

A REPORT ON

KAFKA Deployment And Creating Publisher/Consumer
For ReST Based Services



International Institute Of Information
Technology, Bangalore
Computer Science and Engineering

Guidance by

Prof. Samar Shailendra

Submitted by

Shubham Gupta (IMT2016118)

Suparn S Lele (MT2018122)

INDEX

1. Kafka Introduction

2. Components

2.1 Zoo-Keeper

2.2 Broker

2.3 Producer

2.4 Consumers

2.5 Topic

3. DataSet

4. Screenshots Of Steps

5. Conclusion

6. Git Hub Link

7. References

1.0 Kafka Introduction:

Apache Kafka is an [open-source stream-processing](#) software platform developed by [LinkedIn](#) and donated to the [Apache Software Foundation](#), written in [Scala](#) and [Java](#). The project aims to provide a unified, high-throughput, low-latency platform for handling real-time data feeds.

Kafka uses a binary [TCP](#) design that is optimized for efficiency and relies on a "message set" abstraction that naturally groups messages together to reduce the overhead of the network roundtrip. This "leads to larger network packets, larger sequential disk operations, contiguous memory blocks [...]" which allows Kafka to turn a bursty stream of random message writes into linear writes."

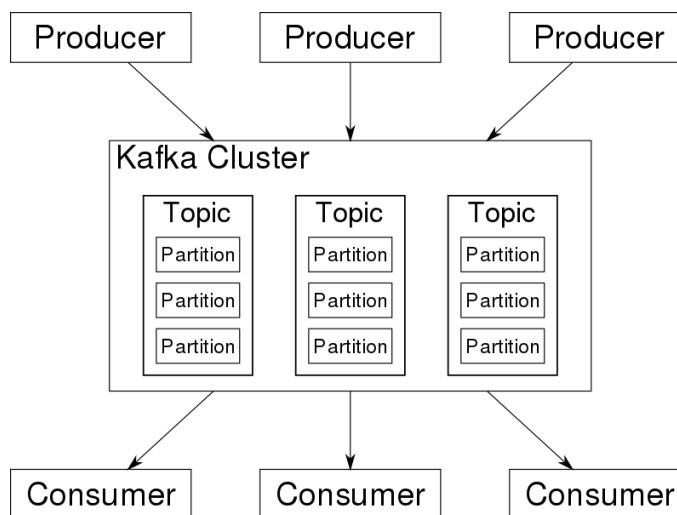


Fig 1. Kafka Architecture

2.0 Components :

2.1 Zookeeper

Apache Zookeeper is a distributed, open-source configuration, synchronization service along with naming registry for distributed applications. ZooKeeper stores a lot of shared information about [Kafka Consumers](#) and Kafka [Brokers](#).

2.2 Broker

A Kafka cluster is made up of multiple Kafka Brokers. Each Kafka Broker has a unique ID (number). Kafka Brokers contain topic log partitions. Connecting to one broker bootstraps a client to the entire Kafka cluster. For failover, you want to start with at least three to five brokers. A Kafka cluster can have, 10, 100, or 1,000 brokers in a cluster if needed.

2.3 Producer

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.

2.4 Consumer

Consumers read data from brokers. Consumers subscribes to one or more topics and consume published messages by pulling data from the brokers.

2.5 Topic

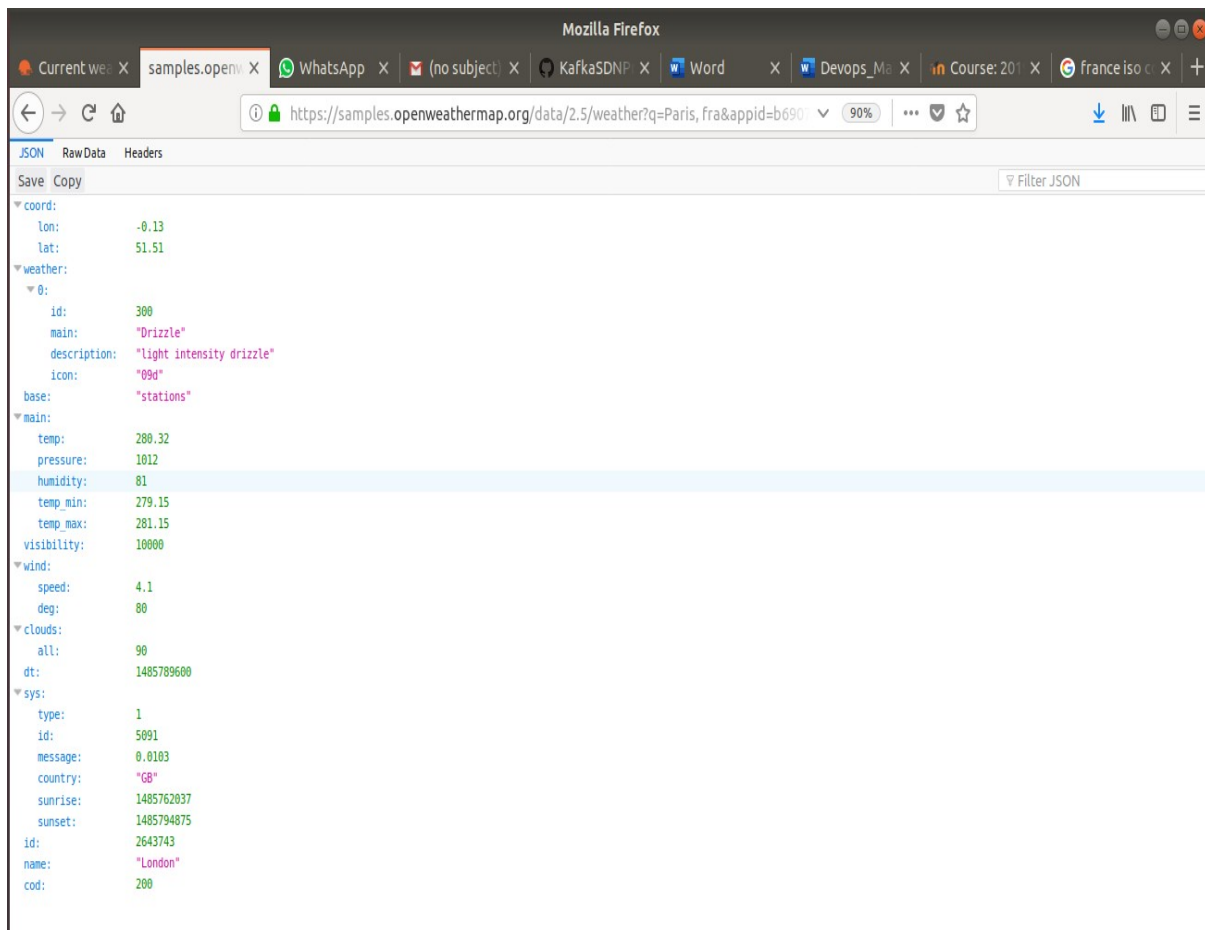
Kafka topic is a named stream of records. Kafka stores topics in logs. A topic log is broken up into partitions. Kafka spreads log's partitions across multiple servers or disks.

3.0 Data :

We are collecting weather data from <https://openweathermap.org/current#name>. Sample output from this website when we put London as a city is provided below.

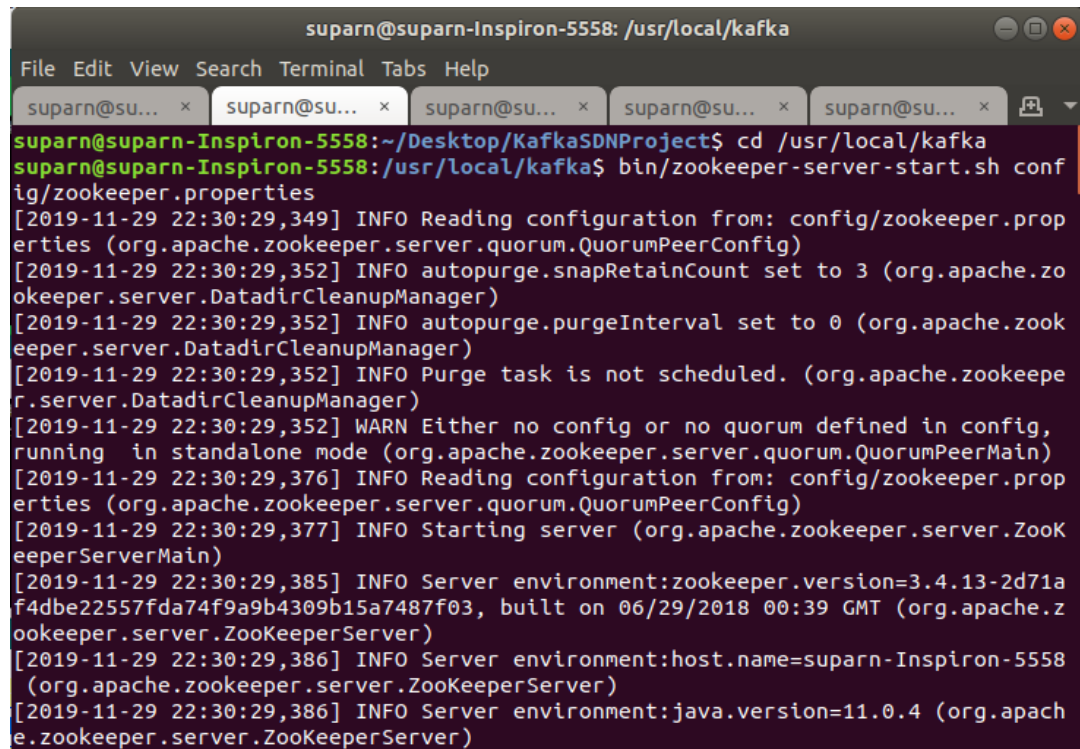
Link :-

<https://samples.openweathermap.org/data/2.5/weather?q=London,uk&appid=b6907d289e10d714a6e88b30761fae22>



4.0 Screenshots of Steps :

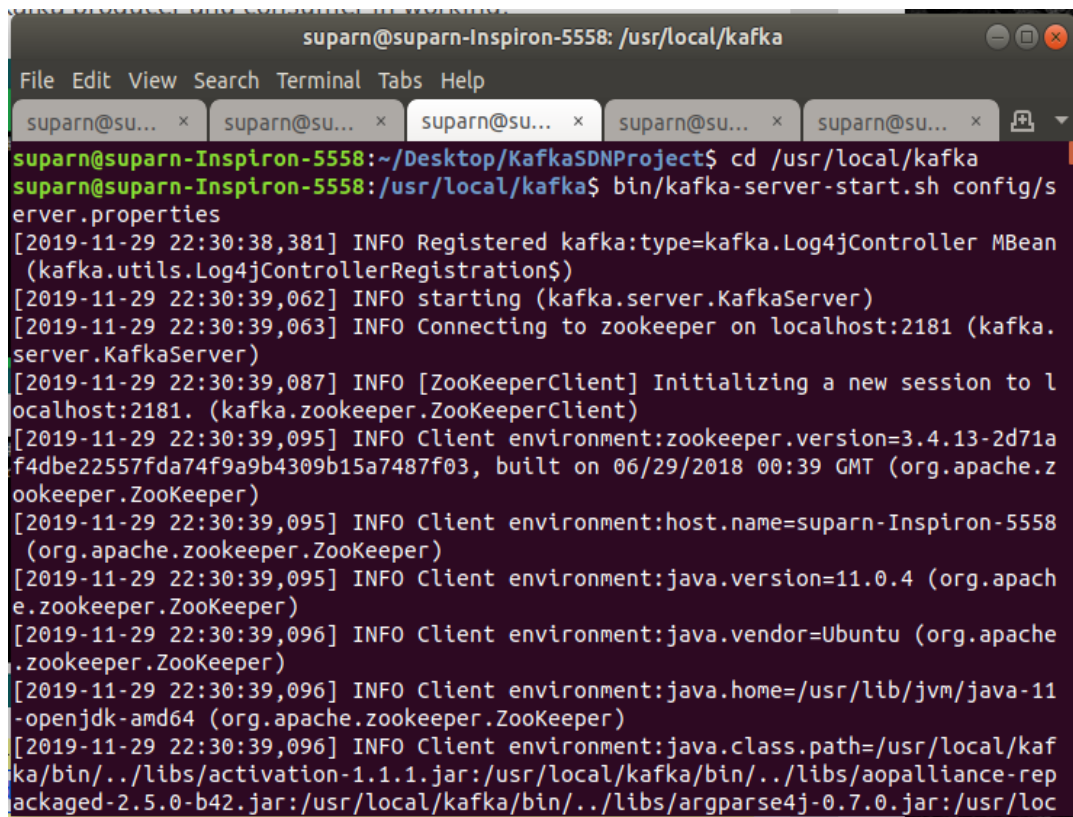
Zoo-Keeper Server



A terminal window titled 'suparn@suparn-Inspiron-5558: /usr/local/kafka' showing the execution of 'bin/zookeeper-server-start.sh config/zookeeper.properties'. The output displays various log messages including configuration reading, autopurge settings, and server startup information.

```
suparn@suparn-Inspiron-5558: /usr/local/kafka
File Edit View Search Terminal Tabs Help
suparn@su... x suparn@su... x suparn@su... x suparn@su... x suparn@su... x
suparn@suparn-Inspiron-5558:~/Desktop/KafkaSDNProject$ cd /usr/local/kafka
suparn@suparn-Inspiron-5558:/usr/local/kafka$ bin/zookeeper-server-start.sh config/zookeeper.properties
[2019-11-29 22:30:29,349] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2019-11-29 22:30:29,352] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DataDirCleanupManager)
[2019-11-29 22:30:29,352] INFO autopurge.purgeInterval set to 0 (org.apache.zookeeper.server.DataDirCleanupManager)
[2019-11-29 22:30:29,352] INFO Purge task is not scheduled. (org.apache.zookeeper.server.DataDirCleanupManager)
[2019-11-29 22:30:29,352] WARN Either no config or no quorum defined in config, running in standalone mode (org.apache.zookeeper.server.quorum.QuorumPeerMain)
[2019-11-29 22:30:29,376] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2019-11-29 22:30:29,377] INFO Starting server (org.apache.zookeeper.server.ZooKeeperServerMain)
[2019-11-29 22:30:29,385] INFO Server environment:zookeeper.version=3.4.13-2d71af4dbe22557fda74f9a9b4309b15a7487f03, built on 06/29/2018 00:39 GMT (org.apache.zookeeper.server.ZooKeeperServer)
[2019-11-29 22:30:29,386] INFO Server environment:host.name=suparn-Inspiron-5558 (org.apache.zookeeper.server.ZooKeeperServer)
[2019-11-29 22:30:29,386] INFO Server environment:java.version=11.0.4 (org.apache.zookeeper.server.ZooKeeperServer)
```

Kafka Server



A terminal window titled "suparn@suparn-Inspiron-5558: /usr/local/kafka" showing the execution of the Kafka server startup script. The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", "Tabs", and "Help". Below the menu bar are several tabs, each labeled "suparn@su...". The terminal output shows the user navigating to the Kafka directory and running the startup script, followed by a series of log messages indicating the server's initialization and connection to ZooKeeper.

```
suparn@suparn-Inspiron-5558:~/Desktop/KafkaSDNProject$ cd /usr/local/kafka
suparn@suparn-Inspiron-5558:/usr/local/kafka$ bin/kafka-server-start.sh config/s
erver.properties
[2019-11-29 22:30:38,381] INFO Registered kafka:type=kafka.Log4jController MBean
(kafka.utils.Log4jControllerRegistration$)
[2019-11-29 22:30:39,062] INFO starting (kafka.server.KafkaServer)
[2019-11-29 22:30:39,063] INFO Connecting to zookeeper on localhost:2181 (kafka.
server.KafkaServer)
[2019-11-29 22:30:39,087] INFO [ZooKeeperClient] Initializing a new session to l
ocalhost:2181. (kafka.zookeeper.ZooKeeperClient)
[2019-11-29 22:30:39,095] INFO Client environment:zookeeper.version=3.4.13-2d71a
f4dbe22557fda74f9a9b4309b15a7487f03, built on 06/29/2018 00:39 GMT (org.apache.z
ookeeper.ZooKeeper)
[2019-11-29 22:30:39,095] INFO Client environment:host.name=suparn-Inspiron-5558
(org.apache.zookeeper.ZooKeeper)
[2019-11-29 22:30:39,095] INFO Client environment:java.version=11.0.4 (org.apach
e.zookeeper.ZooKeeper)
[2019-11-29 22:30:39,096] INFO Client environment:java.vendor=Ubuntu (org.apache
.zookeeper.ZooKeeper)
[2019-11-29 22:30:39,096] INFO Client environment:java.home=/usr/lib/jvm/java-11
-openjdk-amd64 (org.apache.zookeeper.ZooKeeper)
[2019-11-29 22:30:39,096] INFO Client environment:java.class.path=/usr/local/kaf
ka/bin/../libs/activation-1.1.1.jar:/usr/local/kafka/bin/../libs/aopalliance-rep
ackaged-2.5.0-b42.jar:/usr/local/kafka/bin/../libs/argparse4j-0.7.0.jar:/usr/loc
```

Consumer Build

```
suparn@suparn-Inspiron-5558: ~/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master
File Edit View Search Terminal Tabs Help
suparn@suparn-Inspir... x suparn@suparn-Inspir... x root@suparn-Inspiron... x root@suparn-Inspiron... x suparn@suparn-Inspir... x suparn@suparn-Inspir... x
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.1:testCompile (default-testCompile) @ Kafka-Consumer ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master/target/test-classes
[INFO]
[INFO] --- maven-surefire-plugin:2.22.2:test (default-test) @ Kafka-Consumer ---
[INFO]
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 0, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- maven-jar-plugin:3.1.1:jar (default-jar) @ Kafka-Consumer ---
[INFO] Building jar: /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master/target/Kafka-Consumer-0.8.0.jar
[INFO]
[INFO] --- spring-boot-maven-plugin:2.2.1.RELEASE:repackage (repackage) @ Kafka-Consumer ---
[INFO] Replacing main artifact with repackaged archive
[INFO]
[INFO] --- maven-install-plugin:2.5.2:install (default-install) @ Kafka-Consumer ---
[INFO] Installing /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master/target/Kafka-Consumer-0.8.0.jar to /home/suparn/.m2/repository/com/kafka/consumer/Kafka-Consumer/0.8.0/Kafka-Consumer-0.8.0.jar
[INFO] Installing /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master/pom.xml to /home/suparn/.m2/repository/com/kafka/consumer/Kafka-Consumer/0.8.0/Kafka-Consumer-0.8.0.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 7.204 s
[INFO] Finished at: 2019-11-30T20:37:10+05:30
[INFO] -----
suparn@suparn-Inspiron-5558:~/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Consumer-master$
```

Producer Build

```
suparn@suparn-Inspiron-5558: ~/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Producer-master
File Edit View Search Terminal Tabs Help
suparn@suparn-Inspir... x suparn@suparn-Inspir... x root@suparn-Inspiron... x root@suparn-Inspiron... x suparn@suparn-Inspir... x suparn@suparn-Inspir... x
Downloading from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.1/junit-3.8.1.pom
Downloaded from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.1/junit-3.8.1.pom (998 B at 1.5 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/commons-codec/commons-codec/1.6/commons-codec-1.6.pom
Downloaded from central: https://repo.maven.apache.org/maven2/commons-codec/commons-codec/1.6/commons-codec-1.6.pom (11 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-parent/22/commons-parent-22.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-parent/22/commons-parent-22.pom (42 kB at 23 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-shared-utils/0.4/maven-shared-utils-0.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-shared-utils/0.4/maven-shared-utils-0.4.pom (4.0 kB at 4.4 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0.15/plexus-utils-3.0.15.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0.15/plexus-utils-3.0.15.pom (3.1 kB at 5.2 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.1/junit-3.8.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/classworlds/classworlds/1.1-alpha-2/classworlds-1.1-alpha-2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-shared-utils/0.4/maven-shared-utils-0.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/commons-codec/commons-codec/1.6/commons-codec-1.6.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0.15/plexus-utils-3.0.15.jar
Downloaded from central: https://repo.maven.apache.org/maven2/classworlds/classworlds/1.1-alpha-2/classworlds-1.1-alpha-2.jar (38 kB at 43 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-shared-utils/0.4/maven-shared-utils-0.4.jar (155 kB at 99 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/commons-codec/commons-codec/1.6/commons-codec-1.6.jar (233 kB at 128 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/junit/junit/3.8.1/junit-3.8.1.jar (121 kB at 56 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0.15/plexus-utils-3.0.15.jar (239 kB at 60 kB/s)
[INFO] Installing /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Producer-master/target/Kafka-Producer-0.8.0.jar to /home/suparn/.m2/repository/com/kafka/producer/Kafka-Producer/0.8.0/Kafka-Producer-0.8.0.jar
[INFO] Installing /home/suparn/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Producer-master/pom.xml to /home/suparn/.m2/repository/com/kafka/producer/Kafka-Producer/0.8.0/Kafka-Producer-0.8.0.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 08:01 min
[INFO] Finished at: 2019-11-30T20:34:00+05:30
[INFO] -----
suparn@suparn-Inspiron-5558:~/Desktop/SDN-PROJECT/Spring-Boot-Kafka-Producer-master$
```


Consumer 1

```
2019-11-30 20:51:31,796 :: 307114 [org.springframework.kafka.KafkaListenerEndpointContainer#2-0-C-1] DEBUG c.k.consumer.listener.KafkaConsumer
- consumeOne(1) :: Consumed JSON Message:{ Broker Partition : 0 | Message Key : 38 | Message : Message(msgKey=38, msgVal={"coord":{"lon":-0.13,"lat":51.51},"weather":[{"id":300,"main":"Drizzle","description":"light intensity drizzle","icon":"09d"}],"base":"stations","main":{"temp":280.32,"pressure":1012,"humidity":81,"temp_min":279.15,"temp_max":281.15},"visibility":10000,"wind":{"speed":4.1,"deg":80},"clouds":{"all":90},"dt":1485789600,"sys":{"type":1,"id":5091,"message":0.0103,"country":"GB","sunrise":1485762037,"sunset":1485794875},"id":2643743,"name":"London","cod":200}) }
```

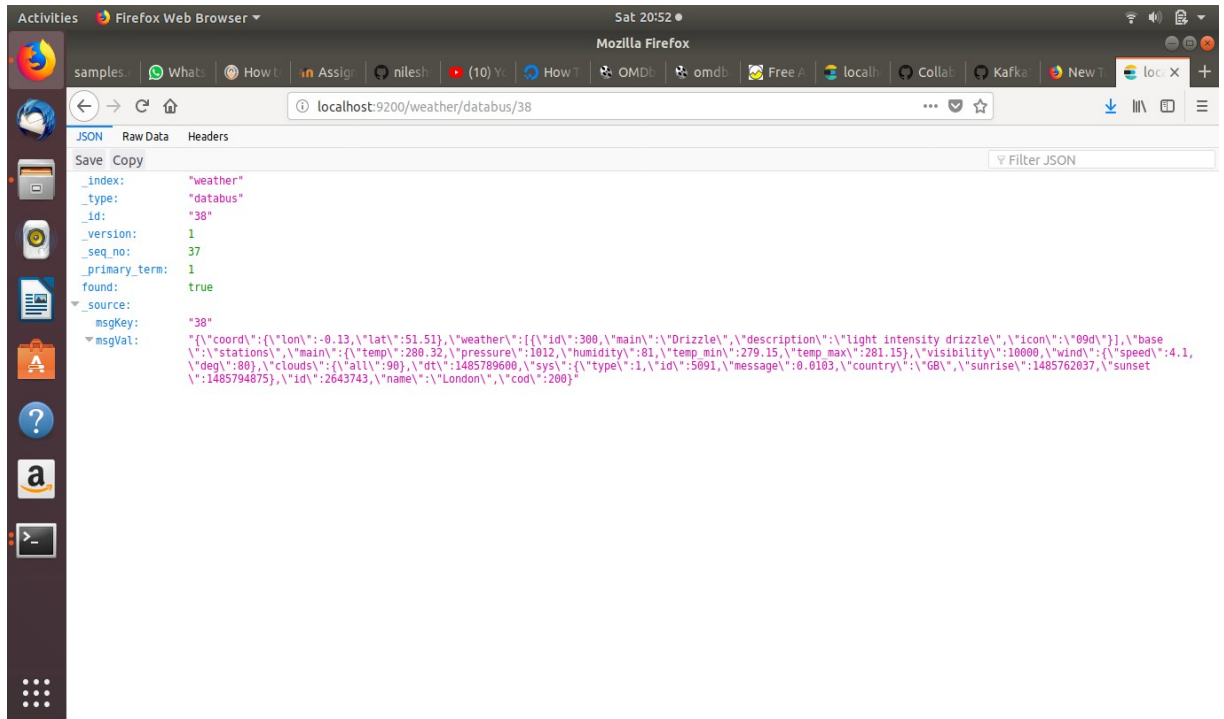
Consumer 2

```
- consumeTwo(2) :: Consumed JSON Message:{ Broker Partition : 0 | Message Key : 1 | Message : Message(msgKey=1, msgVal={"coord":{"lon":-0.13,"lat":51.51},"weather":[{"id":300,"main":"Drizzle","description":"light intensity drizzle","icon":"09d"}],"base":"stations","main":{"temp":280.32,"pressure":1012,"humidity":81,"temp_min":279.15,"temp_max":281.15},"visibility":10000,"wind":{"speed":4.1,"deg":80},"clouds":{"all":90},"dt":1485789600,"sys":{"type":1,"id":5091,"message":0.0103,"country":"GB","sunrise":1485762037,"sunset":1485794875},"id":2643743,"name":"London","cod":200}) } Send To Remote Application.
```

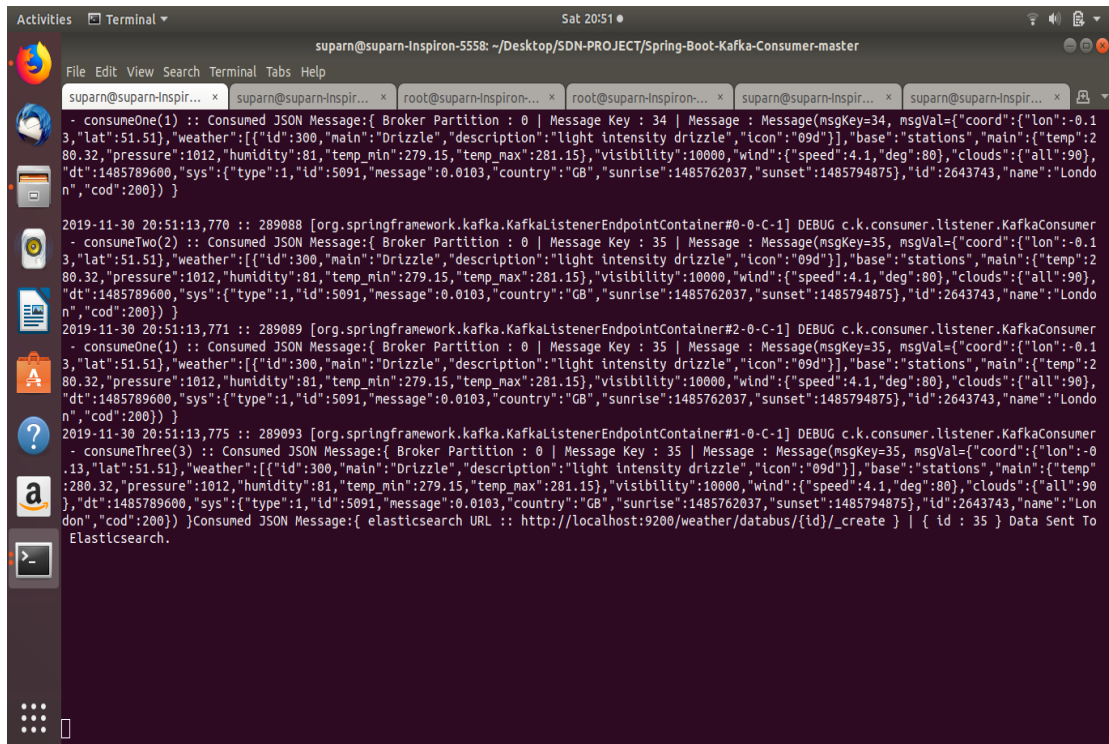
Consumer 3

```
2019-11-30 20:51:31,802 :: 307120 [org.springframework.kafka.KafkaListenerEndpointContainer#1-0-C-1] DEBUG c.k.consumer.listener.KafkaConsumer
- consumeThree(3) :: Consumed JSON Message:{ Broker Partition : 0 | Message Key : 38 | Message : Message(msgKey=38, msgVal={"coord":{"lon":-0.13,"lat":51.51},"weather":[{"id":300,"main":"Drizzle","description":"light intensity drizzle","icon":"09d"}],"base":"stations","main":{"temp":280.32,"pressure":1012,"humidity":81,"temp_min":279.15,"temp_max":281.15},"visibility":10000,"wind":{"speed":4.1,"deg":80},"clouds":{"all":90},"dt":1485789600,"sys":{"type":1,"id":5091,"message":0.0103,"country":"GB","sunrise":1485762037,"sunset":1485794875},"id":2643743,"name":"London","cod":200}) }Consumed JSON Message:{ elasticsearch URL :: http://localhost:9200/weather/databus/{id}/_create } | { id : 38 } Data Sent To Elasticsearch.
```

Elastic Search Sent Data at id 38



Combined Screenshot of all consumers



5.0 Conclusion

From this project we learn about the kafka and its multitasking ability with High efficiency. Joining kafka with SDN reduces the load from the control plane by distributing the load to the multiple consumers.

6.0 Git Hub Link Of the Repository

<https://github.com/suparnlele/SDN-Kafka-Project>

7.0 References:

1. <https://www.digitalocean.com/community/tutorials/how-to-install-elasticsearch-on-an-ubuntu-vps> – Elastic search installation
2. <https://tecadmin.net/install-apache-kafka-ubuntu/> - Kafka Installation
3. <https://samples.openweathermap.org> – Weather Data

