Installation of Apache Kafka

Step 1: Search Apache Kafka in your browser or https://kafka.apache.org/
Step 2: Go to Get started and then to Quickstart.
Step 3: Click the Download link and then click on the first link that appears on the browser.
Step 4: A zip file will get downloaded so extract the folder.
Step 5: Open the command prompt in the folder where both bin and config is present.
Step 6: Start the zookeeper using command prompt.
Step 7: Start the server using command prompt.

Problems Faced:

- When starting the zookeeper you might come across a problem which says command line too long.

  Solution: When kafka folder is extracted then there will be two folders of Kafka made one within the other so make it one folder by copying and pasting.

- When starting the zookeeper you might find the error zookeeper is disabled.

  Solution: Install the latest version of jdk.

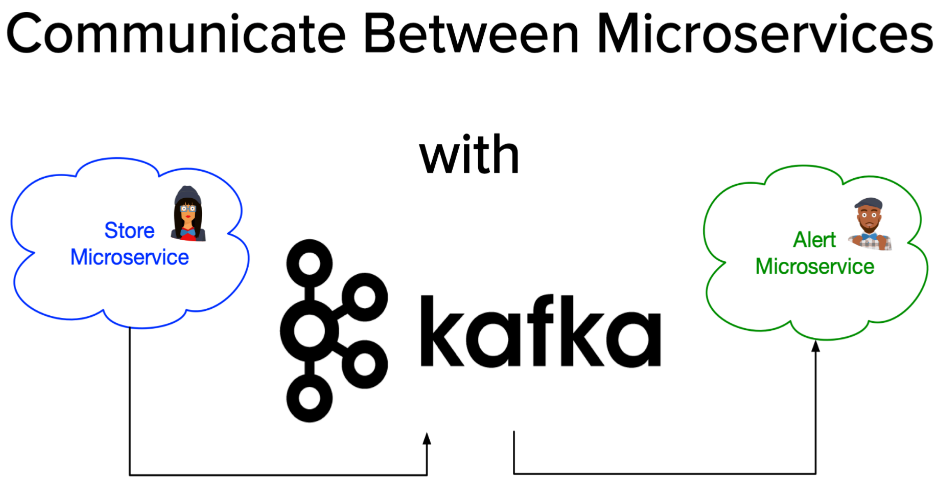
- When starting the zookeeper you might face the error path not found.

  Solution: Go to the environment variables and set the kafka path and the jdk path.

KAFKA

It is like a communication system that helps different parts of a computer system exchange data by publishing and subscribing to topics.

How communication is done?

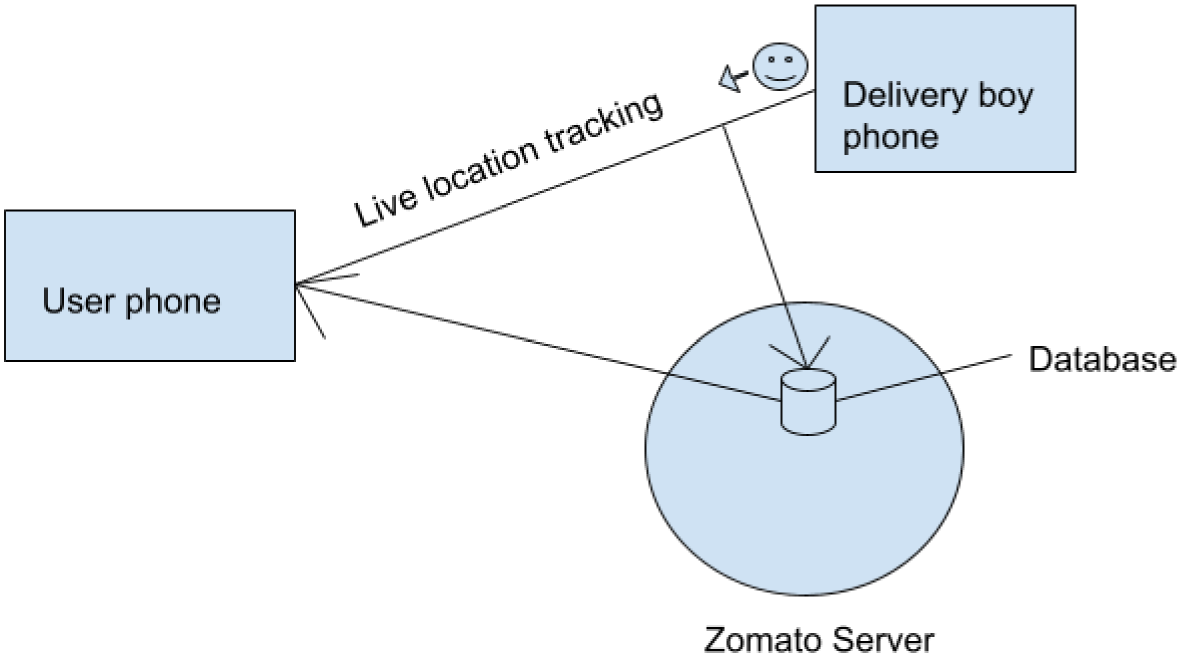


Assuming Store Microservice as producer who will publish or write messages in Kafka and Alert Microservice as consumer who has to subscribe the topic so as to consume.
Multiple consumers can subscribe the topic.

Example:

- OLA Driver location update.
- Zomato live food tracking.
- Notification system to huge user.

Problem (Zomato live food tracking)



Assume:

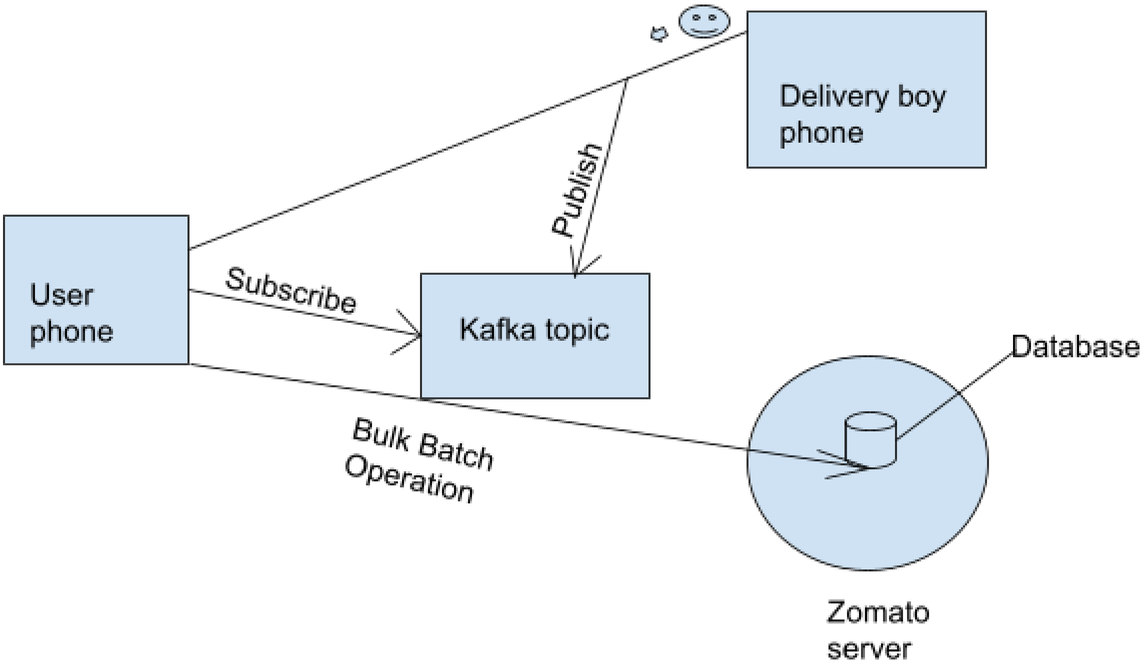
- User phone
- Delivery boy phone
- Zomato server

If the delivery boy moves then its live location should get tracked when a man reaches a certain distance and the data gets stored in the database and send it to the user, when the location gets updated then it will again get updated then will get stored in the database and then again send it to the user. This can be done for the nth time.

But there are many user sso the database gets hit many times. As the throughput of the database is not more(i.e., we can perform less read and write operations) and database gets crashed.

This problem is solved by Apache Kafka.

Solution

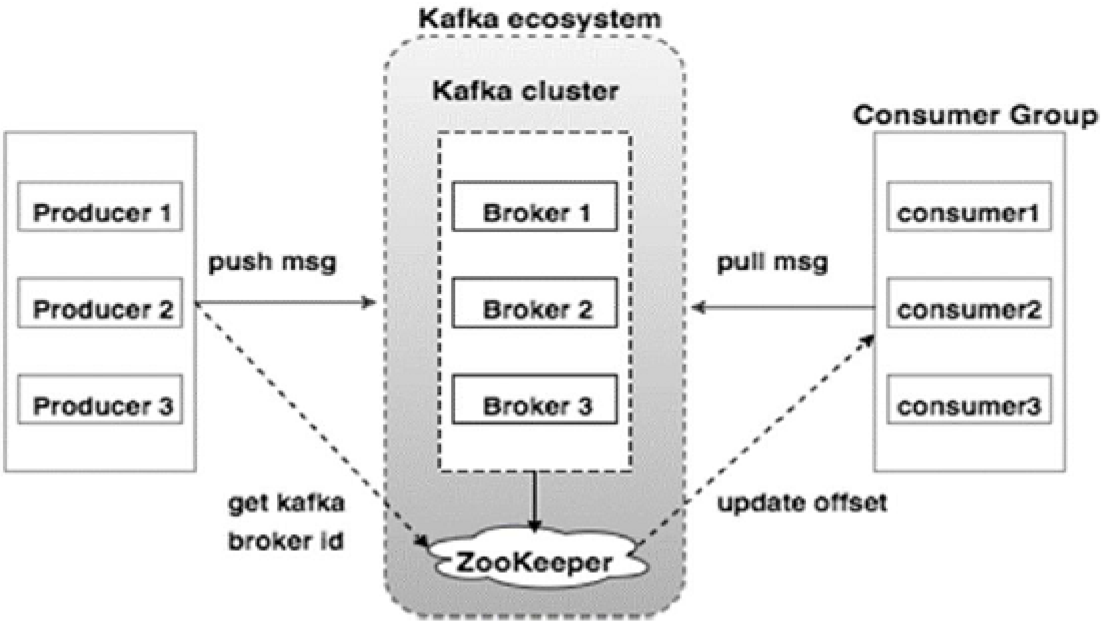


The user will subscribe the topic of Kafka and when the delivery boy moves the location gets tracked and will get published. Now, when all the data gets collected then that bulk data will be get stored in the database.Since Kafka has very high throughput so millions of read and write operations can be performed in less time.

Benefits

- High Throughput: It means millions of records can be read or write in few seconds.
- Fault Tolerance (Replication): If any error comes then we will get our data because Kafka follows replication technique i.e., the copy of the data stored in Kafka will be stored in multiple nodes.
- Durable: Data will not vanish easily.
- Scalable: Since the throughput is more so we can scale it easily.

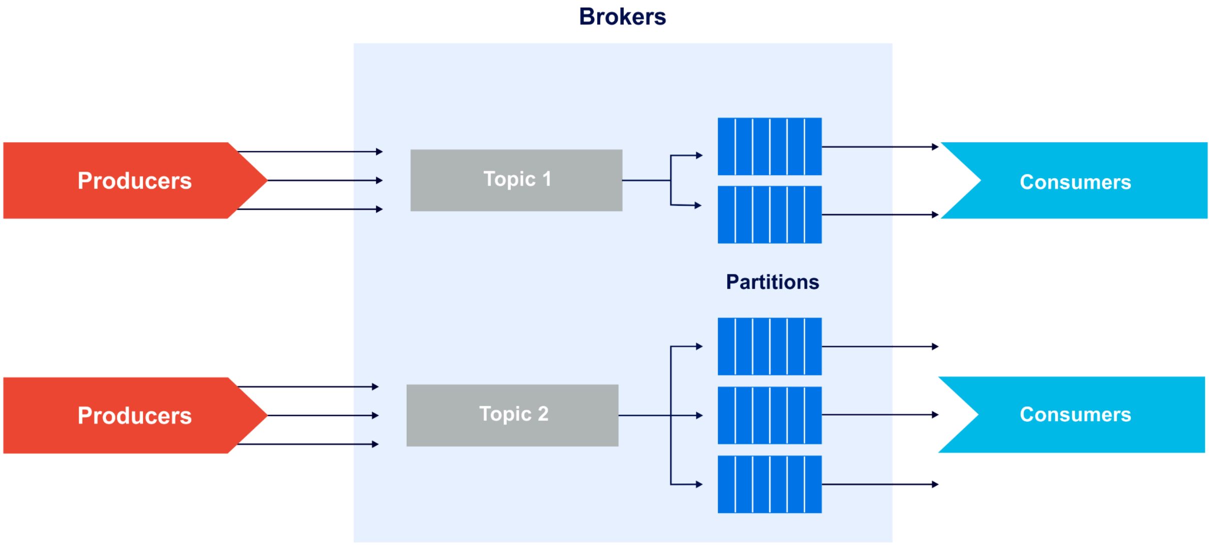
Kafka Architecture



Producer: produces the data (sends the messages)
Consumer: consumes the data
Kafka Ecosystem:

- Zookeeper- helps Kafka to manage the state of the broker
- Kafka Cluster- software which manages Kafka brokers/ servers

There can be multiple brokers/ systems because it is a distributed system.



- Brokers will contain topics to store the data.
- Topics are used to categorize the data.
- Topics contains partitions(used to store data).

- The partitions contains offset values and when the data comes the offset value will keep on increasing and that will keep on getting stored.

NOTE

- Default Zookeeper port no.: 2181
- Default Kafka port no.: 9092

Kafka commands:

To start the zookeeper

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

To start the server

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

To create new topic

-
  ```
  bin\windows\kafka-topics.bat --create --topic user-topic
  --bootstrap-server localhost:9092
  ```

To produce the message

-
  ```
  bin\windows\kafka-console-producer.bat --topic user-topic
  --bootstrap-server localhost:9092
  ```

  NOTE: Ctrl+C to stop

To consume the message

-
  ```
  bin\windows\kafka-console-consumer.bat --topic user-topic
  —-from-beginning --bootstrap-server localhost:9092
  ```

To list existing topics

-
  ```
  bin\windows\kafka-topics.bat --bootstrap-server
  localhost:9092 --list
  ```

To describe a topic

- `bin\windows\kafka-topics.bat --bootstrap-server localhost:9092  --describe --topic mytopic`

To purge a topic

- `bin\windows\kafka-topics.bat --bootstrap-server localhost:9092  --alter --topic mytopic --config retention.ms=1000`

Wait a minute

- `bin\windows\kafka-topics.bat --bootstrap-server localhost:9092  --alter --topic mytopic --delete-config retention.ms`

To delete a topic

- `bin\windows\kafka-topics.bat --bootstrap-server localhost:9092  --delete --topic mytopic`

To get number of messages in a topic

- `bin\windows\kafka-run-class.bat kafka.tools.GetOffsetShell --broker-list localhost:9092 --topic mytopic --time -1 --offsets 1 | awk -F ":" '{sum += $3} END {print sum}'`

To get the earliest offset still in a topic

- `bin\windows\kafka-run-class.bat kafka.tools.GetOffsetShell --broker-list localhost:9092 --topic mytopic --time -2`

To get the latest offset still in a topic

- `bin\windows\kafka-run-class.bat kafka.tools.GetOffsetShell --broker-list localhost:9092 --topic mytopic --time -1`

To consume messages with the console consumer

- `bin\windows\kafka-console-consumer.bat --new-consumer --bootstrap-server localhost:9092 --topic mytopic --from-beginning`

To get the consumer offsets for a topic

- `bin\windows\kafka-consumer-offset-checker.bat --zookeeper=localhost:2181 --topic=mytopic --group=my_consumer_group`

To read from consumer offsets

Add the following property to `config/consumer.properties`: `exclude.internal.topics=false`

- `bin\windows\kafka-console-consumer.bat --consumer.config config/consumer.properties --from-beginning --topic __consumer_offsets --bootstrap-server localhost:9092 --formatter "kafka.coordinator.GroupMetadataManager\$OffsetsMessageFormatter"`

Kafka Consumer Groups

To list the consumer groups known to Kafka

- `bin\windows\kafka-consumer-groups.bat --bootstrap-server localhost:9092  --list` (old api)

- `bin\windows\kafka-consumer-groups.bat --new-consumer --bootstrap-server localhost:9092 --list` (new api)

To view the details of a consumer group

- `bin\windows\kafka-consumer-groups.bat --bootstrap-server localhost:9092 --describe --group <group name>`

Getting the last five messages of a topic

- `kafkacat -C -b localhost:9092 -t mytopic -p 0 -o -5 -e`

Starting the zookeeper shell

- `bin\windows\zookeeper-shell.bat localhost:2181`