



Kafka Deep Dive

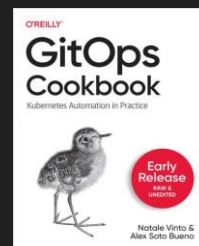
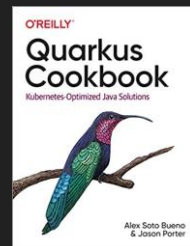
Homework dn.dev/kafka-tutorial



Alex Soto (lordofthejars.com)



- @alexsotob
- asotobue@redhat.com
- Currently Red Hat's Director of Developer Experience
- Featured speaker at technology events around the globe
- A Java Champion since 2017
- Writer, University Professor, Radio collaborator
- A big fan of testing and continuous delivery in 21st century



Agenda



Red Hat
Developer



- What is Kafka
- Why Kafka
- Topics and Partitions
- Consumers and Producers
- Kafka in Kubernetes

What & Why Kafka?



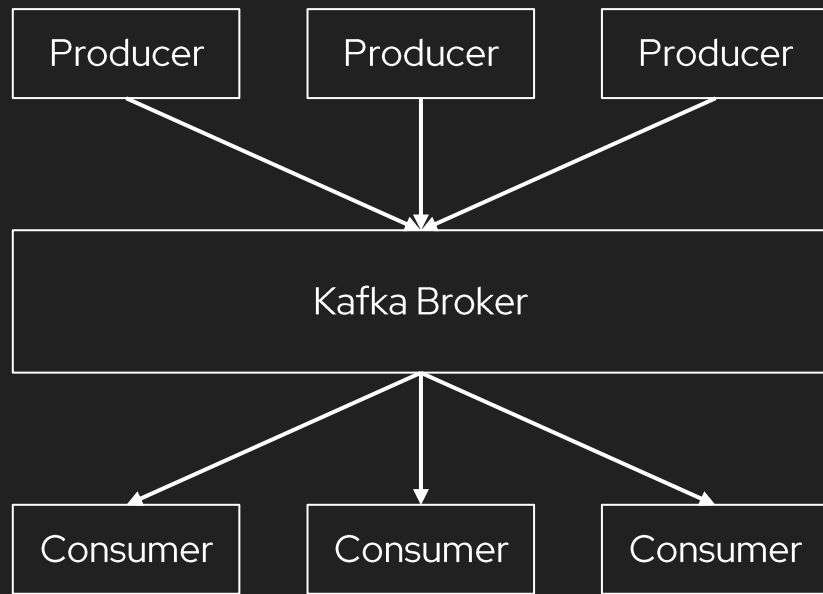
Why did they name it Kafka?

"A system optimized for writing"

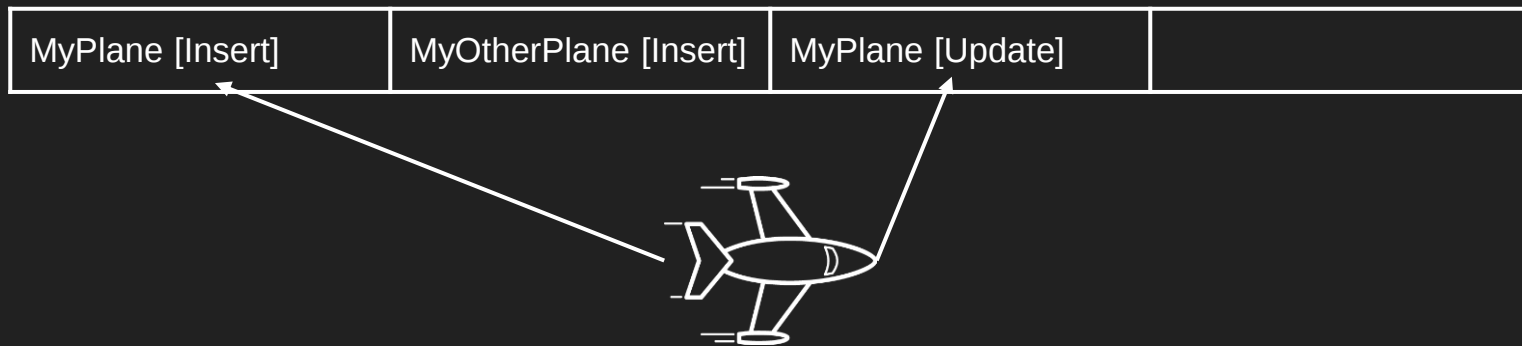
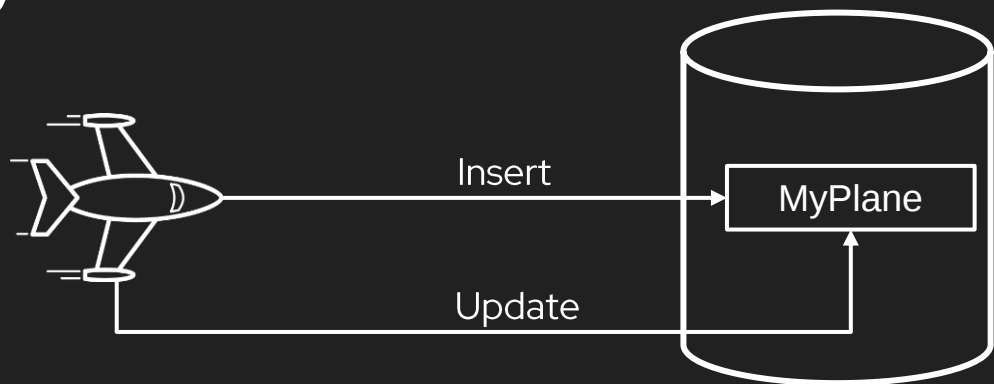


What is Kafka?

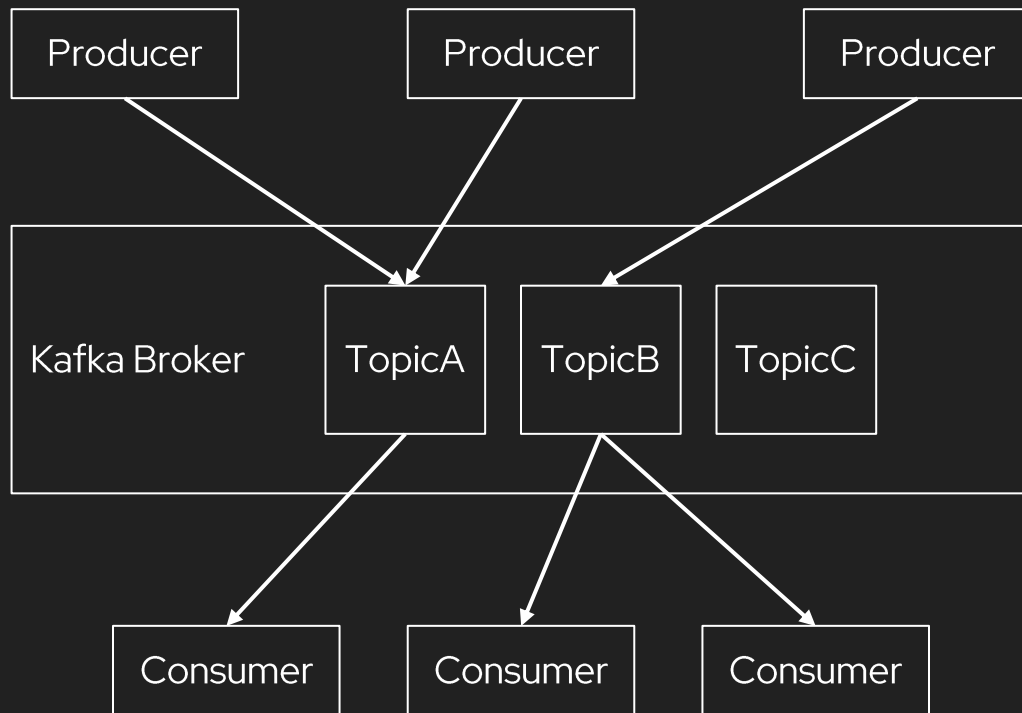
- Publish/Subscribe System
- Fault Tolerance/Scalability by Construction
- Processing Streams in Real Time

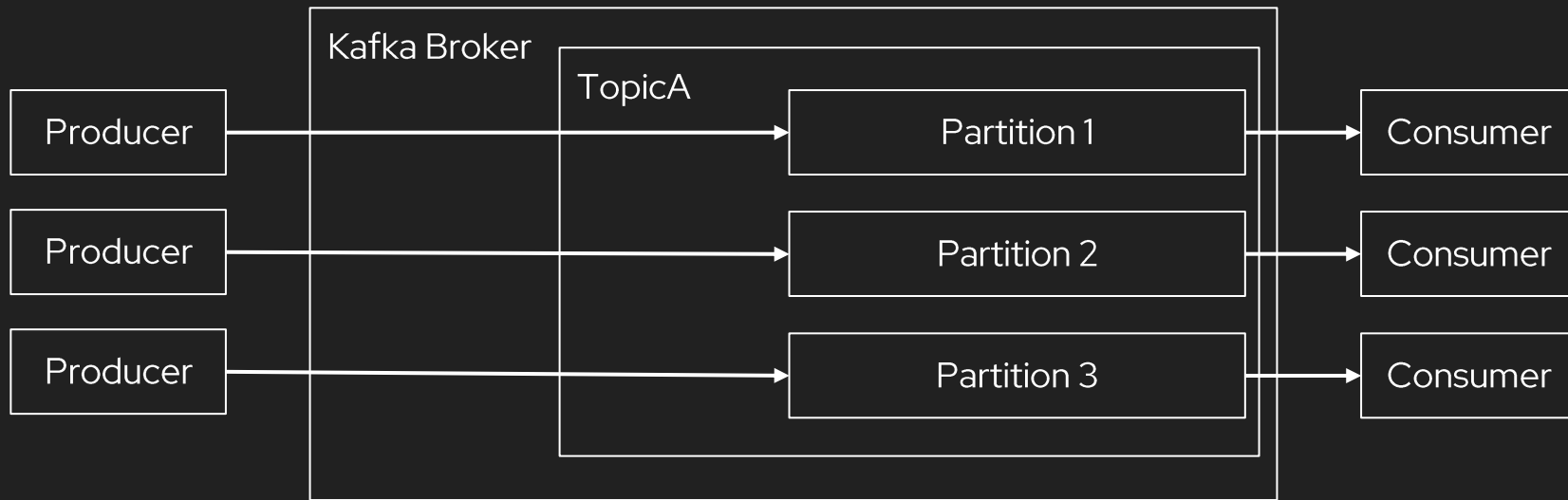


Why Kafka?



Topics & Partitions





Kafka Broker

TopicA

Partition 1

Partition 2

Partition 3

Kafka Broker

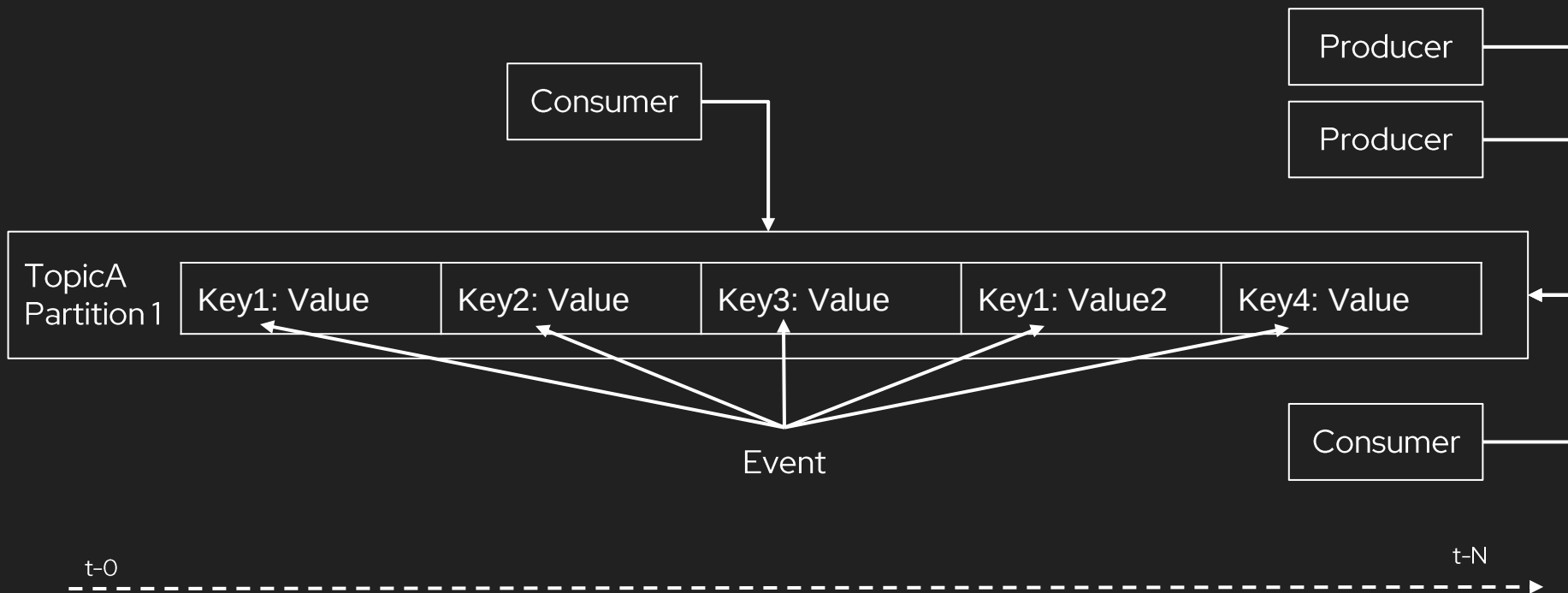
TopicA

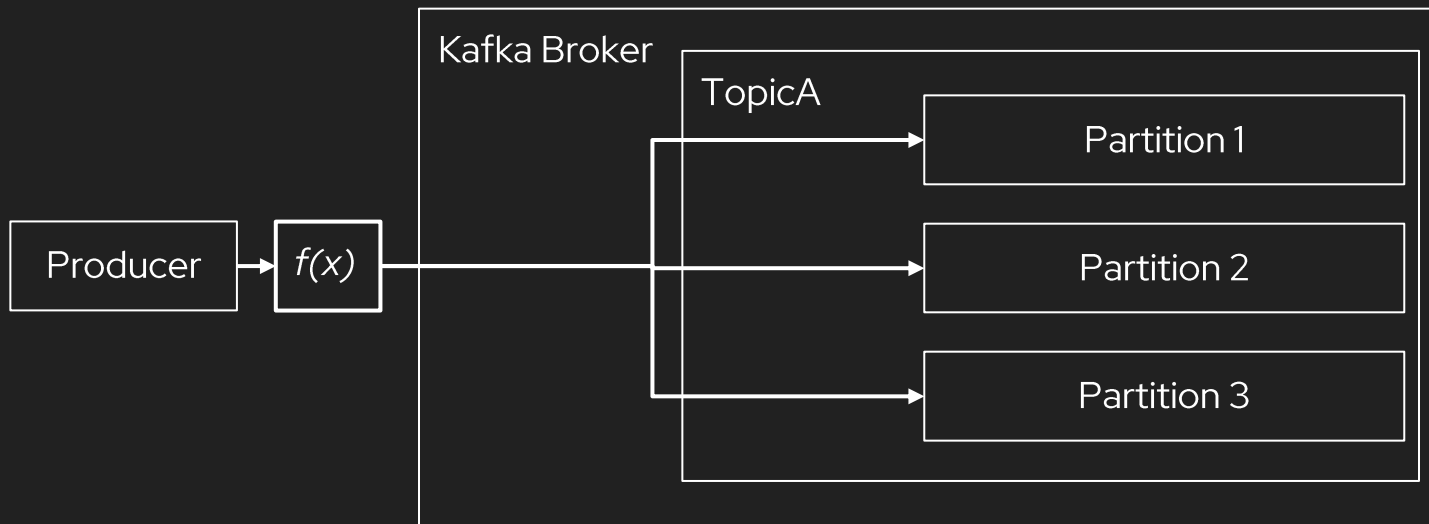
Partition 1

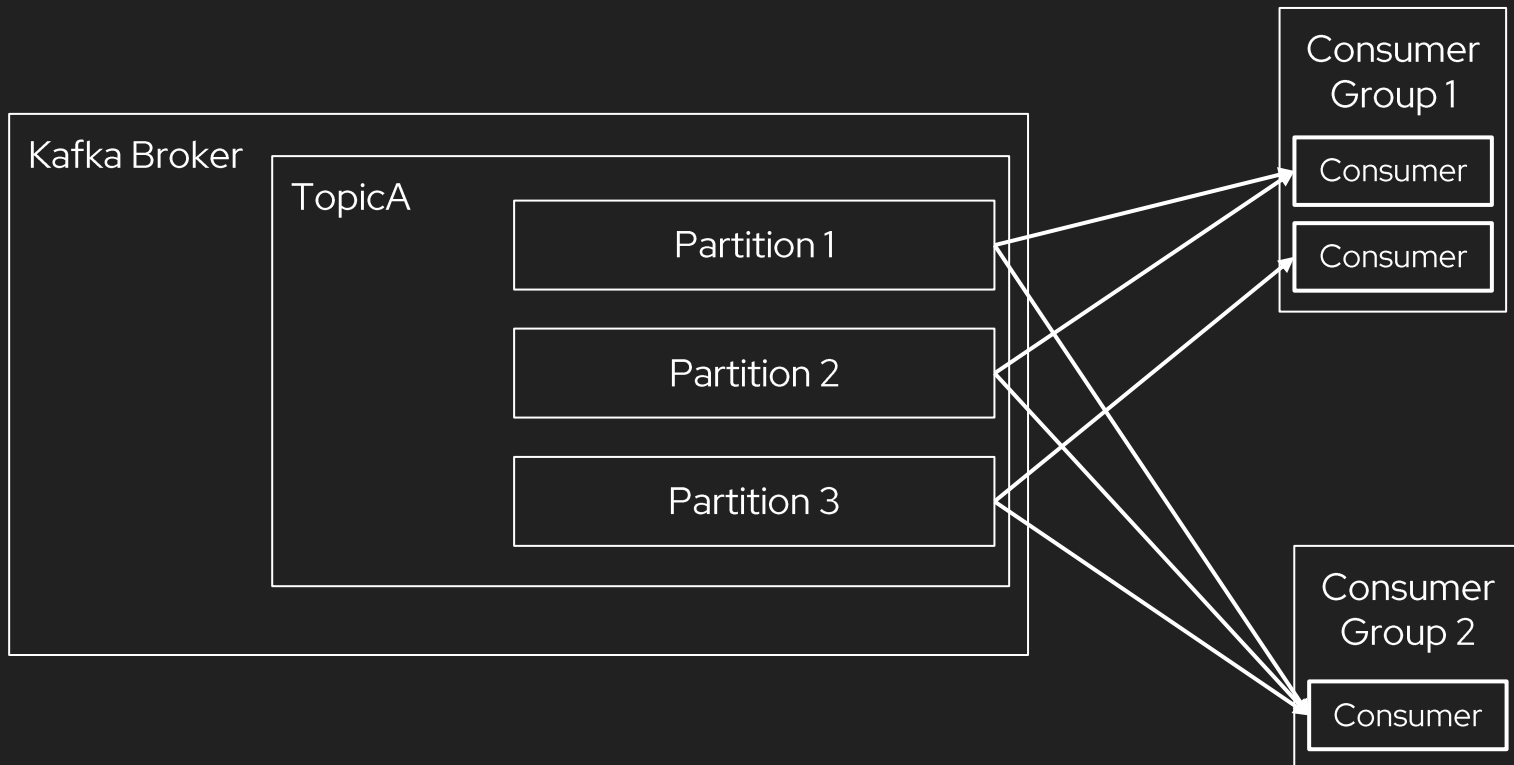
Partition 2

Partition 3

Consumers & Producers







Kafka & Kubernetes

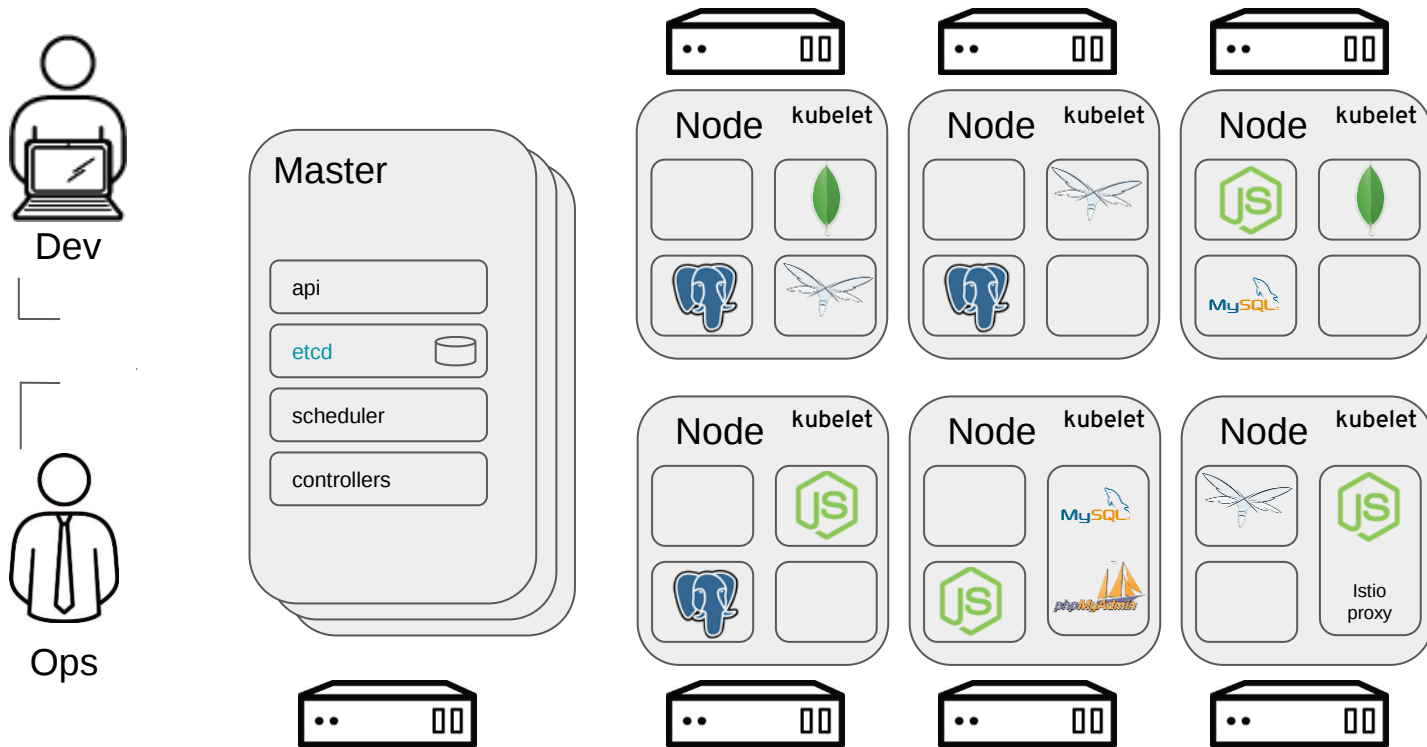
Meet Kubernetes

- Greek for “Helmsman,” also the root of the word “Governor” (from latin: gubernator)
- Container orchestrator
- Supports multiple cloud and bare-metal environments
- Inspired by Google’s experience with containers
- Open source, written in Go
- Manage applications, not machines



OPENSIFT

Kubernetes Cluster - Nodes





STRIMZI

<http://strimzi.io/>

Apache Kafka on Kubernetes & OpenShift

Secure By Default

- Out of the box security
- TLS & SCRAM-SHA supported
- Automated Certificate Management

Configurable

- NodePort, Load Balancer, Ingress options.
- Rack awareness for HA
- Use dedicated nodes for Kafka

Kubernetes-Native

- `kubectl get kafka`
- Operator Based
- Manage Kafka using GitOps



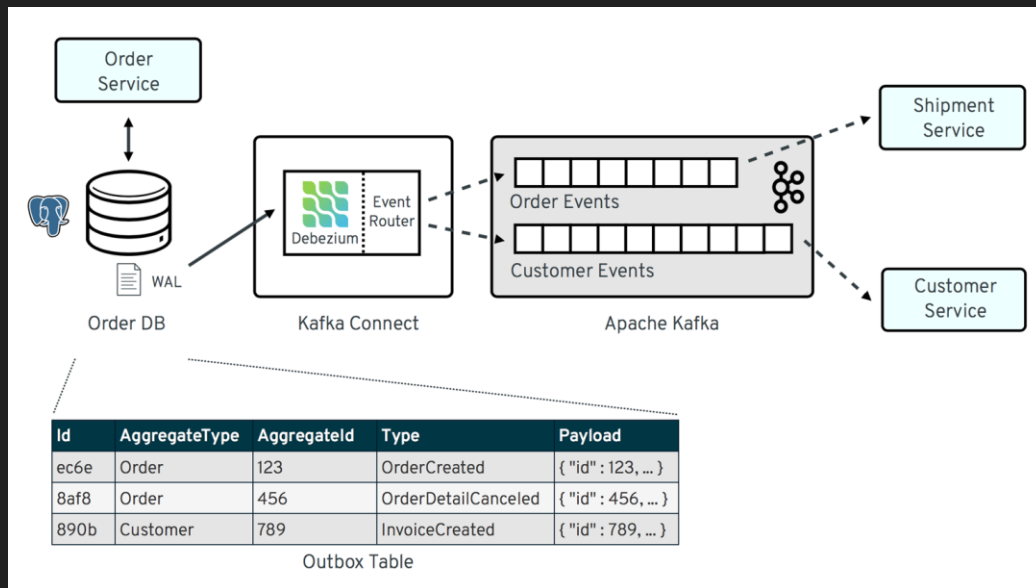
Example of Deployment

```
apiVersion: kafka.strimzi.io/v1beta1
kind: Kafka
metadata:
  name: my-cluster
spec:
  kafka:
    version: 2.4.0
    replicas: 3
    listeners:
      plain: {}
      tls: {}
    config:
      offsets.topic.replication.factor: 3
      transaction.state.log.replication.factor: 3
      transaction.state.log.min.isr: 2
      log.message.format.version: "2.4"
    storage:
      type: ephemeral
  zookeeper:
    replicas: 3
    storage:
      type: ephemeral
  entityOperator:
    topicOperator: {}
    userOperator: {}
```



debezium

<https://www.youtube.com/watch?v=QYbXDp4Vu-8>

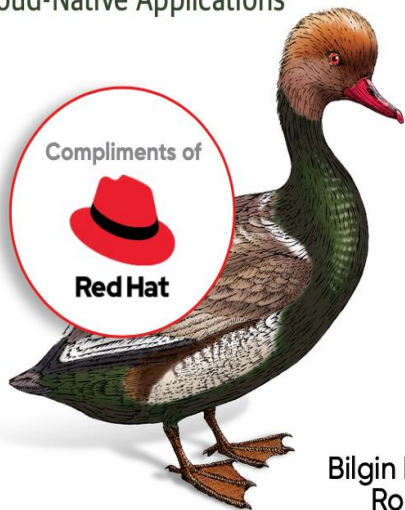


Additional Resources

O'REILLY®

Kubernetes Patterns

Reusable Elements for Designing
Cloud-Native Applications



Bilgin Ibryam &
Roland Huß



Red Hat
Developer

DevNation

[Download](#)

dn.dev/k8spatterns1

dn.dev/kafkamaster

O'REILLY®

Knative Cookbook

Building Effective Serverless Applications
with Kubernetes and OpenShift



Compliments of



Red Hat

Burr Sutter &
Kamesh Sampath



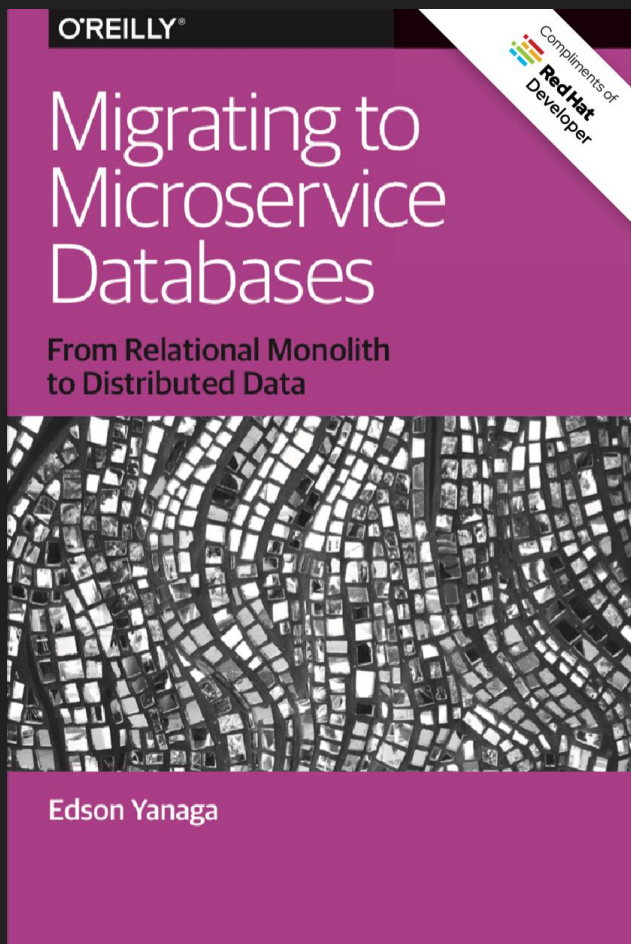
Red Hat
Developer

 **DevNation**

[Download](#)

dn.dev/knative-cookbook

dn.dev/kafkamaster



[Download](#)

bit.ly/mono2microdb



Get started with Red Hat OpenShift Streams for Apache Kafka

Check out this quick two-minute video on how to use OpenShift Streams for Apache Kafka, a fully hosted and managed Kafka service for stream-based applications that delivers streamlined operations and improved application velocity.

[Create a Kafka instance](#)[Connect to a Kafka cluster in the Developer Sandbox →](#)

<https://red.ht/TryKafka>
a

The End