Kafka

Course Summary

Description

Kafka is definitely one of the leading distributed messaging queues, found in use everywhere. More correctly termed “a distributed messaging platform,” it provides three key capabilities: publish-and-subscribe, store streams, process streams. This course introduces the students to using Kafka and architecting systems with it.

Objective

- Overview of Streaming technologies
- Kafka concepts and architecture
- Programming using Kafka API
- Kafka Streams
- Monitoring Kafka
- Tuning / Troubleshooting Kafka

Audience

This course is designed for Developers and Architects.

Topics

- Introduction to Streaming Systems
- Introduction
- Programming With Kafka
- Kafka Streams
- Administering Kafka
- Monitoring and Instrumenting Kafka
- Final workshop (time permitting)

Prerequisite

Prior to taking this course, you should be:
- comfortable with Java programming language (programming exercises are in Java)
- comfortable in Linux environment (be able to navigate Linux command line, run commands)

Duration

Two Days
Course Outline

I. Introduction to Streaming Systems
   A. Fast data
   B. Streaming architecture
   C. Lambda architecture
   D. Message queues
   E. Streaming processors

II. Introduction
   A. Architecture
   B. Comparing Kafka with other queue systems (JMS / MQ)
   C. Kafka concepts: Messages, Topics, Partitions, Brokers, Producers, commit logs
   D. Kafka & Zookeeper
   E. Producing messages
   F. Consuming messages (Consumers, Consumer Groups)
   G. Message retention
   H. Scaling Kafka
      • Labs
        • Getting Kafka up and running
        • Using Kafka utilities

III. Programming With Kafka
   A. Configuration parameters
   B. Producer API (Sending messages to Kafka)
   C. Consumer API (consuming messages from Kafka)
   D. Commits, Offsets, Seeking
   E. Schema with Avro
      • Lab
        • Writing Kafka clients in Java
        • Benchmarking Producer APIs

IV. Kafka Streams
   A. Streams overview and architecture
   B. Streams use cases and comparison with other platforms
   C. Learning Kafka Streaming concepts (KStream, KTable, KStore)
   D. KStreaming operations (transformations, filters, joins, aggregations)
      • Labs
        • Kafka Streaming labs

V. Administering Kafka
   A. Hardware / Software requirements
   B. Deploying Kafka
   C. Configuration of brokers / topics / partitions / producers / consumers
   D. Security: How secure Kafka cluster, and secure client communications (SASL, Kerberos)
   E. Monitoring: monitoring tools
   F. Capacity Planning: estimating usage and demand
   G. Troubleshooting: failure scenarios and recovery

VI. Monitoring and Instrumenting Kafka
   A. Monitoring Kafka
   B. Instrumenting with Metrics library
      • Labs
        • Monitor Kafka cluster
        • Instrument Kafka applications and monitor their performance

VII. Final workshop (time permitting)
   A. Students will build an end-to-end application simulating web traffic and send metrics to Grafana.