**Department of Computer Science and Engineering (Data Science)**

**Subject: Big Data Engineering (DJ19DSL604)**

**AY: 2022-23**

**Experiment 4**

**(Messaging Service)**

Name: Kresha Shah

SAP ID: 60009220080

**Aim:** Implement messaging system using Kafka.

**Theory:**

**Kafka Overview**

Apache Kafka is a distributed publish-subscribe messaging system and a robust queue that can handle a high volume of data and enables you to pass messages from one end-point to another. Kafka is suitable for both offline and online message consumption. Kafka messages are persisted on the disk and replicated within the cluster to prevent data loss. Kafka is built on top of the ZooKeeper synchronization service. It integrates very well with Apache Storm and Spark for real-time streaming data analysis.

**Need for Kafka**

Kafka is a unified platform for handling all the real-time data feeds. Kafka supports low latency message delivery and gives guarantee for fault tolerance in the presence of machine failures. It has the ability to handle a large number of diverse consumers. Kafka is very fast, performs 2 million writes/sec. Kafka persists all data to the disk, which essentially means that all the writes go to the page cache of the OS (RAM). This makes it very efficient to transfer data from page cache to a network socket.

**Kafka Cluster Architecture**

**Department of Computer Science and Engineering (Data Science)**
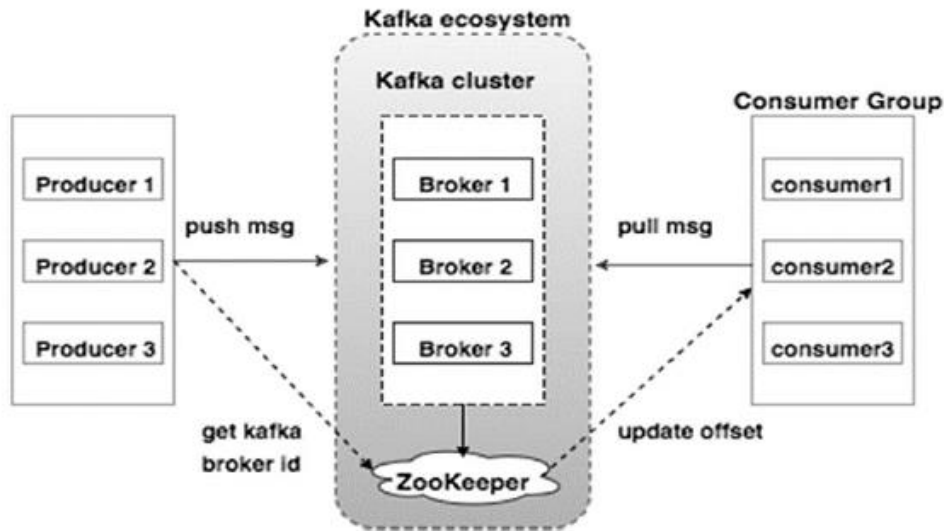


**Broker**

Kafka cluster typically consists of multiple brokers to maintain load balance. Kafka brokers are stateless, so they use ZooKeeper for maintaining their cluster state. One Kafka broker instance can handle hundreds of thousands of reads and writes per second and each broker can handle TB of messages without performance impact. Kafka broker leader election can be done by ZooKeeper.

**ZooKeeper**

ZooKeeper is used for managing and coordinating Kafka broker. ZooKeeper service is mainly used to notify producer and consumer about the presence of any new broker in the Kafka system or failure of the broker in the Kafka system. As per the notification received by the Zookeeper regarding presence or failure of the broker then producer and consumer takes decision and starts coordinating their task with some other broker.
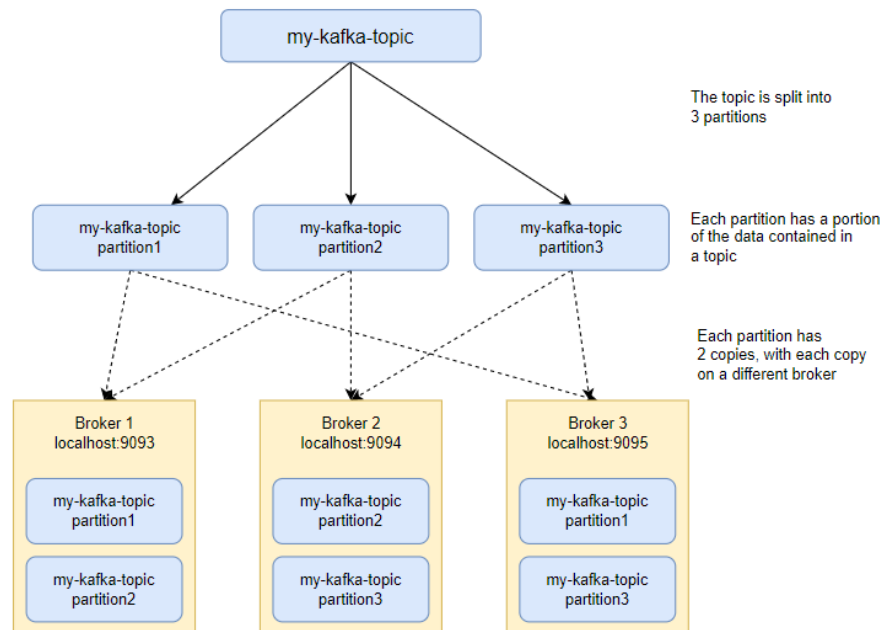
**Producers**

Producers push data to brokers. When the new broker is started, all the producers search it and automatically sends a message to that new broker. Kafka producer doesn't wait for acknowledgements from the broker and sends messages as fast as the broker can handle.

![SVKM logo] **Shri Vile Parle Kelavani Mandal's**
**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

**Department of Computer Science and Engineering (Data Science)**

**Consumers**

Since Kafka brokers are stateless, which means that the consumer has to maintain how many messages have been consumed by using partition offset. If the consumer acknowledges a particular message offset, it implies that the consumer has consumed all prior messages. The consumer issues an asynchronous pull request to the broker to have a buffer of bytes ready to consume. The consumers can rewind or skip to any point in a partition simply by supplying an offset value. Consumer offset value is notified by ZooKeeper.



**Lab Assignment:**

1. Installation of Kafka 2.13-3.0.0 locally.
2. Create a Kafka local cluster with 3 brokers and create a topic to which the data should belong.
3. Send message to a topic using producer and read the data from the cluster using a consumer.
4. Testing Replication after a failed broker.

**Shri Vile Parle Kelavani Mandal's**
**DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING**
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

**Department of Computer Science and Engineering (Data Science)**

```
cs-ds@kmaster: ~/Downloads/kafka_2.13-3.6.1

cs-ds@kmaster:~/Downloads$ tar -xzf kafka_2.13-3.6.1.tgz
cs-ds@kmaster:~/Downloads$ cd kafka_2.13-3.6.1
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/zookeeper-server-start.sh config
/zookeeper.properties
[2024-02-21 11:03:37,156] INFO Reading configuration from: config/zookeeper.prop
erties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-02-21 11:03:37,157] WARN config/zookeeper.properties is relative. Prepend
./ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumPeerC
onfig)
[2024-02-21 11:03:37,159] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zoo
keeper.server.quorum.QuorumPeerConfig)
[2024-02-21 11:03:37,159] INFO secureClientPort is not set (org.apache.zookeeper
.server.quorum.QuorumPeerConfig)
[2024-02-21 11:03:37,159] INFO observerMasterPort is not set (org.apache.zookeep
er.server.quorum.QuorumPeerConfig)
[2024-02-21 11:03:37,159] INFO metricsProvider.className is org.apache.zookeeper
.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumP
eerConfig)
[2024-02-21 11:03:37,160] INFO autopurge.snapRetainCount set to 3 (org.apache.zo
okeeper.server.DatadirCleanupManager)
[2024-02-21 11:03:37,160] INFO autopurge.purgeInterval set to 0 (org.apache.zook
eeper.server.DatadirCleanupManager)
[2024-02-21 11:03:37,160] INFO Purge task is not scheduled. (org.apache.zookeepe
r.server.DatadirCleanupManager)
```

```
cs-ds@kmaster: ~/Downloads/kafka_2.13-3.6.1

cs-ds@kmaster:~/Downloads$ bin/kafka-server-start.sh config/server.properties
bash: bin/kafka-server-start.sh: No such file or directory
cs-ds@kmaster:~/Downloads$ cd kafka_2.13-3.6.1
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-server-start.sh config/ser
ver.properties
[2024-02-21 11:05:32,893] INFO Registered kafka:type=kafka.Log4jController MBean
 (kafka.utils.Log4jControllerRegistration$)
[2024-02-21 11:05:33,065] INFO Setting -D jdk.tls.rejectClientInitiatedRenegotia
tion=true to disable client-initiated TLS renegotiation (org.apache.zookeeper.co
mmon.X509Util)
[2024-02-21 11:05:33,120] INFO Registered signal handlers for TERM, INT, HUP (or
g.apache.kafka.common.utils.LoggingSignalHandler)
[2024-02-21 11:05:33,121] INFO starting (kafka.server.KafkaServer)
[2024-02-21 11:05:33,121] INFO Connecting to zookeeper on localhost:2181 (kafka.
server.KafkaServer)
[2024-02-21 11:05:33,129] INFO [ZooKeeperClient Kafka server] Initializing a new
 session to localhost:2181. (kafka.zookeeper.ZooKeeperClient)
[2024-02-21 11:05:33,137] INFO Client environment:zookeeper.version=3.8.3-6ad6d3
64c7c0bcf0de452d54ebefa3058098ab56, built on 2023-10-05 10:34 UTC (org.apache.zo
okeeper.ZooKeeper)
[2024-02-21 11:05:33,137] INFO Client environment:host.name=kmaster (org.apache.
zookeeper.ZooKeeper)
[2024-02-21 11:05:33,137] INFO Client environment:java.version=1.8.0_392 (org.ap
ache.zookeeper.ZooKeeper)
```

```
cs-ds@kmaster: ~/Downloads/kafka_2.13-3.6.1

cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-topics.sh --create --topic
 quickstart-events --bootstrap-server localhost:9092
Created topic quickstart-events.
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ $ bin/kafka-topics.sh --describe --t
opic quickstart-events --bootstrap-server localhost:9092
$: command not found
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-topics.sh --describe --top
ic quickstart-events --bootstrap-server localhost:9092
Topic: quickstart-events        TopicId: dqzL_nLwRE6-GCFxgG5ODg PartitionCount:
1       ReplicationFactor: 1    Configs:
        Topic: quickstart-events        Partition: 0    Leader: 0       Replicas
: 0     Isr: 0
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-console-producer.sh --topi
c quickstart-events --bootstrap-server localhost:9092
>This is trinity event
>this is IDPT event
>This is hackniche
>^Ccs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-console-consumer.sh --t
opic quickstart-events --from-beginning --bootstrap-server localhost:9092
This is trinity event
this is IDPT event
^CProcessed a total of 2 messages
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-console-consumer.sh --topi
c quickstart-events --from-beginning --bootstrap-server localhost:9092
```

```
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-console-consumer.sh --topi
c quickstart-events --from-beginning --bootstrap-server localhost:9092
This is trinity event
this is IDPT event
```

```
cs-ds@kmaster: ~/Downloads/kafka_2.13-3.6.1

^CProcessed a total of 2 messages
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ echo "plugin.path=libs/connect-file-
3.6.1.jar"
plugin.path=libs/connect-file-3.6.1.jar
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ echo -e "foo\nbar" > test.txt
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/connect-standalone.sh config/con
nect-standalone.properties config/connect-file-source.properties config/connect-
file-sink.properties
[2024-02-21 11:27:02,964] INFO Kafka Connect worker initializing ... (org.apache
.kafka.connect.cli.AbstractConnectCli:114)
[2024-02-21 11:27:02,965] INFO WorkerInfo values:
        jvm.args = -Xms256M, -Xmx2G, -XX:+UseG1GC, -XX:MaxGCPauseMillis=20, -XX:
InitiatingHeapOccupancyPercent=35, -XX:+ExplicitGCInvokesConcurrent, -XX:MaxInli
neLevel=15, -Djava.awt.headless=true, -Dcom.sun.management.jmxremote, -Dcom.sun.
management.jmxremote.authenticate=false, -Dcom.sun.management.jmxremote.ssl=fals
e, -Dkafka.logs.dir=/home/cs-ds/Downloads/kafka_2.13-3.6.1/bin/../logs, -Dlog4j.
configuration=file:bin/../config/connect-log4j.properties
        jvm.spec = Private Build, OpenJDK 64-Bit Server VM, 1.8.0_392, 25.392-b0
8
        jvm.classpath = /home/cs-ds/Downloads/kafka_2.13-3.6.1/bin/../libs/activ
ation-1.1.1.jar:/home/cs-ds/Downloads/kafka_2.13-3.6.1/bin/../libs/aopalliance-r
epackaged-2.6.1.jar:/home/cs-ds/Downloads/kafka_2.13-3.6.1/bin/../libs/argparse4
j-0.7.0.jar:/home/cs-ds/Downloads/kafka_2.13-3.6.1/bin/../libs/audience-annotati
ons-0.12.0.jar:/home/cs-ds/Downloads/kafka_2.13-3.6.1/bin/../libs/caffeine-2.9.3
```

```
[2024-02-21 11:28:16,345] WARN [Consumer clientId=console-consumer, groupId=cons
ole-consumer-7929] Error while fetching metadata with correlation id 2 : {connec
t-test=LEADER_NOT_AVAILABLE} (org.apache.kafka.clients.NetworkClient)
echo Another line>> test.txt
^CProcessed a total of 0 messages
schema:type:string: command not found
schema:type:string: command not found
...: command not found
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ echo -e "foo\nbar"
foo
bar
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ echo -e "foo\nbar" test.txt
foo
bar test.txt
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ more test.txt
foo
bar
cs-ds@kmaster:~/Downloads/kafka_2.13-3.6.1$ bin/kafka-console-consumer.sh --boot
strap-server localhost:9092 --topic connect-test --from-beginning
{"schema":{"type":"string","optional":false},"payload":"foo"}
{"schema":{"type":"string","optional":false},"payload":"bar"}
...
echo Another line>> test.txt
```