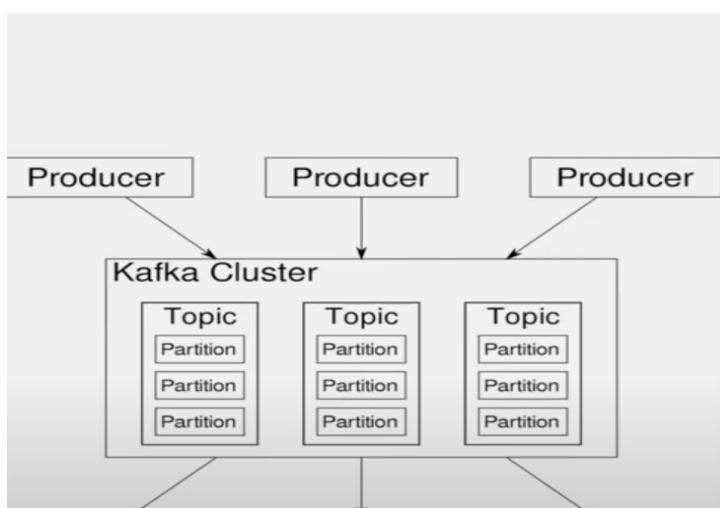
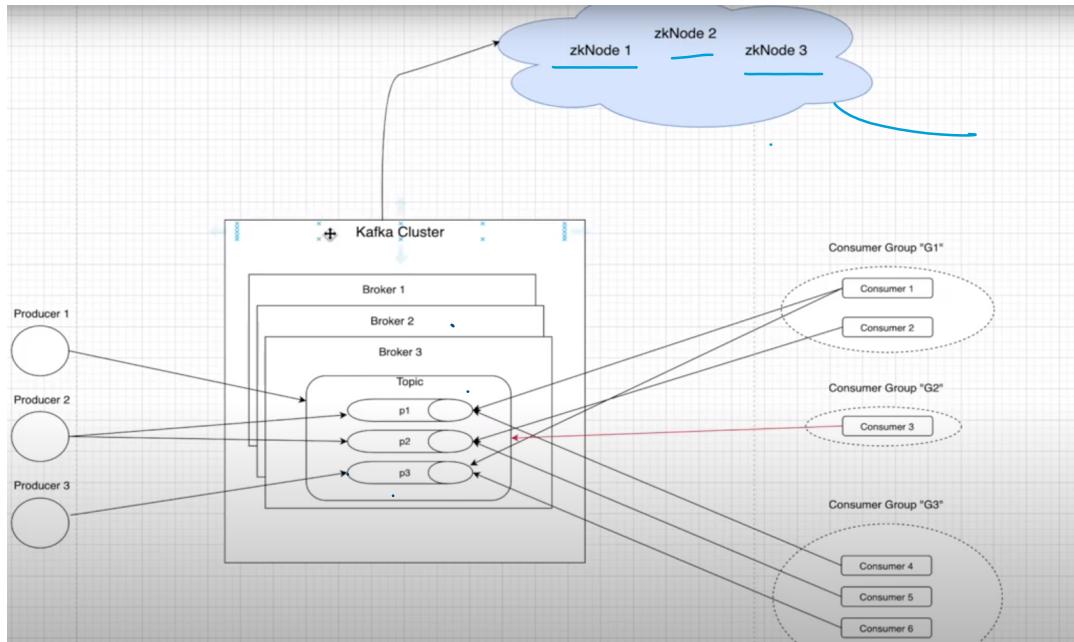
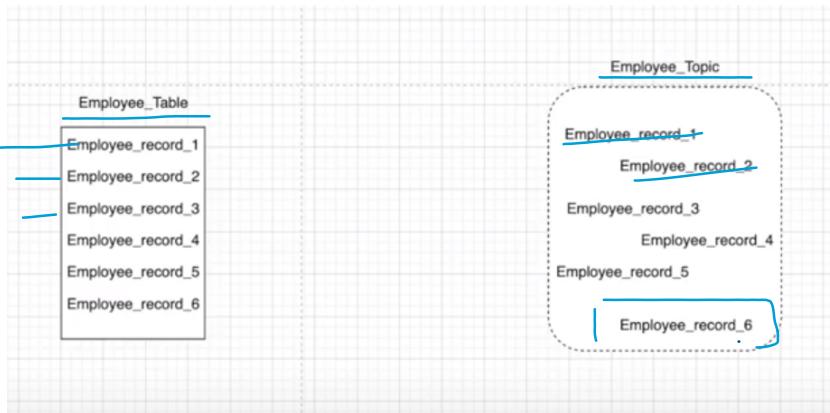
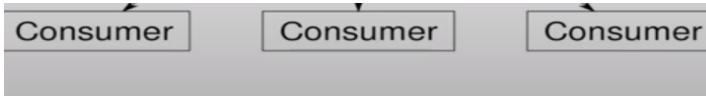


## Kafka Architecture

06:51 PM



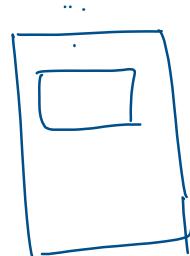


## PARTITIONS

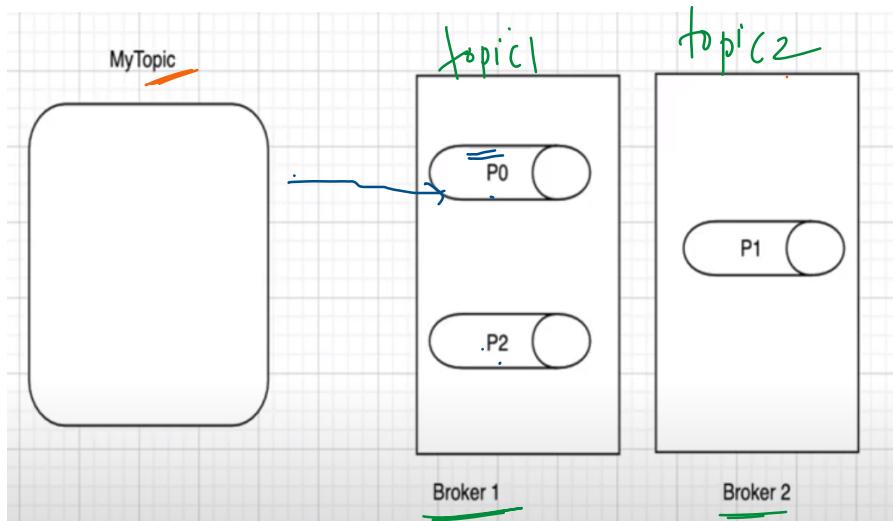
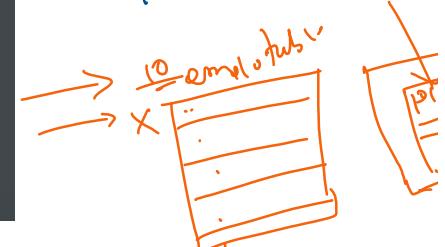
Topics are split in Partitions.



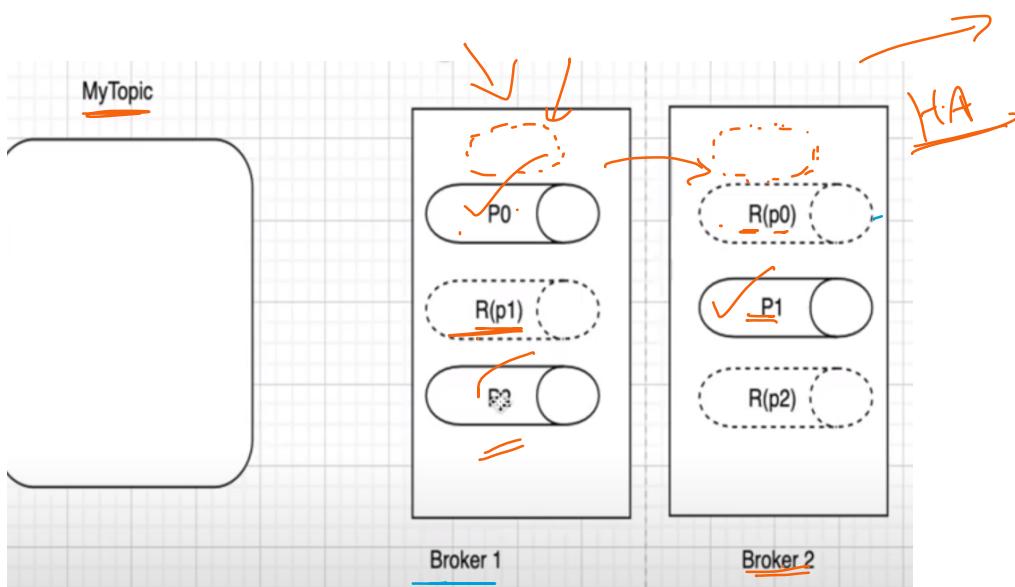
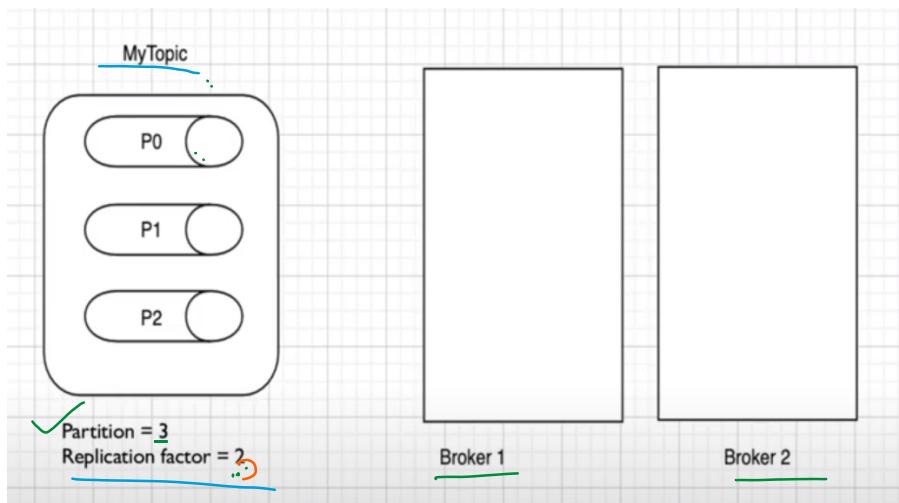
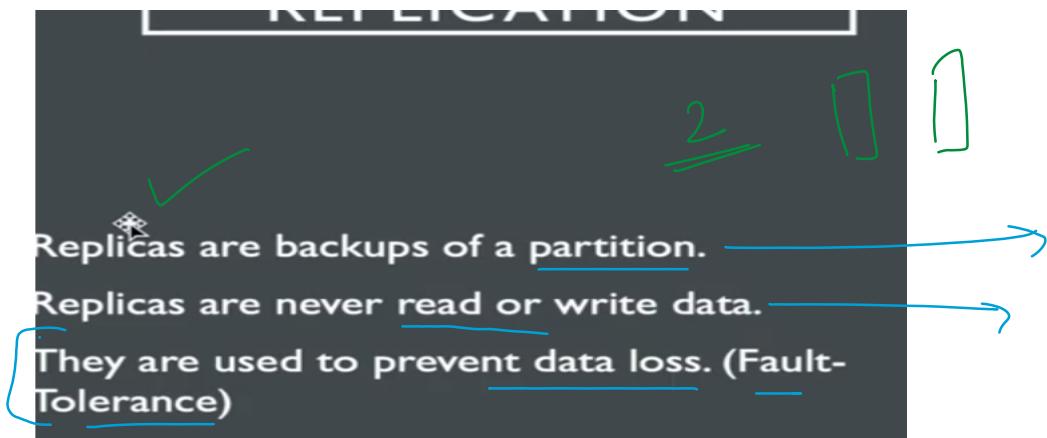
All the messages within a partition are ordered and immutable.



Each message within a partition has a unique Id associated known as **Offset**.

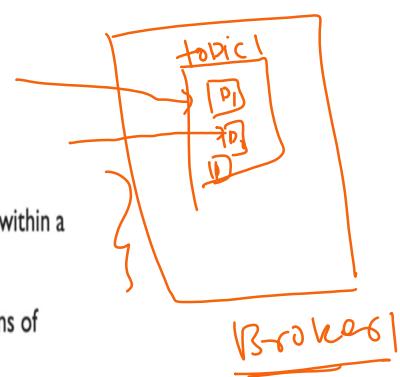


## REPLICA & REPLICATION



Producers are applications which write/publish data to the topics within a cluster using the Producing APIs.

Producers can write data either on the topic level (All the partitions of that topic) or specific partitions of the topic.



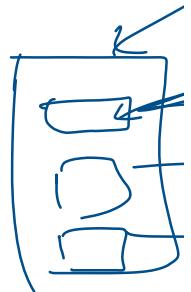


Consumers are applications which read/consume data from the topics within a cluster using the Consuming APIs. →

Consumers can read data either on the topic level (All the partitions of that topic) or specific partitions of the topic.

Consumers are always associated with exactly one **Consumer Group**.

A Consumer Group is a group of related consumers that perform a task.



Brokers are simple software processes who maintain and manage the published messages.

Also known as kafka servers.

Brokers also manage the **consumer-offsets** and are responsible for the delivery of messages to the right consumers.

A Set of brokers who are communicating with each other to perform the management and maintenance task are collectively known as **Kafka Cluster**.

We can add more brokers in a already running kafka cluster without any downtime.



Zookeeper is used to monitor kafka Cluster and co-ordinate with each broker.

Keeps all the metadata information related to kafka cluster in the form of a key-value pair.

Metadata includes

1. Configuration information.

2. Health status of each broker.

It is used for the controller election within kafka cluster.

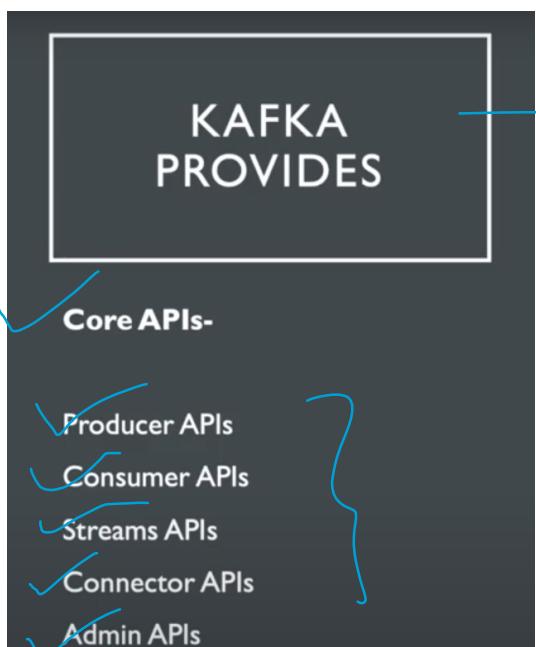
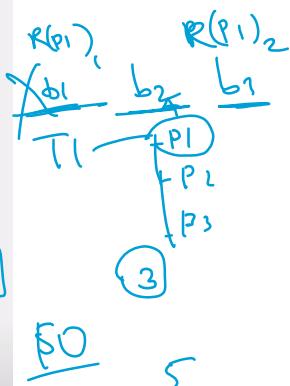
A Set of Zookeepers nodes working together to manage other distributed systems is known as **Zookeeper Cluster** or "**Zookeeper Ensemble**".



Apache Zookeeper is a software developed by Apache which acts as a centralized service used to maintain naming and configuration data and to provide synchronization within distributed systems. Zookeeper keeps track of the status of the Kafka cluster nodes and it also keeps track of Kafka topics, partitions, etc.

## ✓ KAFKA FEATURES

1. **Scalable.**  
Horizontal Scaling is done by adding new brokers to the existing clusters.
2. **Fault Tolerance.**  
Kafka clusters can handle failures because of its distributed nature.
3. **Durable.**  
Kafka uses "Distributed commit log" which means messages persists on disk as fast as possible.
4. **Performance**  
Kafka has **high throughput** for both publishing and subscribing messages.
5. **No Data loss**  
It ensures no data loss if we configure it properly.
6. **Zero Down Time**  
It ensures zero downtime when required number of brokers are present in the cluster.
7. **Reliability**  
Kafka is reliable because it provides above features.



, B.T.S.B

SB

