Messaging system :

Messaging system is to transfer the message from one application to another application. Generally to achieve this messaging system use MOM :Message Oriented Middlware. Message can be any type like byte format, stream format, json format, object format etc.

This messaging concept or MOM we use after web service.

Client (End User) Server application

Client send the

End user they call spring boot application or they can call front end technologies and front end technologies call backend technologies to get the data

If we want to share the data between two application. But in web application two application can communicate with each other but both application must be web application.

Here both the application doesn’t hold the data. Those application must be interact with file system or database system to store the data. Here data is persistence

But some time we want to share of any type of data, data is very huge and those data we don’t want to store permanently. Those application not mandatory web application they can be any type of application.

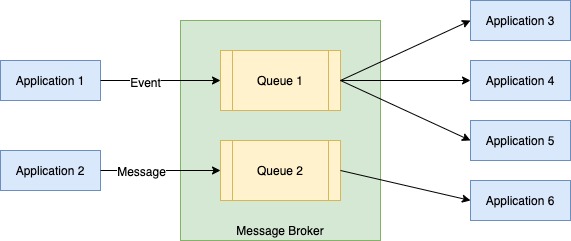
application MOM application

the application which send message to MOM is known as producer and application which receive from message from MOM is known as consumer. These application can be any type of application.

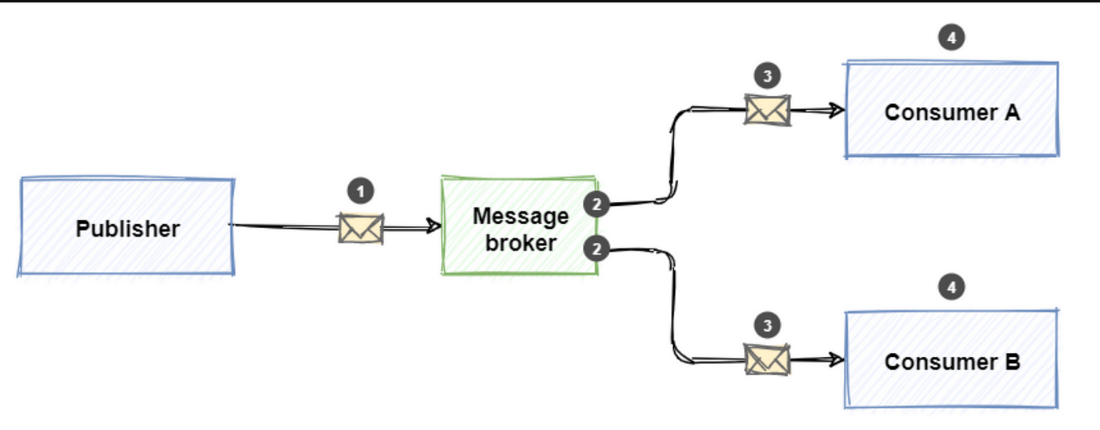
To achieve these messaging concept we use two type

1. P To P communication Point to Point communication
2. Pub and Sub communication Publisher and Subscriber

In P2P communication we use queue as container to hold the message. In P2P communication only one consumer can receive the message. 1 to 1.



In Pub/Sub communication we use topic as container to hold the message. Many consumer can receive same message at same time if they are connect to MOM. 1 to many communication.



To implement messaging service

MQ : Message Queue : Product

ActiveMQ

RabitMQ

SonicMQ

Apache Kafka

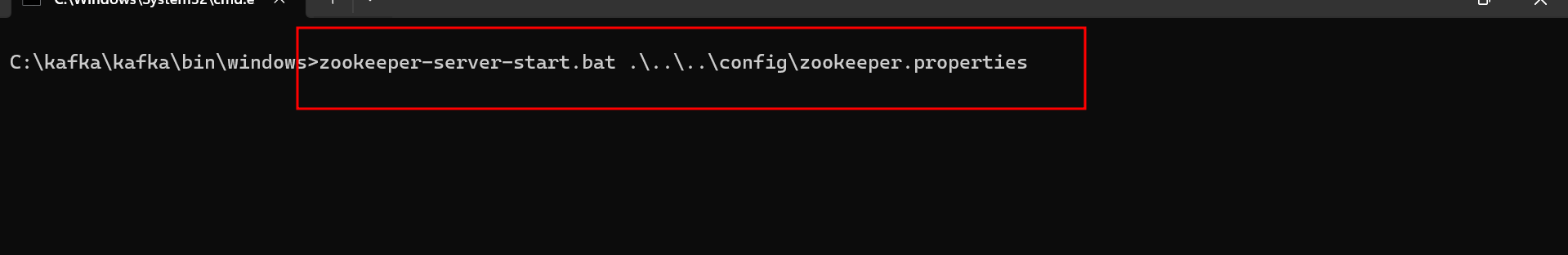
Apache Kafka : Apache Kafka is an open source software platform develop by Apache organization. Apache Kafka internal logic written using Java and Scala language. Apache Kafka mainly use to distribute message using pub and sub. In Apache Kafka data is very huge, Zero fault tolerance. Apache Kafka internally use stream concept to share huge data between one application to another application. Apache Kafka share the data using asynchronous concept.



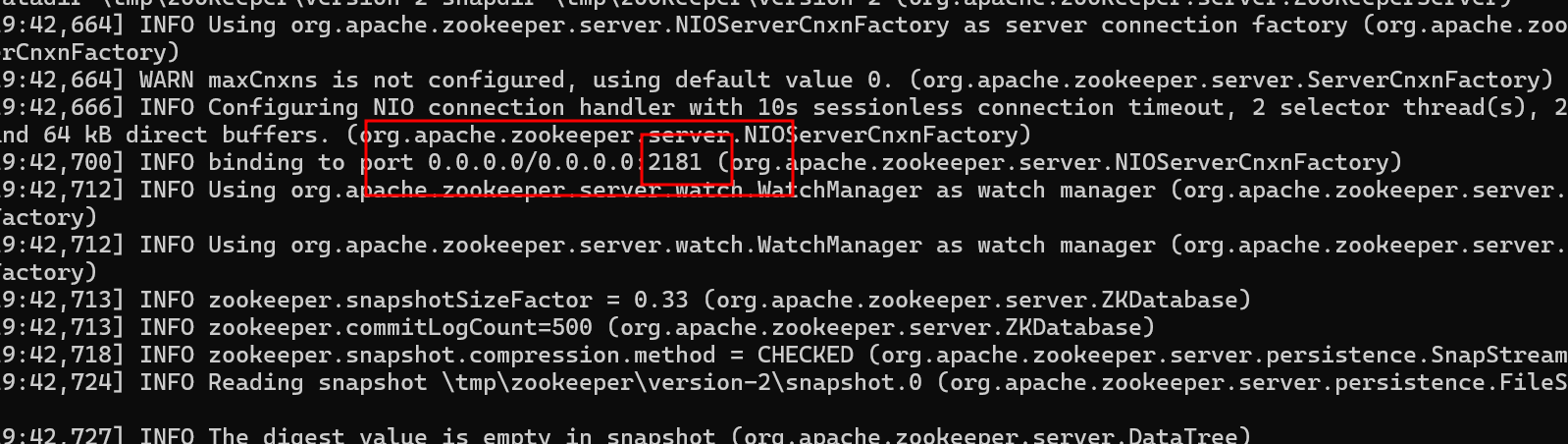
First we need Apache Kafka Software

First we need start zookeeper

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

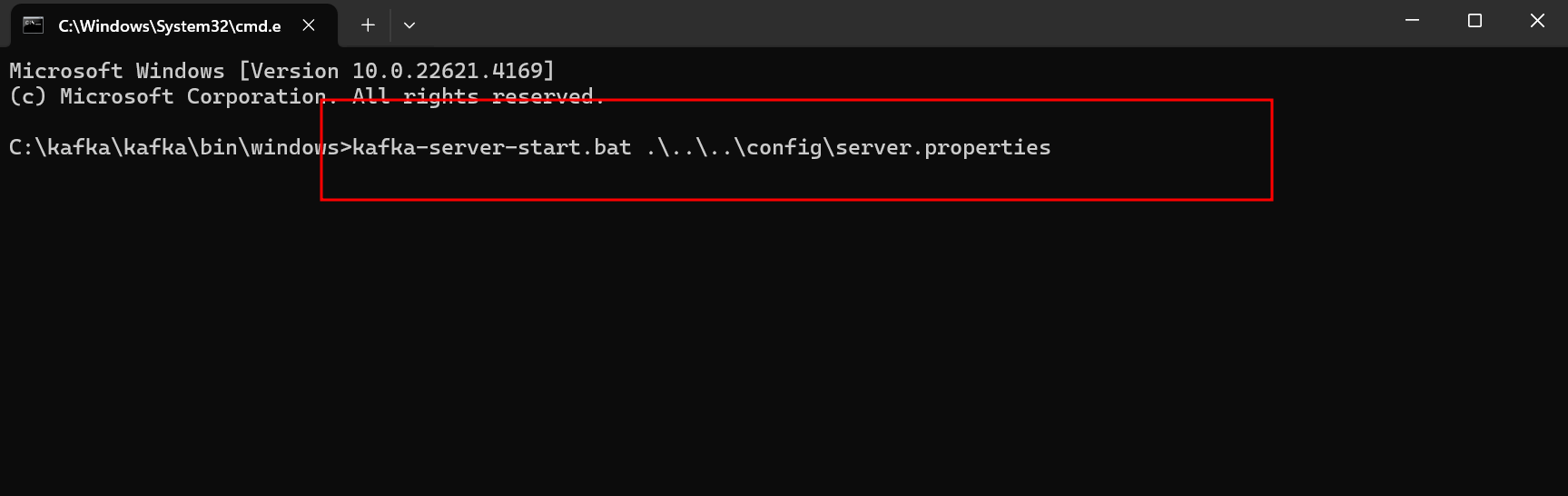


By default zookeeper port number is 2181

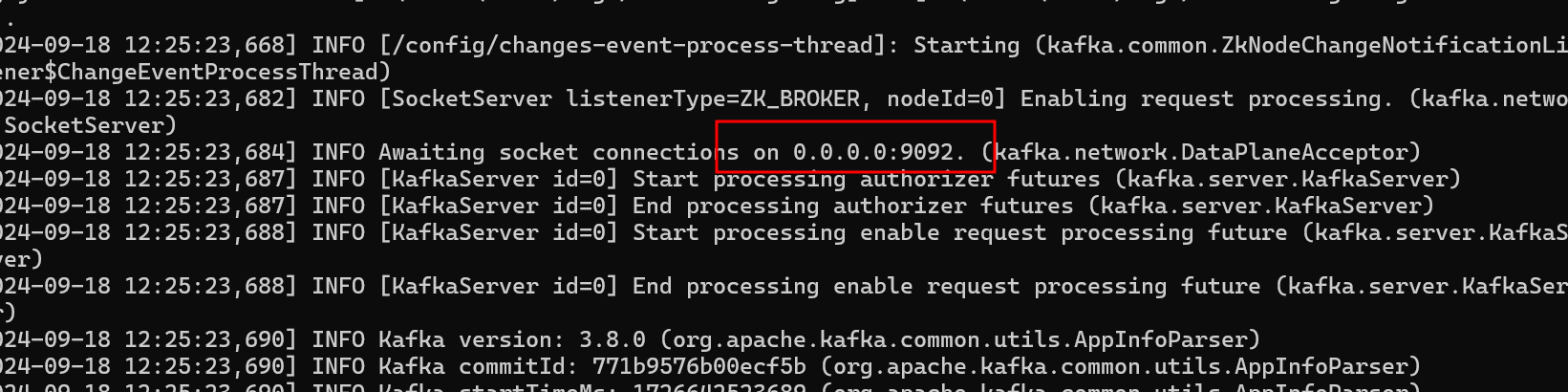


Now we need to start Apache Kafka Server

**kafka-server-start.bat .\..\..\config\server.properties**



Apache Kafka default port number is 9092



Producer : A producer is a program using any language which connect to kafak server using hostname and port number. It is use to send the message of any types.

Consumer : a consumer is a program using any language which connect to kafka server using hostname and port number. It is use to consume the message of any types.

Topics : topic is a type of container which hold the message. Topic can split the message into n number of partitions for scalability.

Broker : A Apache Kafka broker is a type of server that store the data using topic and it serve to client ie producer and consumer.

Zookeeper : Zookeeper is a another type of service for managing Apache Kafka broker and its meta data information like partition information etc.

We will interact with Apache Kafka using command prompt(Admin).

To verify how topic names present in broker.

kafka-topics.bat --bootstrap-server localhost:9092 –list

Now we will create new topics

kafka-topics.bat --bootstrap-server localhost:9092 --create --topic topic1

to check topic property

kafka-topics.bat --bootstrap-server localhost:9092 --describe --topic topic1

send the message to topic using console

kafka-console-producer.bat --bootstrap-server localhost:9092 --topic topic1

it will ask us the message we write the message if you wan to exists or terminal cntr + C/D

if we want to get the message from Kafka topic we need run the below command

kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic topic1

if any producer produce the message we will get else no.

if you need the message from beginning from that topic

kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --topic topic1

One producer and many consumer get the same message from Apache Kafka topic

