Kafka Monitoring Demo

mimacom
stay ahead

Plan

1. General Presentation of the **docker-compose components**
2. Introduction to the tools : **Prometheus and Grafana**
3. The existing beans through **JConsole**
4. Sending metics to **Prometheus with an agent**
5. Sending metics to **Prometheus through Spring Boot** libraries
6. Tour of **Grafana Dashboards** and **Key Metrics**

mimacom
stay ahead

Links

- JConsole: localhost:10092
- broker metrics: http://localhost:7071/metrics
- app metrics: http://localhost:8080/actuator/prometheus
- prometheus : http://localhost:9090
- grafana : http://localhost:3000

Appendix

- Throughput:
  - Records Processed per Second: The rate at which records are being processed by the Kafka Streams application.
  - Bytes Read/Write per Second: The amount of data being read from and written to Kafka topics per second.
- Latency:
  - Processing Latency: The time taken to process each record.
  - End-to-End Latency: The time from when a record is produced to when it is fully processed.

- Error Rates:
  - Deserialization Errors: Number of errors encountered while deserializing records.
  - Production Errors: Number of errors encountered while producing records to Kafka topics.
- Consumer Lag:
  - Consumer Lag: The difference between the latest offset in the Kafka topic and the current offset processed by the consumer. High lag indicates the consumer is falling behind.

- Task Metrics:
  - Task Creation/Destruction Rate: The rate at which tasks are created and destroyed, which can indicate rebalancing activities.
  - Active/Standby Tasks: Number of active and standby tasks, indicating the current load distribution.
- Commit Metrics:
  - Commit Rate: Frequency of commit operations.
  - Commit Latency: Time taken to commit the processed records.

- State Store Metrics:
  - Store Size: Size of the state stores, which can indicate the volume of data being maintained.
  - Read/Write Rate: Rate at which read and write operations are performed on state stores.
- Thread Metrics:
  - Thread Count: Number of threads currently running in the Kafka Streams application.
  - Thread Idle Ratio: Percentage of time threads are idle, which can indicate the application's efficiency.