如果在一开始接触这门技术时能看到这本书就好了”。在开始写这本书的时候，我们就是以这句话作为写作目标。我们开发 Kafka，在生产环境运行 Kafka，帮助很多公司构建基于 Kafka 的系统，帮助他们管理数据管道，积累了很多经验，但也困惑：“应该把哪些东西分享给 Kafka 新用户，让他们从新手变成专家？”这本书就是我们日常工作最好的写照：运行 Kafka 并帮助其他人更好地使用 Kafka。

我们相信，书中提供的这些内容能够帮助 Kafka 用户在生产环境运行 Kafka 以及基于 Kafka 构建健壮的高性能应用程序。我们列举了一些非常流行的应用场景：用于事件驱动微服务系统的消息总线、流式应用和大规模数据管道。这本书通俗易懂，能够帮助每一个 Kafka 用户在任意的架构或应用场景里使用好 Kafka。书中介绍了如何安装和配置 Kafka、如何使用 Kafka API、Kafka 的设计原则和可靠性保证，以及 Kafka 的一些架构细节，如复制协议、控制器和存储层。我们相信，Kafka 的设计原理和内部架构不仅会成为分布式系统构建者的兴趣所在，对于那些在生产环境部署 Kafka 或使用 Kafka 构建应用程序的人来说也是非常有用的。越是了解 Kafka，就越是能够更好地作出权衡。

在软件工程里，条条道路通罗马，每一个问题都有多种解决方案。Kafka 为专家级别的用户提供了巨大的灵活性，而新手则需要克服陡峭的学习曲线才能成为专家。Kafka 通常会告诉你如何使用某个功能特性，但不会告诉你为什么要用它或者为什么不该用它。我们会尽可能地解释我们的设计决策和权衡背后的缘由，以及用户在哪些情况下应该或不应该使用 Kafka 提供的特性。

## 读者对象

这本书是为使用 Kafka API 开发应用程序的工程师和在生产环境安装、配置、调优、监控 Kafka 的运维工程师（也可以叫作 SRE、运维人员或系统管理员）而写的。我们也考虑到了数据架构师和数据工程师，他们负责设计和构建整个组织的数据基础架构。某些章节（特别是第 3 章、第 4 章和第 11 章）主要面向 Java 开发人员，并假设读者已经熟悉基本的 Java 语言编程，比如异常处理和并发编程。其他章节（特别是第 2 章、第 8 章、第 9 章和



第 10 章) 则假设读者在 Linux 的运行、存储和网络配置方面有一定的经验。本书的其余部分则讨论了一般性的软件架构, 不要求读者具备特定的知识。

另一类可能对本书感兴趣的人是那些经理或架构师, 他们不直接使用 Kafka, 但会与使用 Kafka 的工程师打交道。他们有必要了解 Kafka 所能提供的保证机制, 以及他们的同事在构建基于 Kafka 的系统时所作出的权衡。这本书可以成为企业管理人员的利器, 确保他们的工程师在 Kafka 方面训练有素, 让他们的团队了解他们本该知道的知识。

## 排版约定

本书使用了下列排版约定。

- **黑体**  
表示新术语或重点强调的内容。
- 等宽字体 (`constant width`)  
表示程序片段, 以及正文中出现的变量、函数名、数据库、数据类型、环境变量、语句和关键字等。
- 加粗等宽字体 (`constant width bold`)  
表示应该由用户输入的命令或其他文本。
- 等宽斜体 (`constant width italic`)  
表示应该由用户输入的值或根据上下文确定的值替换的文本。



该图标表示提示或建议。



该图标表示一般注记。



该图标表示警告或警示。

## 使用代码示例

本书是要帮你完成工作的。一般来说, 如果本书提供了示例代码, 你可以把它用在你的程序或文档中。除非你使用了很大一部分代码, 否则无需联系我们获得许可。比如, 用本书

的几个代码片段写一个程序就无需获得许可，销售或分发 O'Reilly 图书的示例光盘则需要获得许可；引用本书中的示例代码回答问题无需获得许可，将书中大量的代码放到你的产品文档中则需要获得许可。

我们很希望但并不强制要求你在引用本书内容时加上引用说明。引用说明一般包括书名、作者、出版社和 ISBN。例如“Kafka 权威指南，作者 Neha Narkhede、Gwen Shapira 和 Todd Palino (O'Reilly)，版权归 Neha Narkhede、Gwen Shapira 和 Todd Palino 所有，978-1-4919-3616-0”。

如果你觉得自己对示例代码的用法超出了上述许可的范围，欢迎你通过 [permissions@oreilly.com](mailto:permissions@oreilly.com) 与我们联系。

## O'Reilly Safari



Safari (原来叫 Safari Books Online) 是面向企业、政府、教育从业者和个人的会员制培训和参考咨询平台。

我们向会员开放成千上万本图书以及培训视频、学习路线、交互式教程和专业视频。这些资源来自 250 多家出版机构，其中包括 O'Reilly Media、Harvard Business Review、Prentice Hall Professional、Addison-Wesley Professional、Microsoft Press、Sams、Que、Peachpit Press、Adobe、Focal Press、Cisco Press、John Wiley & Sons、Syngress、Morgan Kaufmann、IBM Redbooks、Packt、Adobe Press、FT Press、Apress、Manning、New Riders、McGraw-Hill、Jones & Bartlett 和 Course Technology。

更多信息，请访问 <http://oreilly.com/safari>。

## 联系我们

请把对本书的评价和问题发给出版社。

美国：

O'Reilly Media, Inc.  
1005 Gravenstein Highway North  
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街 2 号成铭大厦 C 座 807 室 (100035)  
奥莱利技术咨询 (北京) 有限公司

O'Reilly 的每一本书都有专属网页，你可以在那儿找到本书的相关信息，包括勘误表、示例代码以及其他信息。本书的网站地址是 <http://oreil.ly/2tVmYjk>。

对于本书的评论和技术性问题，请发送电子邮件到：[bookquestions@oreilly.com](mailto:bookquestions@oreilly.com)

要了解更多 O'Reilly 图书、培训课程、会议和新闻的信息，请访问以下网站：

<http://www.oreilly.com>

我们在 Facebook 的地址如下：<http://facebook.com/oreilly>

请关注我们的 Twitter 动态：<http://twitter.com/oreillymedia>

我们的 YouTube 视频地址如下：<http://www.youtube.com/oreillymedia>

## 致谢

我们想感谢众多为 Kafka 和它的生态系统做出贡献的人。如果没有他们艰辛的工作，就不会有这本书的问世。特别感谢 Jay Kreps、Neha Narkhede 和 Jun Rao，以及他们在 LinkedIn 的同事和领导，他们创造了 Kafka，并把它捐献给了 Apache 软件基金会。

很多人在早前为本书提供了很多有价值的反馈，我们非常感激他们为此付出的时间，也很钦佩他们的专业能力，这些人包括：Apurva Mehta、Arseniy Tashoyan、Dylan Scott、Ewen Cheslack-Postava、Grant Henke、Ismael Juma、James Cheng、Jason Gustafson、Jeff Holoman、Joel Koshy、Jonathan Seidman、Matthias Sax、Michael Noll、Paolo Castagna。我们还想感谢众多在网站上留下评论和反馈的读者。

很多审稿人提供了有价值的意见，极大改进了本书的质量。书中的遗留错误理应由我们作者负责。

我们要感谢 O'Reilly 编辑 Shannon Cutt 的鼓励、耐心和深谋远虑。对于一个作者来说，与 O'Reilly 一起合作是一段非凡的经历——他们所提供的支持，从工具到签名售书，都是无可匹敌的。我们感谢每一个参与本书相关工作的人，很感激他们愿意与我们一起工作。

另外，我们也想感谢我们的领导和同事，感谢他们在我们写作这本书的过程中给予的帮助和鼓励。

Gwen 要感谢她的丈夫 Omer Shapira，在她写书的几个月时间里，他一直给予她支持和耐心。还有她的父亲 Lior Shapira，让她学会了如何在困难面前不轻言放弃，尽管这种生活哲学总是让她麻烦不断。

Todd 要感谢他的妻子 Marcy 和女儿 Bella 及 Kaylee，她们一直在背后默默地支持他。因为有了她们的支持，他才有更多的时间写作，才能厘清思路，坚持到最后。

## 电子书

扫描如下二维码，即可购买本书电子版。



# 目录

序	xiii
前言	xv
第 1 章 初识 Kafka	1
1.1 发布与订阅消息系统	1
1.1.1 如何开始	2
1.1.2 独立的队列系统	3
1.2 Kafka 登场	4
1.2.1 消息和批次	4
1.2.2 模式	4
1.2.3 主题和分区	5
1.2.4 生产者和消费者	5
1.2.5 broker 和集群	6
1.2.6 多集群	7
1.3 为什么选择 Kafka	8
1.3.1 多个生产者	8
1.3.2 多个消费者	8
1.3.3 基于磁盘的数据存储	9
1.3.4 伸缩性	9
1.3.5 高性能	9
1.4 数据生态系统	9
1.5 起源故事	11
1.5.1 LinkedIn 的问题	11
1.5.2 Kafka 的诞生	12

1.5.3 走向开源 .....	12
1.5.4 命名 .....	13
1.6 开始 Kafka 之旅 .....	13
<b>第 2 章 安装 Kafka</b> .....	<b>14</b>
2.1 要事先行 .....	14
2.1.1 选择操作系统 .....	14
2.1.2 安装 Java .....	14
2.1.3 安装 Zookeeper .....	15
2.2 安装 Kafka Broker .....	17
2.3 broker 配置 .....	18
2.3.1 常规配置 .....	18
2.3.2 主题的默认配置 .....	19
2.4 硬件的选择 .....	23
2.4.1 磁盘吞吐量 .....	23
2.4.2 磁盘容量 .....	23
2.4.3 内存 .....	23
2.4.4 网络 .....	24
2.4.5 CPU .....	24
2.5 云端的 Kafka .....	24
2.6 Kafka 集群 .....	24
2.6.1 需要多少个 broker .....	25
2.6.2 broker 配置 .....	25
2.6.3 操作系统调优 .....	26
2.7 生产环境的注意事项 .....	28
2.7.1 垃圾回收器选项 .....	28
2.7.2 数据中心布局 .....	29
2.7.3 共享 Zookeeper .....	29
2.8 总结 .....	30
<b>第 3 章 Kafka 生产者——向 Kafka 写入数据</b> .....	<b>31</b>
3.1 生产者概览 .....	32
3.2 创建 Kafka 生产者 .....	33
3.3 发送消息到 Kafka .....	34
3.3.1 同步发送消息 .....	35
3.3.2 异步发送消息 .....	35
3.4 生产者的配置 .....	36
3.5 序列化器 .....	39
3.5.1 自定义序列化器 .....	39
3.5.2 使用 Avro 序列化 .....	41

3.5.3 在 Kafka 里使用 Avro	42
3.6 分区	45
3.7 旧版的生产者 API	46
3.8 总结	47
<b>第 4 章 Kafka 消费者——从 Kafka 读取数据</b>	<b>48</b>
4.1 KafkaConsumer 概念	48
4.1.1 消费者和消费者群组	48
4.1.2 消费者群组和分区再均衡	51
4.2 创建 Kafka 消费者	52
4.3 订阅主题	53
4.4 轮询	53
4.5 消费者的配置	55
4.6 提交和偏移量	57
4.6.1 自动提交	58
4.6.2 提交当前偏移量	59
4.6.3 异步提交	59
4.6.4 同步和异步组合提交	61
4.6.5 提交特定的偏移量	61
4.7 再均衡监听器	62
4.8 从特定偏移量处开始处理记录	64
4.9 如何退出	66
4.10 反序列化器	67
4.11 独立消费者——为什么以及怎样使用没有群组的消费者	71
4.12 旧版的消费者 API	71
4.13 总结	72
<b>第 5 章 深入 Kafka</b>	<b>73</b>
5.1 集群成员关系	73
5.2 控制器	74
5.3 复制	74
5.4 处理请求	76
5.4.1 生产请求	78
5.4.2 获取请求	78
5.4.3 其他请求	80
5.5 物理存储	81
5.5.1 分区分配	81
5.5.2 文件管理	82
5.5.3 文件格式	83
5.5.4 索引	84

5.5.5	清理	84
5.5.6	清理的工作原理	84
5.5.7	被删除的事件	86
5.5.8	何时会清理主题	86
5.9	总结	86
<b>第 6 章</b>	<b>可靠的数据传递</b>	<b>87</b>
6.1	可靠性保证	87
6.2	复制	88
6.3	broker 配置	89
6.3.1	复制系数	89
6.3.2	不完全的首领选举	90
6.3.3	最少同步副本	91
6.4	在可靠的系统里使用生产者	92
6.4.1	发送确认	92
6.4.2	配置生产者的重试参数	93
6.4.3	额外的错误处理	94
6.5	在可靠的系统里使用消费者	94
6.5.1	消费者的可靠性配置	95
6.5.2	显式提交偏移量	95
6.6	验证系统可靠性	97
6.6.1	配置验证	98
6.6.2	应用程序验证	98
6.6.3	在生产环境监控可靠性	99
6.7	总结	100
<b>第 7 章</b>	<b>构建数据管道</b>	<b>101</b>
7.1	构建数据管道时需要考虑的问题	102
7.1.1	及时性	102
7.1.2	可靠性	102
7.1.3	高吞吐量和动态吞吐量	103
7.1.4	数据格式	103
7.1.5	转换	104
7.1.6	安全性	104
7.1.7	故障处理能力	104
7.1.8	耦合性和灵活性	105
7.2	如何在 Connect API 和客户端 API 之间作出选择	105
7.3	Kafka Connect	106
7.3.1	运行 Connect	106
7.3.2	连接器示例——文件数据源和文件数据池	107

7.3.3	连接器示例——从 MySQL 到 ElasticSearch	109
7.3.4	深入理解 Connect	114
7.4	Connect 之外的选择	116
7.4.1	用于其他数据存储的摄入框架	116
7.4.2	基于图形界面的 ETL 工具	117
7.4.3	流式处理框架	117
7.5	总结	117
<b>第 8 章</b>	<b>跨集群数据镜像</b>	<b>118</b>
8.1	跨集群镜像的使用场景	118
8.2	多集群架构	119
8.2.1	跨数据中心通信的一些现实情况	119
8.2.2	Hub 和 Spoke 架构	120
8.2.3	双活架构	121
8.2.4	主备架构	123
8.2.5	延展集群	127
8.3	Kafka 的 MirrorMaker	128
8.3.1	如何配置	129
8.3.2	在生产环境部署 MirrorMaker	130
8.3.3	MirrorMaker 调优	132
8.4	其他跨集群镜像方案	134
8.4.1	优步的 uReplicator	134
8.4.2	Confluent 的 Replicator	135
8.5	总结	135
<b>第 9 章</b>	<b>管理 Kafka</b>	<b>136</b>
9.1	主题操作	136
9.1.1	创建主题	137
9.1.2	增加分区	138
9.1.3	删除主题	138
9.1.4	列出集群里的所有主题	139
9.1.5	列出主题详细信息	139
9.2	消费者群组	140
9.2.1	列出并描述群组	140
9.2.2	删除群组	142
9.2.3	偏移量管理	142
9.3	动态配置变更	143
9.3.1	覆盖主题的默认配置	143
9.3.2	覆盖客户端的默认配置	145
9.3.3	列出被覆盖的配置	145



9.3.4	移除被覆盖的配置	146
9.4	分区管理	146
9.4.1	首选的首领选举	146
9.4.2	修改分区副本	147
9.4.3	修改复制系数	150
9.4.4	转储日志片段	151
9.4.5	副本验证	152
9.5	消费和生产	153
9.5.1	控制台消费者	153
9.5.2	控制台生产者	155
9.6	客户端 ACL	157
9.7	不安全的操作	157
9.7.1	移动集群控制器	157
9.7.2	取消分区重分配	157
9.7.3	移除待删除的主题	158
9.7.4	手动删除主题	158
9.8	总结	159
<b>第 10 章 监控 Kafka</b>		<b>160</b>
10.1	度量指标基础	160
10.1.1	度量指标在哪里	160
10.1.2	内部或外部度量	161
10.1.3	应用程序健康检测	161
10.1.4	度量指标的覆盖面	161
10.2	broker 的度量指标	162
10.2.1	非同步分区	162
10.2.2	broker 度量指标	166
10.2.3	主题和分区的度量指标	173
10.2.4	Java 虚拟机监控	174
10.2.5	操作系统监控	175
10.2.6	日志	176
10.3	客户端监控	177
10.3.1	生产者度量指标	177
10.3.2	消费者度量指标	179
10.3.3	配额	181
10.4	延时监控	182
10.5	端到端监控	183
10.6	总结	183