Simplifying **Apache Kafka** on **Kubernetes** with **Strimzi**

Paolo Patierno
Gantigmaa Selenge

Agenda

- Introduction to Kafka and Strimzi
- New features
  - KRaft
  - Tiered Storage
  - Auto Rebalancing
- Exciting upcoming features

Apache Kafka

- Leading distributed event streaming platform
  - Scales horizontally
  - Highly available and fault tolerant
  - Wide variety of use cases
- Open source project
  - Part of the Apache Software Foundation
  - Licensed under Apache License 2.0

CLOUD NATIVE
COMPUTING FOUNDATION

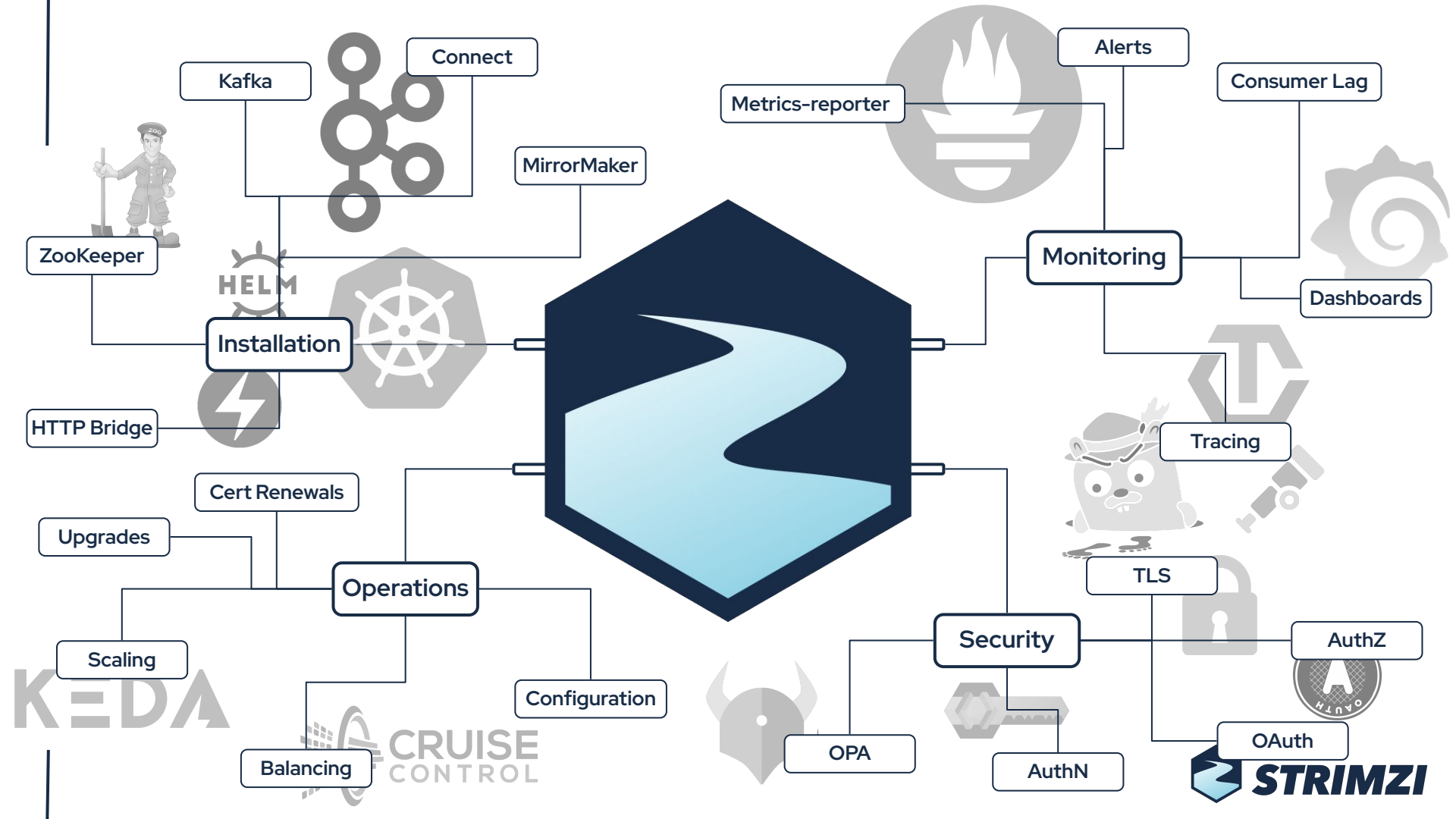
What is Strimzi

- Open Source project / community (Apache License 2.0)
- CNCF Incubating project
- Manages on Apache Kafka on Kubernetes
  - Based on the operator pattern
  - Provides operators for running and managing Apache Kafka and its components
  - Reduces operational overhead

STRIMZI

Kubernetes-native Kafka

- Adopts the operator pattern
  - Uses CRDs to extend Kubernetes API and define Kafka resources
  - Integrates Kafka knowledge into the operator
- Manages Kafka resources through operator pattern
  - Kafka topics, users and connector


Kafka
Connect
MirrorMaker
ZooKeeper
HELM
Installation
HTTP Bridge
Metrics-reporter
Alerts
Consumer Lag
Monitoring
Dashboards
Tracing
TLS
Cert Renewals
Upgrades
Operations
Scaling
KEDA
Balancing
CRUISE CONTROL
Configuration
Security
AuthZ
OAUTH
OAuth
OPA
AuthN
STRIMZI

What's new?

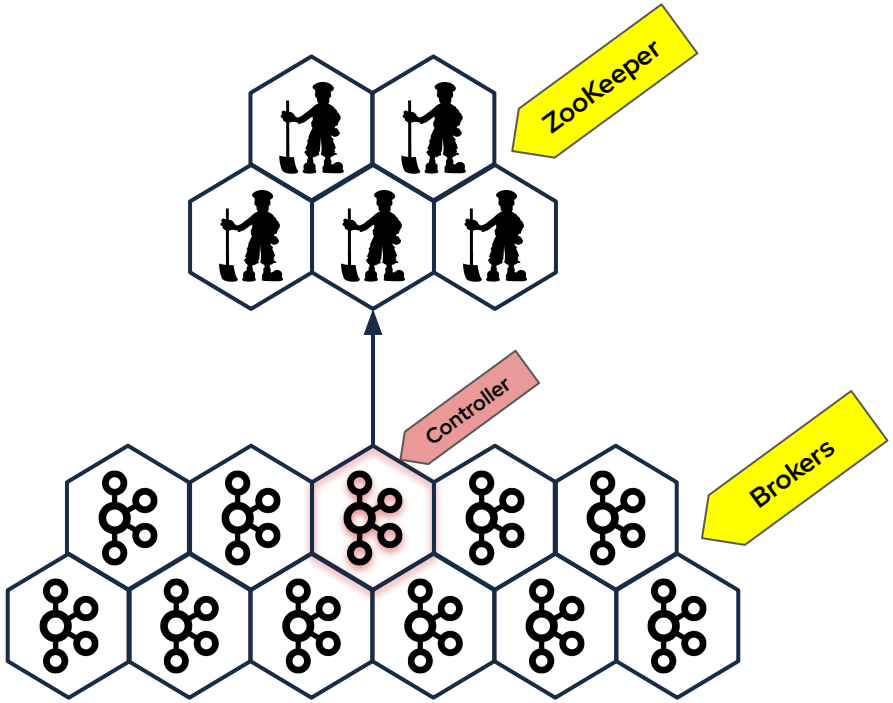
STRIMZI

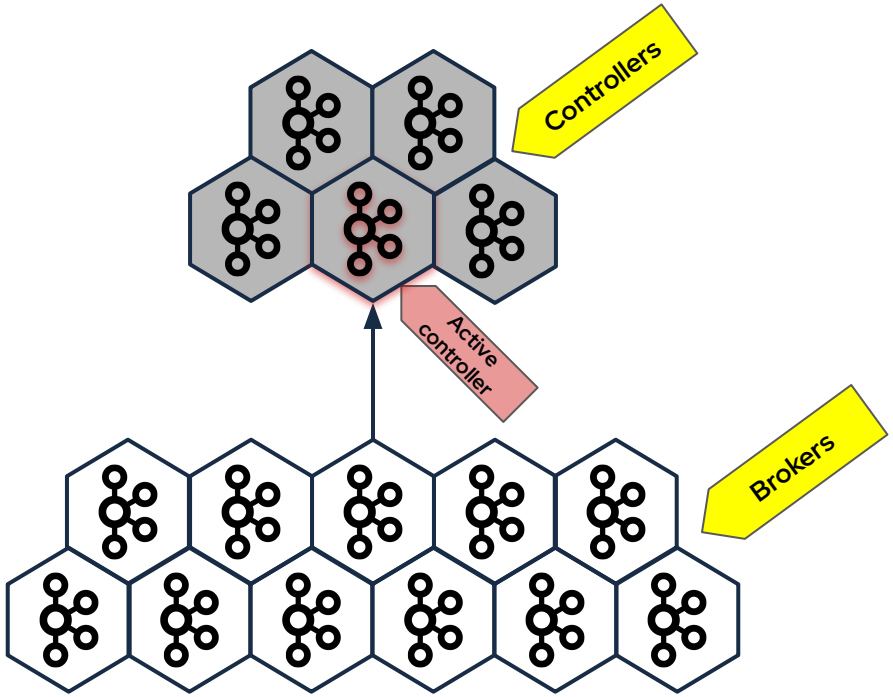
KRaft

STRIMZI

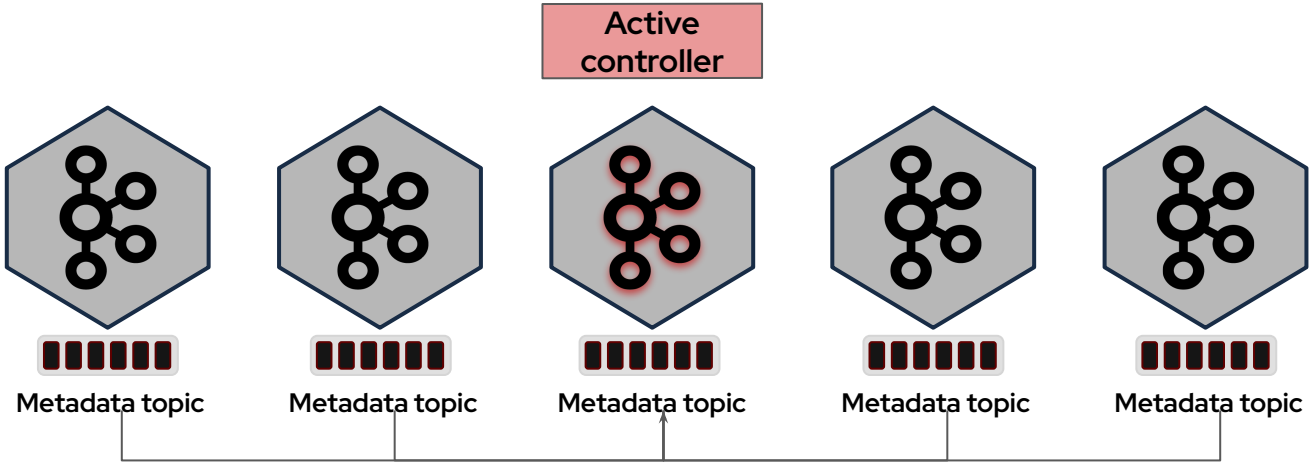
KRaft

- Removes Zookeeper dependency for metadata management
  - Replaces by Kafka's own implementation based on Raft protocol
  - Simplifies deployment and management of Kafka clusters
  - Improves scalability, efficiency and performance


ZooKeeper
Controller
Brokers

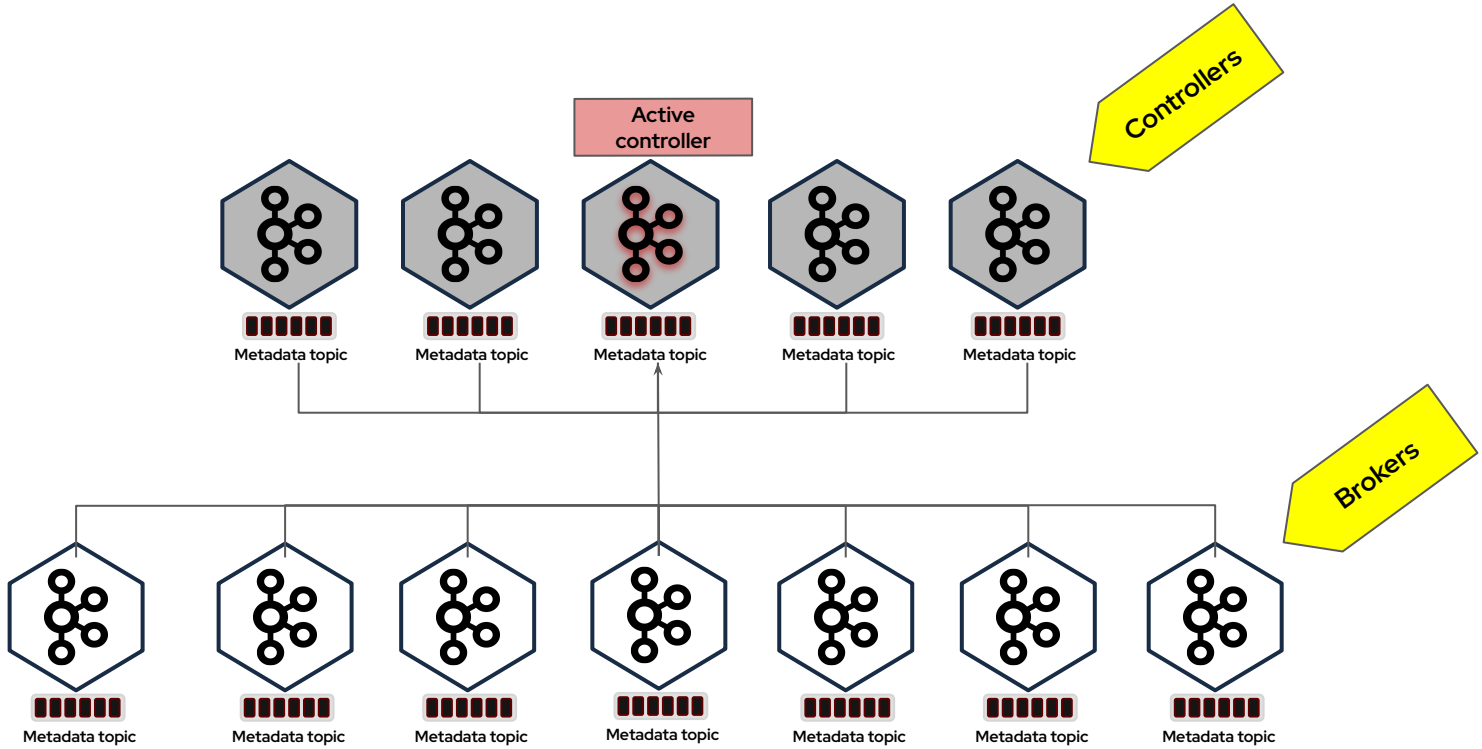


STRIMZI

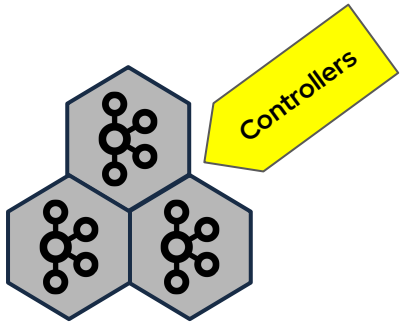


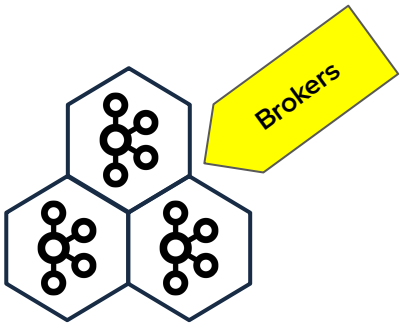
`__cluster_metadata`

STRIMZI

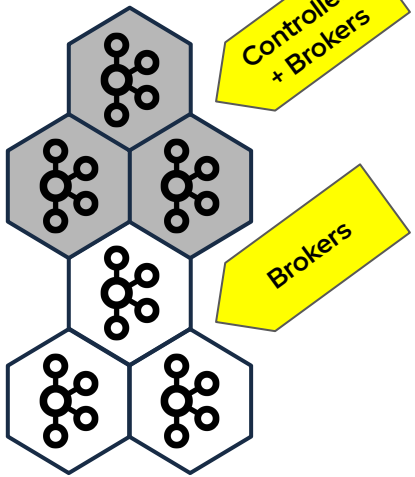

Controllers
Active controller
Metadata topic
Metadata topic
Metadata topic
Metadata topic
Metadata topic
Brokers
Metadata topic
Metadata topic
Metadata topic
Metadata topic
Metadata topic
Metadata topic
Metadata topic

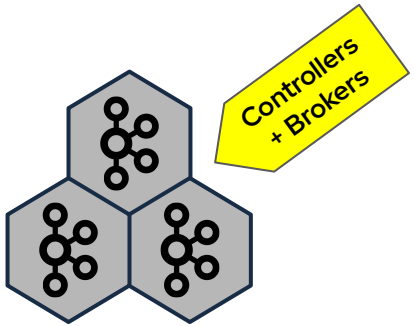
STRIMZI

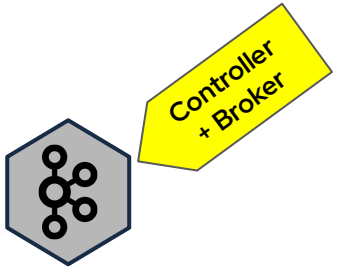




Recommended production setup



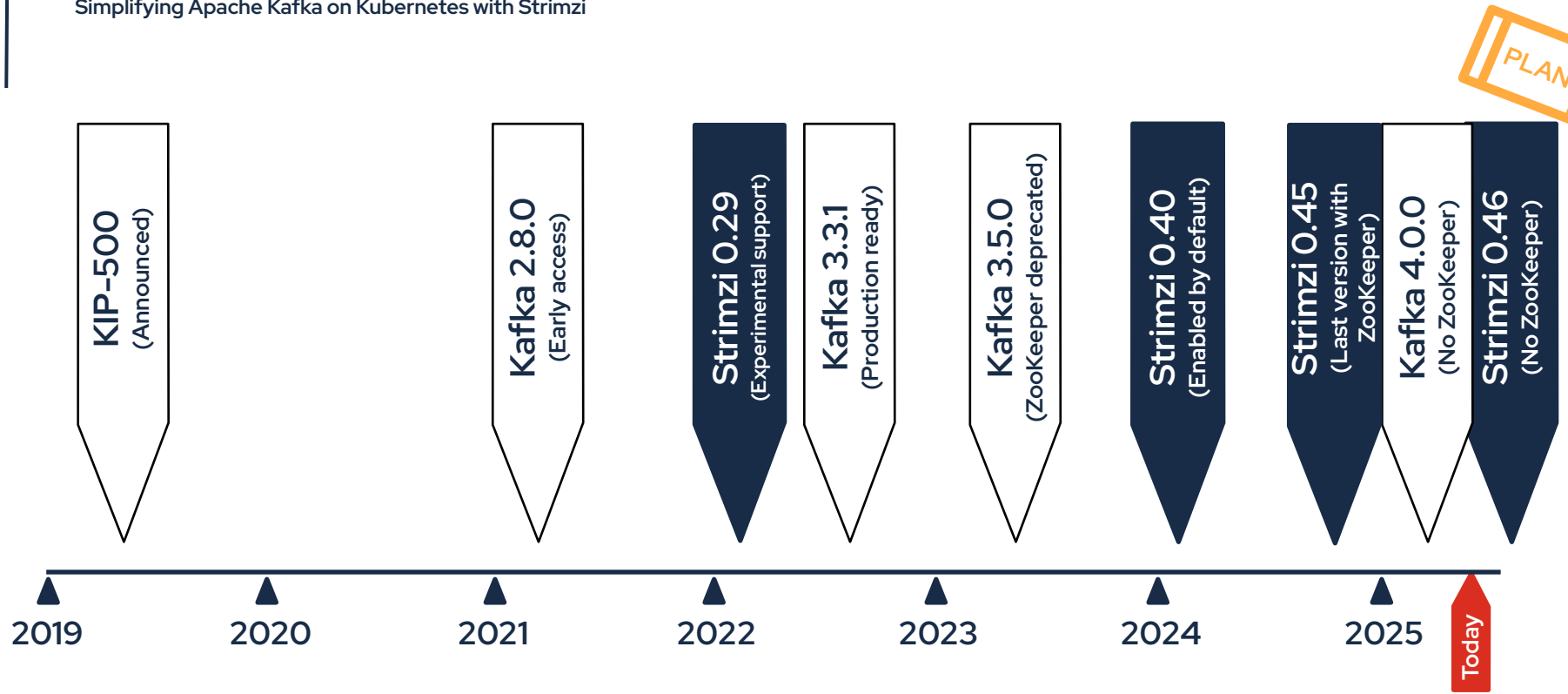




STRIMZI

Migration

- Existing ZooKeeper based clusters can be migrated to KRaft
- KRaft and migration are supported in Strimzi => you can use it today!
  - Migration to KRaft is driven by users through annotations in semi-automated way
  - Cannot be fully automated due to difference in architecture and configurations
  - Provides rollback option up to a certain point


PLAN
KIP-500
(Announced)
Kafka 2.8.0
(Early access)
Strimzi 0.29
(Experimental support)
Kafka 3.3.1
(Production ready)
Kafka 3.5.0
(ZooKeeper deprecated)
Strimzi 0.40
(Enabled by default)
Strimzi 0.45
(Last version with ZooKeeper)
Kafka 4.0.0
(No ZooKeeper)
Strimzi 0.46
(No ZooKeeper)
2019
2020
2021
2022
2023
2024
2025
Today

STRIMZI

Timeline

- Strimzi 0.45.0 (the current release)
  - Supports Apache Kafka 3.8.0 and 3.9.0
  - **Last version with ZooKeeper support**
  - We plan to provide “extended support”
  - **ZooKeeper-based Kafka clusters need to be migrated to KRaft**

Timeline

- Strimzi 0.46.0 (next release)
  - Supports Apache Kafka 3.9.0 and 4.0.0
  - **Only KRaft-based Kafka clusters supported**
  - **ZooKeeper-based Kafka clusters need to be migrated to KRaft before upgrading to Strimzi 0.46.0**
  - Mirror Maker 1 support will be removed as well

Learn more about KRaft

https://strimzi.io/kraft/

STRIMZI

Tiered Storage

STRIMZI

Why do you want to use it?

- Cost efficiency
- Scalability
- Faster recovery and rebalancing
- Simplified cluster operations

STRIMZI

The “usual” Kube-native approach

- New **`kafka.tieredStorage`** properties in the **Kafka** resource

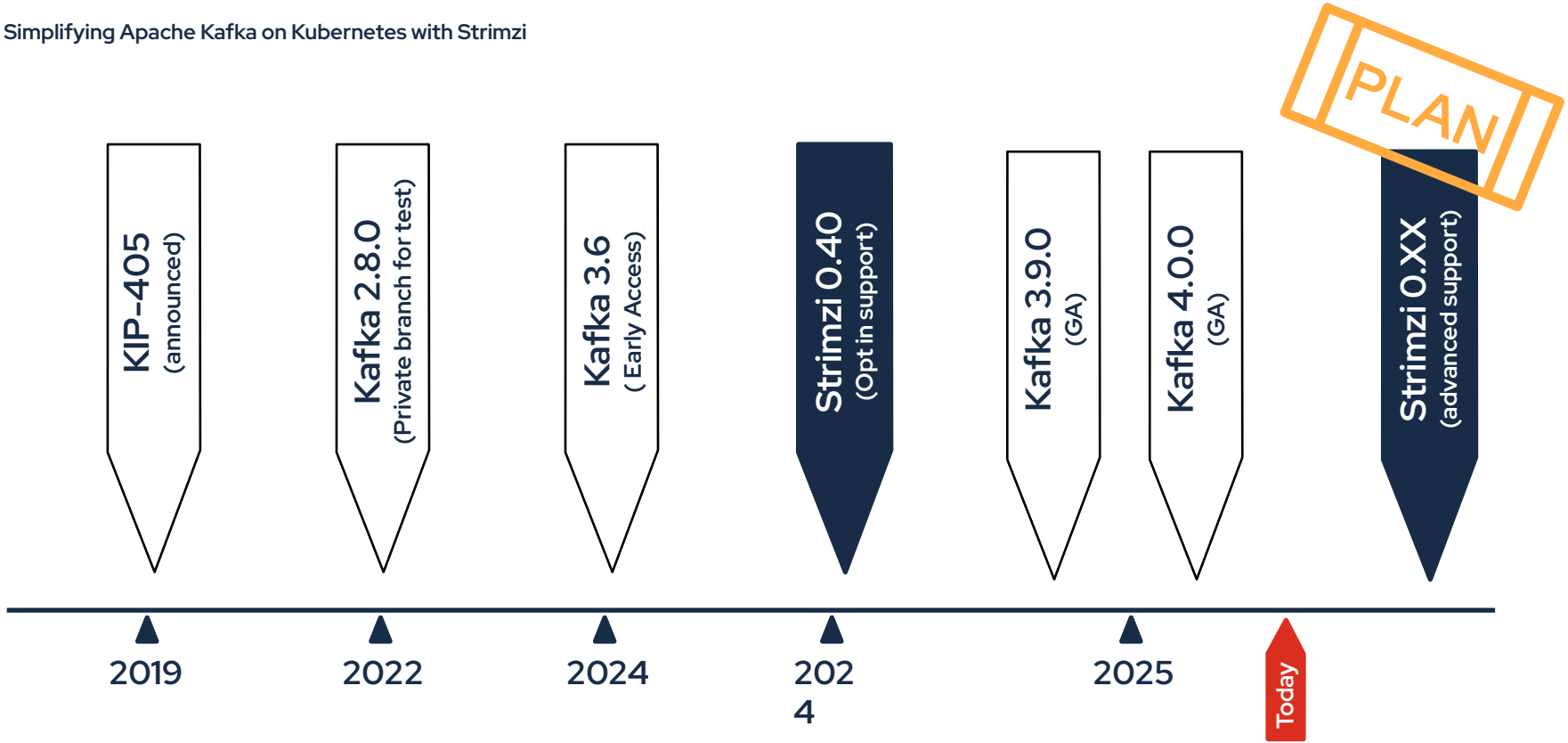
```
kafka:
  tieredStorage:
    type: custom
    remoteStorageManager:
      className: com.example.kafka.tiered.storage.s3.S3RemoteStorageManager
      classPath: /opt/kafka/plugins/tiered-storage-s3/*
      config:
        # A map with String keys and String values. Key fields will be automatically
        # prefixed with `rsm.config.` and appended to Kafka broker config.
        storage.bucket.name: my-bucket
  config:
    ...
    rlmm.config.remote.log.metadata.topic.replication.factor: 1
```


STRIMZI

More insights ...

- Support for tiered storage with custom plugins
  - Allows to configure tiered storage in Kafka brokers
- Plugins options
  - Aiven's open source plugins:

    https://github.com/Aiven-Open/tiered-storage-for-apache-kafka
    - Amazon S3, Google Cloud Storage, Azure Blob Storage and file storage
  - Custom implementation


PLAN
KIP-405
(announced)
Kafka 2.8.0
(Private branch for test)
Kafka 3.6
(Early Access)
Strimzi 0.40
(Opt in support)
Kafka 3.9.0
(GA)
Kafka 4.0.0
(GA)
Strimzi 0.XX
(advanced support)
2019
2022
2024
202
4
2025
Today

STRIMZI

Auto-rebalancing on cluster scaling

STRIMZI

Auto-rebalancing

- Automatically triggers a rebalance when brokers are added or removed
  - Makes it easier to scale the Kafka cluster up or down
  - Moves partition replicas to new brokers after they were added
  - Moves partition replicas from brokers before they are removed
- Rebalancing via a `KafkaRebalance` template

What's next?

STRIMZI

What's next?

- Improved certificate management
- Kafka cluster self-healing with Cruise Control
- V1 APIs and Strimzi 1.0.0
- Gateway API support
- Stretch clusters

STRIMZI

Improved certificate management

- Better integration with externals providers of TLS certificates
  - cert-manager to be the first supported tool
  - pluggable architecture to have more providers in the future
  - easy integration within the users' process for certificates management

Kafka cluster self-healing

- Strimzi leverages Cruise Control for cluster rebalancing
  - Manual rebalancing involves the usage of a **`KafkaRebalance`** custom resource
  - Auto-rebalancing helps only on scaling the cluster up or down
- Plan to integrate the Cruise Control self-healing feature
  - Anomaly detectors to detect broker or disk failures, goal violation, topic anomalies and more
  - Anomalies being notified and fixed where possible
- https://github.com/strimzi/proposals/pull/145

v1 APIs and Strimzi 1.0.0

- The Strimzi community decided to wait for ZooKeeper removal long time ago
  - The time is finally here!
  - After the next release without ZooKeeper we'll start working on v1 API (CRDs)
- v1 and v1beta2 will be supported together for some time
- Strimzi 1.0.0 will come later by supporting v1 only

Gateway API support

- Gateway API aims at replacing Ingress
  - Should provide better compatibility between providers
- Can be already used to expose a Strimzi-based Kafka cluster
  - Requires manual configuration
- The plan is about having a built-in support within Strimzi

Stretch clusters

- Having Kafka cluster nodes running on different Kubernetes clusters
- Kafka is very sensitive to latency so it could not be always possible
  - Suitable for Metropolitan Area Networks / Metro DR
- Simplyfing migration between clusters
- https://github.com/strimzi/proposals/pull/129

STRIMZI

STRIMZI

Join us

https://strimzi.io/join-us/

STRIMZI

Thank you

 Website: https://strimzi.io

 GitHub: https://github.com/strimzi

 Twitter: @strimziio

 YouTube: https://youtube.com/c/Strimzi

 LinkedIn: https://www.linkedin.com/company/strimzi

STRIMZI