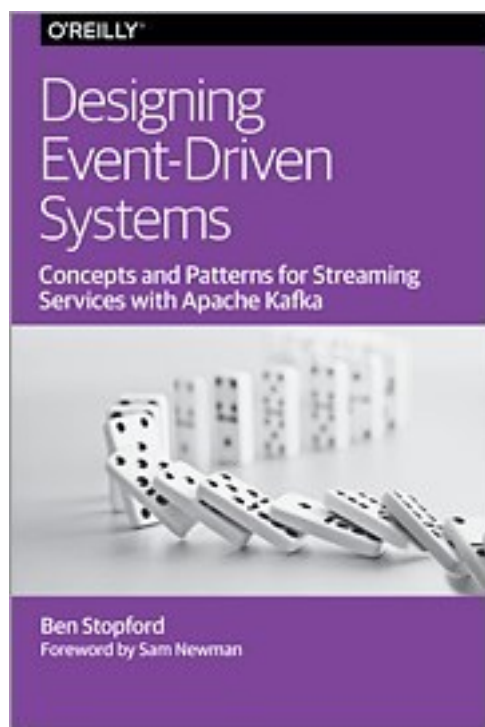


Designing Event-Driven Systems



[Designing Event-Driven Systems 下载链接1](#)

著者:Ben Stopford

出版者:

出版时间:2018-3

装帧:

isbn:9781492038252

作者介绍:

目录: Foreword
Preface
How to Read This Book
Acknowledgments
I. Setting the Stage

1. Introduction
2. The Origins of Streaming
3. Is Kafka What You Think It Is?
Kafka Is Like REST but Asynchronous?
Kafka Is Like a Service Bus?
Kafka Is Like a Database?
What Is Kafka Really? A Streaming Platform
4. Beyond Messaging: An Overview of the Kafka Broker
The Log: An Efficient Structure for Retaining and Distributing Messages
Linear Scalability
Segregating Load in Multiservice Ecosystems
Maintaining Strong Ordering Guarantees
Ensuring Messages Are Durable
Load-Balance Services and Make Them Highly Available
Compacted Topics
Long-Term Data Storage
Security
Summary
- II. Designing Event-Driven Systems
5. Events: A Basis for Collaboration
Commands, Events, and Queries
Coupling and Message Brokers
Is Loose Coupling Always Good?
Essential Data Coupling Is Unavoidable
Using Events for Notification
Using Events to Provide State Transfer
Which Approach to Use
The Event Collaboration Pattern
Relationship with Stream Processing
Mixing Request- and Event-Driven Protocols
Summary
6. Processing Events with Stateful Functions
Making Services Stateful
The Event-Driven Approach
The Pure (Stateless) Streaming Approach
The Stateful Streaming Approach
The Practicalities of Being Stateful
Summary
7. Event Sourcing, CQRS, and Other Stateful Patterns
Event Sourcing, Command Sourcing, and CQRS in a Nutshell
Version Control for Your Data
Making Events the Source of Truth
Command Query Responsibility Segregation
Materialized Views
Polyglot Views
Whole Fact or Delta?
Implementing Event Sourcing and CQRS with Kafka
Build In-Process Views with Tables and State Stores in Kafka Streams
Writing Through a Database into a Kafka Topic with Kafka Connect
Writing Through a State Store to a Kafka Topic in Kafka Streams
Unlocking Legacy Systems with CDC
Query a Read-Optimized View Created in a Database
Memory Images/Prepopulated Caches
The Event-Sourced View

Summary

III. Rethinking Architecture at Company Scales

8. Sharing Data and Services Across an Organization

Encapsulation Isn't Always Your Friend

The Data Dichotomy

What Happens to Systems as They Evolve?

The God Service Problem

The REST-to-ETL Problem

Make Data on the Outside a First-Class Citizen

Don't Be Afraid to Evolve

Summary

9. Event Streams as a Shared Source of Truth

A Database Inside Out

Summary

10. Lean Data

If Messaging Remembers, Databases Don't Have To

Take Only the Data You Need, Nothing More

Rebuilding Event-Sourced Views

Kafka Streams

Databases and Caches

Handling the Impracticalities of Data Movement

Automation and Schema Migration

The Data Divergence Problem

Summary

IV. Consistency, Concurrency, and Evolution

11. Consistency and Concurrency in Event-Driven Systems

Eventual Consistency

Timeliness

Collisions and Merging

The Single Writer Principle

Command Topic

Single Writer Per Transition

Atomicity with Transactions

Identity and Concurrency Control

Limitations

Summary

12. Transactions, but Not as We Know Them

The Duplicates Problem

Using the Transactions API to Remove Duplicates

Exactly Once Is Both Idempotence and Atomic Commit

How Kafka's Transactions Work Under the Covers

Store State and Send Events Atomically

Do We Need Transactions? Can We Do All This with Idempotence?

What Can't Transactions Do?

Making Use of Transactions in Your Services

Summary

13. Evolving Schemas and Data over Time

Using Schemas to Manage the Evolution of Data in Time

Handling Schema Change and Breaking Backward Compatibility

Collaborating over Schema Change

Handling Unreadable Messages

Deleting Data

Triggering Downstream Deletes

Segregating Public and Private Topics

Summary

V. Implementing Streaming Services with Kafka

14. Kafka Streams and KSQL

A Simple Email Service Built with Kafka Streams and KSQL

Windows, Joins, Tables, and State Stores

Summary

15. Building Streaming Services

An Order Validation Ecosystem

Join-Filter-Process

Event-Sourced Views in Kafka Streams

Collapsing CQRS with a Blocking Read

Scaling Concurrent Operations in Streaming Systems

Rekey to Join

Repartitioning and Staged Execution

Waiting for N Events

Reflecting on the Design

A More Holistic Streaming Ecosystem

Summary

• • • • • ([收起](#))

[Designing Event-Driven Systems 下载链接1](#)

标签

编程

计算机

Kafka

英文版

电纸书

评论

好书，对过去event-driven领域所有探索对一次总结，围绕kafka讨论了事件驱动模型中会遇到的问题

好久不看技术书了，感觉不错，以后吹牛底气又足了一点点

被迫四天读完了这本书，完全是填鸭式学习。书中较靠前的经验之谈确实在先前项目的实际运用中遇到了，后面还有需要咀嚼的内容也不少。应该会给系统设计带来不少帮助。
。

[Designing Event-Driven Systems 下载链接1](#)

书评

[Designing Event-Driven Systems 下载链接1](#)