

Exercise 7: Stream Processing with Kafka Streams (Part I)

Deadline: 02.06.2024; 23:59 CEST

Project Abstract

In the eight week's exercise on Event-driven and Process-oriented Architectures we will investigate the use of Kafka Streams for stateless stream processing.

① Labs

In the practical part of the lecture we ask you to work on the following labs to get started with Kafka Streams:

- Lab11 – YellingApp¹: This lab acts as a basic hello-world example for Kafka Streams with single-event processing. Here you find a simple topology consisting of one event producer, one processor to translate the original event, and one event consumer. Try to run the project and study the implementation of the stream processing app.
- Lab12Part1 – CryptoSentiment²: This project shows the implementation of a more complex stream processing topology for stateless processing as introduced in the lecture. The app features event processors to filter, branch, translate, merge and enrich events. It also demonstrates how to serialize and deserialize records using JSON and Avro. Try to run the project and study the implementation of the stream processing app.
- Lab12Part2 – Eye Tracking Events Processing (Part1)³: Building upon the stream processing techniques introduced in the CryptoSentiment lab, this project simulates the generation, processing, and consumption of eye-tracking gaze data. The data contains information about the position of the eyes and the size of the pupils. The project consists of three main components: a gaze producer, an event processing topology, and a gaze consumer. Try to run the project, study the implementation of the event processing topology and extend it as described in the *Exercise* Section of the project README.

② Suggestions for Software Project

For your software application in Assignment 2 we suggest that you implement one or several stream processing applications including both stateless and stateful stream processing:

- At least several stateless operations (Week 8)
- Usage of both streams and tables (Week 9)
- Consider data from more than one stream (Week 9)
- Interactive queries (Week 9)
- Windowed operations (Week 10)

We suggest to use Avro schemes for serialization and deserialization of records in Kafka (registry-less or with a schema registry).

¹<https://github.com/scs-edpo/lab11-kafka-streams-yellingapp>

²<https://github.com/scs-edpo/lab12Part1-kafka-streams-cryptosentiment>

³<https://github.com/scs-edpo/lab12Part2-kafka-streams-eyeTracking1>

Software Project Expectations

The labs in Part 1 of this sheet are meant to be done during class and are **not graded**. In Part 2 we expect you to prepare a report on the project-related implementations that you have done in your group. For each developed stream processing application the **topology** should be described in **graphical form** and you should **elaborate** on the chosen **stream processing patterns**. Feel free to use the provided data sets for your experiments.

Hand-in Instructions

The report and implementation for Part 2 will **be graded** as part of the software application for your group to be handed in for the second part of the lecture until the semester end by the deadline indicated on the top of this sheet. Each group member **must explicitly indicate** which part she/he/they has/have been working on. Please approach the tutors for individual feedback regarding this assignment.