Faculty of Engineering & Applied Science

Cloud Computing - 74293

Project Milestone 2:

Data Ingestion Software - Kafka Clusters

Taha Hashmat - 100689792

Google Link for the Recorded Videos:

https://drive.google.com/drive/folders/15Ui2cvfRFwTqpWsbf58LWiyAqe-lp_qK?usp=sharing

Question 4

What is EDA? What are its advantages and disadvantages?

Answer 4

EDA: Event Driven Architecture is a software architecture which prioritizes events as compared to resting data in general. Even though big data involves large chunks of data being processed simultaneously, the more data you have the longer it will take to process it. EDA works around this drawback, and focuses on data being used to change it state (i.e. an event)

Advantages:

- Loosely coupled, which allows programmers to use multiple different technologies for different components (producers and consumers) while constructing an event driven architecture
- Real time data processing which ensures that data is updated every millisecond
- EDA can be considered a distributed model. For this reason an Event Driven Architecture is considered to be highly scalable in accordance to the number of events being processed
- Branching off of the loose coupling, EDA's are highly versatile since unrelated components can be used in different settings and networks i.e. are functionally independent

Disadvantages:

- There is a possibility that events may be replicated across the architecture which can lead to possible miscommunication
- Since each application that utilizes Event Driven Architecture consists of hundreds of producers and consumers, keeping track of each component and making sure it is functional and working error free is quite difficult, thus third party tools are required for error detection and handling.

Question 5

In Kafka, what's meant by cluster, broker, topic, replica, partition, zookeeper, controller, leader, consumer, producer, and consumer group?

1. Cluster: Defined as one or more Kafa brokers running simultaneously. Collectively this is known as a Kafka Cluster.
2. Broker: Containers running Kafka, by performing read and write operations and managing partitions and its replications
3. Topic: Vaguely topics can be compared to databases, they are used by producers to perform write operations and are by used consumers to perform read operations
4. Replica: Kafka involves creating multiple copies of data for security purposes which is why they are distributed across multiple brokers. These copies are called replicas
5. Partition: Each topic in Kafka can be split into partitions which are basically split up 'logs' of data of a topic
6. Zookeeper: In case a Kafka system broker fails, a zoo-keeper is used to reassign brokers, and notify the producers and consumers of the presence of a new broker
7. Controller: Amongst the many brokers, one broker is responsible for managing all the replicas and partitions that have been created in the cluster.
8. Leader: The leader (A Kafka broker) is responsible for all the operations on the rest of the partitions which are its followers. The leader is updated with read/write operations and all of its followers access the leader to update themselves.
9. Consumer: A consumer is used to subscribe to the messages that have been published by the producer.
10. Producer: A producer pushes or publishes messages/records to a cluster or more specifically a broker.
11. Consumer Group: Consumer groups consist of multiple consumers that are subscribed to messages published by producers. Consumers from the same consumer group that are subscribed to a topic will each receive infiltration from a different subset of the partition of that topic.