BWT Task-10 Exercise

Submitted By: ABRAR SAEED

1. What is Apache Kafka, and how it assists in making Streaming Jobs?

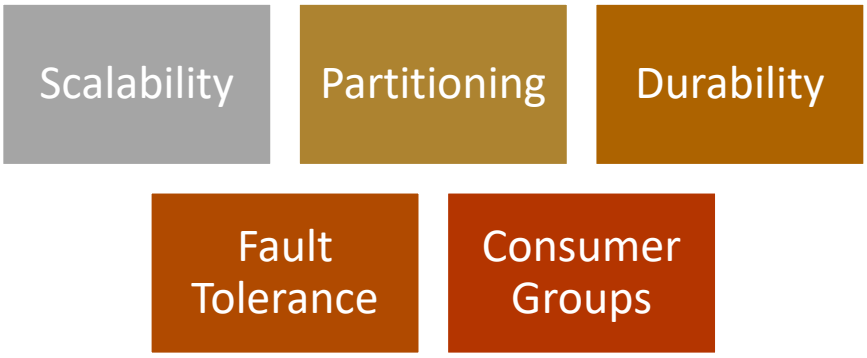
Apache Kafka is a distributed streaming platform which is described to process real-time data feeds and is an open source. LinkedIn created it in the first instance though is currently under the Apache Software Foundation. The initial and most central idea that Kafka offers is the segmented commit log which can be thought of as a structured log of the data written that can be read in real time.

It is commonly used for:

- Real-time data streaming: Capturing and processing data as it happens, in real-time.
- Building data pipelines: Moving data between different systems, such as databases, applications, and analytics tools.
- Event sourcing: Maintaining a consistent record of events in distributed systems.
- Log aggregation: Collecting and organizing log data from various sources.

2. In what cases do we prefer streaming jobs?

Kafka is highly suitable for creating streaming jobs due to its architecture and features:



- Ideal for real time processing due to its ability to handle high throughput of messages with low latency.
- Performance and Fault Tolerance is increased because data is partitioned for parallel processing.
- Data is not lost even if a server crashes.
- Data is replicated across multiple servers.
- Allows multiple consumers to read from the same topic.

3. Brokers, Zookeeper, Kafka Server, Kafka Clients, KRaft, Kafka Cluster

- **Brokers**: Kafka brokers are servers used in storage and management of data. Broker obtains messages from producers, assigns offsets and stores them to the respective

topic partitions. These messages are also given by brokers to the consumers. When setting up a Kafka cluster usually there are many brokers as this ensures availability and scalability of the system.

- **Zookeeper:** Zookeeper is another platform that is employed on Kafka to address the issues of coordination of the cluster, the broker's metadata, election of leaders on the partition as well as the configuration of the framework. Kafka brokers are co-coordinating components and a Zookeeper makes sure that these brokers are in different states. However, Kafka is moving toward the non-Zookeeper architecture using KRaft.
- **Kafka Server:** This term sometimes refers to Kafka broker instance in the server. Indeed, it accepts raw data from producers and stores it and distributes it at the request of the consumers. The Kafka server is the center of the Kafka cluster's messaging responsibility.
- **Kafka Clients:** These are the producers and consumers making use of the Kafka cluster. Kafka has a set of topics where producers write data to the topics while consumers read data from the topics. Kafka consumers can be developed in different programming languages such as Java, python, go etc.
- **Kraft:** KRaft is Kafka's new architecture to replace the Zookeeper with consensus protocol based on Raft. KRaft reduces external dependencies and alleviates the overhead of dependency management The Key-Value store is relocated from External Services to Kafka brokers.
- **Kafka Cluster:** In Kafka, the term 'cluster' is used to refer to a Kafka brokers group. It gives high availability and scalability by dispersing data across multiple nodes (brokers) as well as partitioning topics. In terms of scalability Kafka is very efficient since a Kafka cluster can generate high throughputs from large data volumes.