**iDigiCloudTech Pvt. Ltd.**

**Research on Kafka Concurrent consumer processing**

**What is Apache Kafka?**

Apache Kafka is a distributed data store optimized for ingesting and processing streaming data in real-time. Streaming data is data that is continuously generated by thousands of data sources, which typically send the data records in simultaneously. A streaming platform needs to handle this constant influx of data and process the data sequentially and incrementally.

Kafka provides three main functions to its users:

* Publish and subscribe to streams of records
* Effectively store streams of records in the order in which records were generated
* Process streams of records in real-time

Kafka is primarily used to build real-time streaming data pipelines and applications that adapt to the data streams. It combines messaging, storage, and stream processing to allow storage and analysis of both historical and real-time data.

Basic Concepts for Kafka

* **Producer**:

An application that is the source of the data stream is what we call a producer. To generate tokens or messages and further publish them to one or more topics in the Kafka cluster, we use Apache Kafka Producer.

* **Consumer**:

Kafka consumers are typically part of a consumer group. When multiple consumers are subscribed to a topic and belong to the same consumer group, each consumer in the group will receive messages from a different subset of the partitions in the topic.

* **Broker**:

A Broker is a Kafka server that runs in a Kafka Cluster. Kafka Brokers form a cluster. The Kafka Cluster consists of many Kafka Brokers on many servers. Broker sometimes refer to more of a logical system or as Kafka as a whole

* **Cluster**:

A Kafka cluster consists of one or more servers (Kafka brokers) running Kafka. Producers are processes that push records into Kafka topics within the broker. A consumer pulls records off a Kafka topic.

* **Topic**. A Topic is a category/feed name to which records are stored and published. As said before, all Kafka records are organized into topics. Producer applications write data to topics and consumer applications read from topics.
* **Partition**:

Partitioning takes the single topic log and breaks it into multiple logs, each of which can live on a separate node in the Kafka cluster. This way, the work of storing messages, writing new messages, and processing existing messages can be split among many nodes in the cluster.

* **Offset**:

The offset is a simple integer number that is used by Kafka to maintain the current position of a consumer. That's it. The current offset is a pointer to the last record that Kafka has already sent to a consumer in the most recent poll. So, the consumer doesn't get the same record twice because of the current offset.

* **Consumer Groups:**

Kafka assigns the partitions of a topic to the consumer in a group so that each partition is consumed by exactly one consumer in the group. ... Kafka guarantees that a message is only ever read by a single consumer in the group. Consumers can see the message in the order they were stored in the log.

* **ZooKeeper** is used in distributed systems for service synchronization and as a naming registry. When working with Apache Kafka, ZooKeeper is primarily used to track the status of nodes in the Kafka cluster and maintain a list of Kafka topics and messages.

Some useful links for understanding:

https://kafka.apache.org/documentation/

https://www.tibco.com/reference-center/what-is-apache-kafka

https://aws.amazon.com/msk/what-is-kafka/

Download Apache Kafka from the below link:

<https://kafka.apache.org/downloads>

**Please follow the below steps to start the Apache Kafka**

1. To Start zookeeper, open a terminal and use the below command

.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties

A picture containing text

Description automatically generated

1. To start kafka, open another terminal and use below command

.\bin\windows\kafka-server-start.bat .\config\server.properties

A screenshot of a computer

Description automatically generated with medium confidence

1. Created a topic named “IDC-topic” with replication factor as 1 and 3 partitions

Open another terminal and use the below command

.\bin\windows\kafka-topics.bat --create --topic IDC-topic --bootstrap-server localhost:9092 --replication-factor 1 --partitions 3

Text

Description automatically generated

1. Create the project using dependencies and import it into the IDE.
2. The code is attached herewith.

Created Two packages

**Producer**

* Create property objects for Producer
* Create the Producer
* Send Data Asynchronous
* flush and close Procedure

**Consumer**

* Create property object for consumer
* Create Consumer
* Poll and consume records

1. To launch consumer, open another terminal and use below command

.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic IDC-topic --group java.group

Text

Description automatically generated

1. Published 20 messages to the Producer IDC\_51 to IDC\_70, which can be validated in the console, and below is the SS and log for the same, messages were distributed among the partitions, and offsets were incremented from the latest for each partition.

A picture containing text

Description automatically generated

21:05:36.530 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 2,

Offset: 26

@TimeStamp: 1645803336492

21:05:36.530 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 2,

Offset: 27

@TimeStamp: 1645803336502

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 2,

Offset: 28

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 2,

Offset: 29

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 2,

Offset: 30

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 2,

Offset: 31

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 10

@TimeStamp: 1645803336502

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 11

@TimeStamp: 1645803336502

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 12

@TimeStamp: 1645803336502

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 13

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 14

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 15

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 16

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 17

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 0,

Offset: 18

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 1,

Offset: 8

@TimeStamp: 1645803336502

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 1,

Offset: 9

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 1,

Offset: 10

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 1,

Offset: 11

@TimeStamp: 1645803336503

21:05:36.531 [kafka-producer-network-thread | producer-1] INFO com.kafka.kafkademo.kafka.Producer -

Received record Metadata.

Topic: IDC-topic,

Pratition: 1,

Offset: 12

@TimeStamp: 1645803336503

21:05:36.532 [main] INFO org.apache.kafka.clients.producer.KafkaProducer - [Producer clientId=producer-1] Closing the Kafka producer with timeoutMillis = 9223372036854775807 ms.

21:05:36.532 [kafka-producer-network-thread | producer-1] DEBUG org.apache.kafka.clients.producer.internals.Sender - [Producer clientId=producer-1] Beginning shutdown of Kafka producer I/O thread, sending remaining records.

21:05:36.538 [kafka-producer-network-thread | producer-1] DEBUG org.apache.kafka.clients.producer.internals.Sender - [Producer clientId=producer-1] Shutdown of Kafka producer I/O thread has completed.

21:05:36.539 [main] INFO org.apache.kafka.common.metrics.Metrics - Metrics scheduler closed

21:05:36.539 [main] INFO org.apache.kafka.common.metrics.Metrics - Closing reporter org.apache.kafka.common.metrics.JmxReporter

21:05:36.539 [main] INFO org.apache.kafka.common.metrics.Metrics - Metrics reporters closed

21:05:36.540 [main] INFO org.apache.kafka.common.utils.AppInfoParser - App info kafka.producer for producer-1 unregistered

21:05:36.540 [main] DEBUG org.apache.kafka.clients.producer.KafkaProducer - [Producer clientId=producer-1] Kafka producer has been closed

Process finished with exit code 0

1. When Consumer consumes the messages posted by the producer, those are distributed among partitions, and below is the log for the same.

21:13:15.048 [main] DEBUG org.apache.kafka.clients.consumer.internals.Fetcher - [Consumer clientId=consumer-IDC-group-consumer-1, groupId=IDC-group-consumer] Sending READ\_UNCOMMITTED IncrementalFetchRequest(toSend=(IDC-topic-2, IDC-topic-0, IDC-topic-1), toForget=(), toReplace=(), implied=(), canUseTopicIds=True) to broker Pranay:9092 (id: 0 rack: null)

21:13:15.048 [main] DEBUG org.apache.kafka.clients.NetworkClient - [Consumer clientId=consumer-IDC-group-consumer-1, groupId=IDC-group-consumer] Sending FETCH request with header RequestHeader(apiKey=FETCH, apiVersion=13, clientId=consumer-IDC-group-consumer-1, correlationId=11) and timeout 30000 to node 0: FetchRequestData(clusterId=null, replicaId=-1, maxWaitMs=500, minBytes=1, maxBytes=52428800, isolationLevel=0, sessionId=1721812637, sessionEpoch=1, topics=[FetchTopic(topic='IDC-topic', topicId=Dj6Zk7\_nTKqiPhfNUacUbA, partitions=[FetchPartition(partition=2, currentLeaderEpoch=0, fetchOffset=32, lastFetchedEpoch=-1, logStartOffset=-1, partitionMaxBytes=1048576), FetchPartition(partition=0, currentLeaderEpoch=0, fetchOffset=19, lastFetchedEpoch=-1, logStartOffset=-1, partitionMaxBytes=1048576), FetchPartition(partition=1, currentLeaderEpoch=0, fetchOffset=13, lastFetchedEpoch=-1, logStartOffset=-1, partitionMaxBytes=1048576)])], forgottenTopicsData=[], rackId='')

21:13:15.058 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_51,Topic: IDC-topic,

Pratition: 2,

Offset: 26

@TimeStamp: 1645803336492

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_56,Topic: IDC-topic,

Pratition: 2,

Offset: 27

@TimeStamp: 1645803336502

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_57,Topic: IDC-topic,

Pratition: 2,

Offset: 28

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_59,Topic: IDC-topic,

Pratition: 2,

Offset: 29

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_66,Topic: IDC-topic,

Pratition: 2,

Offset: 30

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_67,Topic: IDC-topic,

Pratition: 2,

Offset: 31

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_52,Topic: IDC-topic,

Pratition: 0,

Offset: 10

@TimeStamp: 1645803336502

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_53,Topic: IDC-topic,

Pratition: 0,

Offset: 11

@TimeStamp: 1645803336502

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_55,Topic: IDC-topic,

Pratition: 0,

Offset: 12

@TimeStamp: 1645803336502

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_60,Topic: IDC-topic,

Pratition: 0,

Offset: 13

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_61,Topic: IDC-topic,

Pratition: 0,

Offset: 14

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_62,Topic: IDC-topic,

Pratition: 0,

Offset: 15

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_63,Topic: IDC-topic,

Pratition: 0,

Offset: 16

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_65,Topic: IDC-topic,

Pratition: 0,

Offset: 17

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_70,Topic: IDC-topic,

Pratition: 0,

Offset: 18

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_54,Topic: IDC-topic,

Pratition: 1,

Offset: 8

@TimeStamp: 1645803336502

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_58,Topic: IDC-topic,

Pratition: 1,

Offset: 9

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_64,Topic: IDC-topic,

Pratition: 1,

Offset: 10

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_68,Topic: IDC-topic,

Pratition: 1,

Offset: 11

@TimeStamp: 1645803336503

21:13:15.059 [main] INFO com.kafka.kafkademo.kafka.Consumer -

Received record Metadata.

Key: key\_69,Topic: IDC-topic,

Pratition: 1,

Offset: 12

@TimeStamp: 1645803336503