APACHE KAFKA, IIOT, INDUSTRIAL IOT, INDUSTRY 4.0, INTERNET OF THINGS, KAFKA CONNECT, KAFKA STREAMS, KSQL, MQTT, SPARKPLUG

Apache Kafka and MQTT (Part 1 of 5) – Overview and Comparison

🕐 4 minute read

Apache Kafka and MQTT are a perfect combination for many IoT use cases. This blog series covers various use cases across industries including connected vehicles, manufacturing, mobility services, and smart city. This is part 1: Overview + Comparison.

By 👤 KAI WAEHNER · 15. March 2021   🅕 25   🐦   📌

Apache Kafka and MQTT are a perfect combination for many IoT use cases. This **blog series** covers the pros and cons of both technologies. Various use cases across industries, including connected vehicles, manufacturing, mobility services, and smart city are explored. The examples use different architectures, including lightweight edge scenarios, hybrid integrations, and serverless cloud solutions. This post is **part one: Overview and Comparison**.

Gartner predicts: *"Around 10% of **enterprise-generated data is created and processed outside a traditional centralized data center or cloud**. By 2025, this figure **will reach 75%**"*. Hence, looking at the combination of MQTT and Kafka makes a lot of sense!

Apache Kafka + MQTT Blog Series

The first blog post explores the relation between MQTT and Apache Kafka. Afterward, the other four blog posts discuss various use cases, architectures, and reference deployments.

- **Part 1 – Overview (THIS POST)**: Relation between Kafka and MQTT, pros and cons, architectures
- **Part 2 – Connected Vehicles**: MQTT and Kafka in a private cloud on Kubernetes; use case: remote control and command of a car
- **Part 3 – Manufacturing**: MQTT and Kafka at the edge in a smart factory; use case: Bidirectional OT-IT integration with Sparkplug B between PLCs, IoT Gateways, Data Historian, MES, ERP, Data Lake, etc.
- **Part 4 – Mobility Services**: MQTT and Kafka leveraging serverless cloud infrastructure; use case: Traffic jam prediction service using machine learning
- **Part 5 – Smart City**: MQTT at the edge connected to fully-managed Kafka in the public cloud; use case: Intelligent traffic routing by combining and correlating different 1st and 3rd party services

Subscribe to my newsletter to get updates immediately after the publication. Besides, I will also update the above list with direct links to this blog series's

posts as soon as published.

Apache Kafka vs. MQTT

MQTT is an open standard for a publish/subscribe messaging protocol. Open source and commercial solutions provide implementations of different MQTT standard version. MQTT was **built for IoT use cases**, including constrained devices and unreliable networks. However, it was not built for data integration and data processing.

Contrary to the above, **Apache Kafka** is not an IoT platform. Instead, Kafka is an **event streaming platform** and used the underpinning of an event-driven architecture for various use cases across industries. It provides a **scalable, reliable, and elastic real-time platform for messaging, storage, data integration, and stream processing**.

To clarify, **MQTT and Kafka complement each other**. Both have their strength and weaknesses. They typically do not compete with each other.

When (not) to use MQTT?

This section explores the trade-offs of both technologies.

Pros of MQTT

- Lightweight
- Built for poor connectivity / high latency scenarios (e.g., mobile networks!)
- High scalability and availability (with the right MQTT broker)
- ISO Standard
- Most popular IoT protocol
- Deployable on all infrastructures (edge, data center, public cloud)

Cons of MQTT

- Only pub/sub messaging, no stream processing, no data integration
- Asynchronous processing (clients can be offline for a long time)
- No reprocessing of events

When (not) to use Apache Kafka?

Pros of Kafka

- Processing data in motion – not just pub/sub messaging, but also including stream processing and data integration
- High throughput
- Large scale
- High availability
- Long term storage and buffering for real decoupling of producers and consumers
- Reprocessing of events
- Good integration to the rest of the enterprise
- Deployable on all infrastructures (edge, data center, public cloud)

Cons of Kafka

- Not built for tens of thousands of connections
- Requires stable network and good infrastructure
- No IoT-specific features like keep alive, last will, or testament

TL;DR

Choose **MQTT** for messaging if you have a bad network, tens of thousands of clients, or the need for a **lightweight push-based messaging** solution, then MQTT is the right choice. Elsewhere, **Kafka, a powerful event streaming platform,** is probably a great choice for messaging, data integration, and data processing. In many IoT use cases, the architecture combines both technologies.

Example: Predictive Maintenance with 100,000 Connected Cars

We have built a "simple demo" with Confluent and HiveMQ running on cloud-native Kubernetes infrastructure. The use case implements **data processing from 100,000 connected cars and real-time predictive maintenance** with TensorFlow:
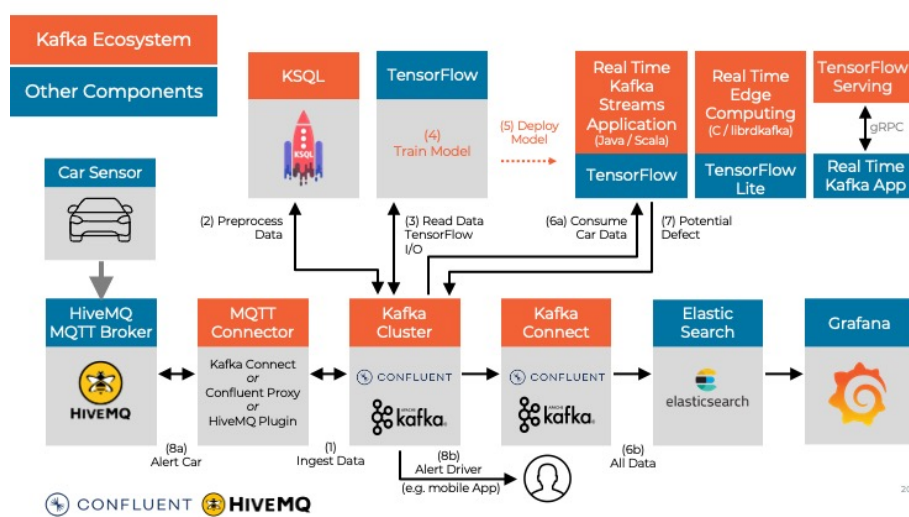
More details, a video of the demo, and the code are available on Github: 100,000 Connected Cars and Predictive Maintenance in Real-Time with MQTT, Kafka, Kubernetes, and TensorFlow.

Additionally, I blogged a lot about Kafka and Machine Learning. For instance, take a look at streaming machine learning without a data lake with Kafka and TensorFlow.

Further Slides, Articles, and Demos

I already created a lot of material (including blogs, slides, videos) around Kafka and MQTT. Check this out if you need to learn about the different broker alternatives, integration options, and best practices for the combination.

- Best Practices for Streaming IoT Data with MQTT and Apache Kafka
- Integration between Apache Kafka and MQTT with Kafka Connect
- Apache Kafka Native MQTT at Scale with Confluent Cloud and Waterstream
- HiveMQ's native Kafka Integration

The following slide deck covers the content of this blog series on a high level:

Click on the button to load the content from www.slideshare.net.

Load content

Kafka + MQTT = Match Made in Heaven

In conclusion, **Apache Kafka and MQTT are a perfect combination for many IoT use cases**. Follow the blog series to learn about use cases such as connected vehicles, manufacturing, mobility services, and smart city. Every blog post also includes real-world deployments from companies across industries. It is key to understand the different architectural options to make the right choice for your project.

What are your experiences and plans in IoT projects? What use case and architecture did you implement? Let's connect on LinkedIn and discuss it! Stay informed about new blog posts by subscribing to my newsletter.

SHARE 25   TWEET

Dont' miss my next post. Subscribe!

First name

Last name

Email Address *

SIGN UP

We don't spam! Read more in our privacy policy

Kai Waehner

builds cloud-native event streaming infrastructures for real-time data processing and analytics

VIEW COMMENTS (2)