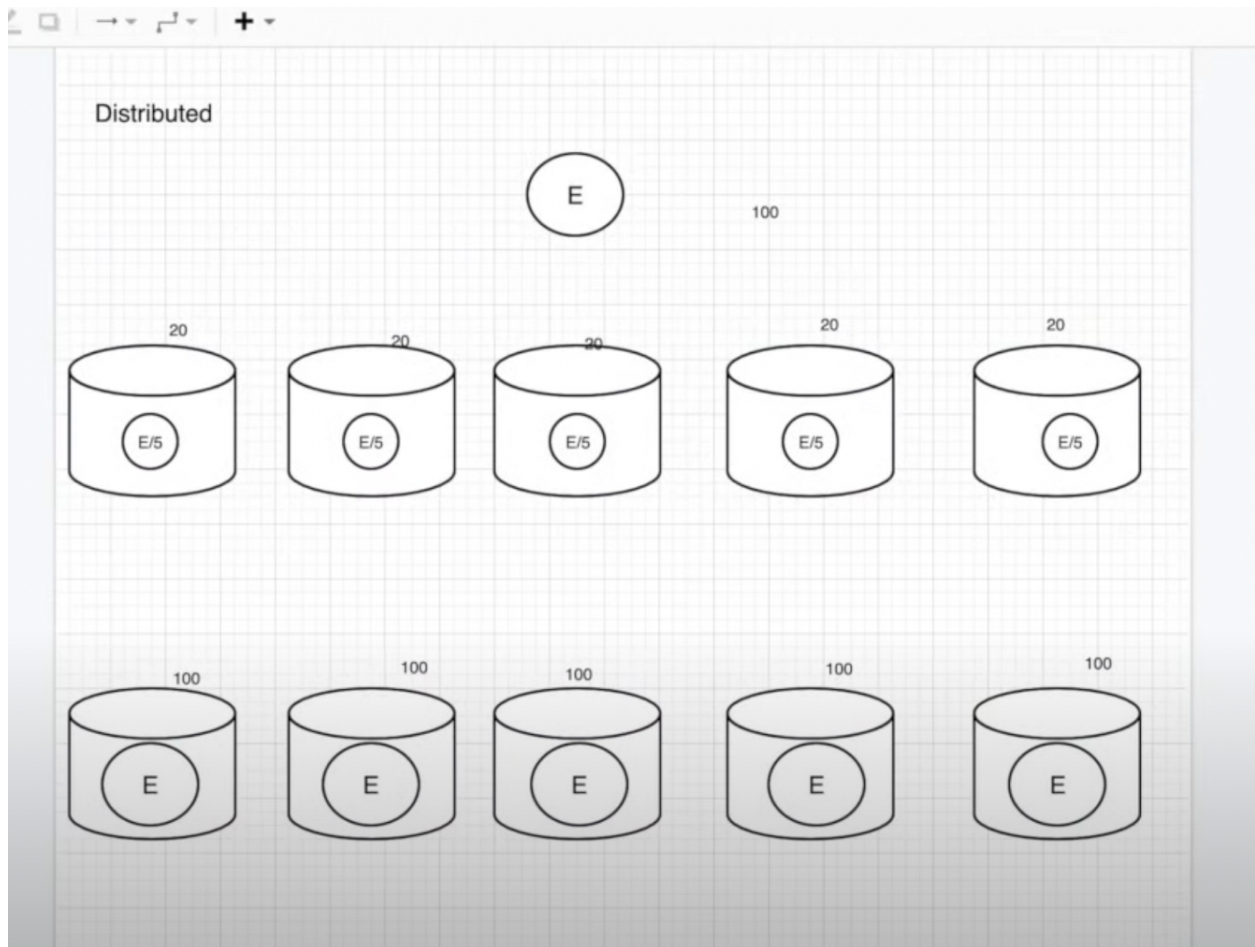


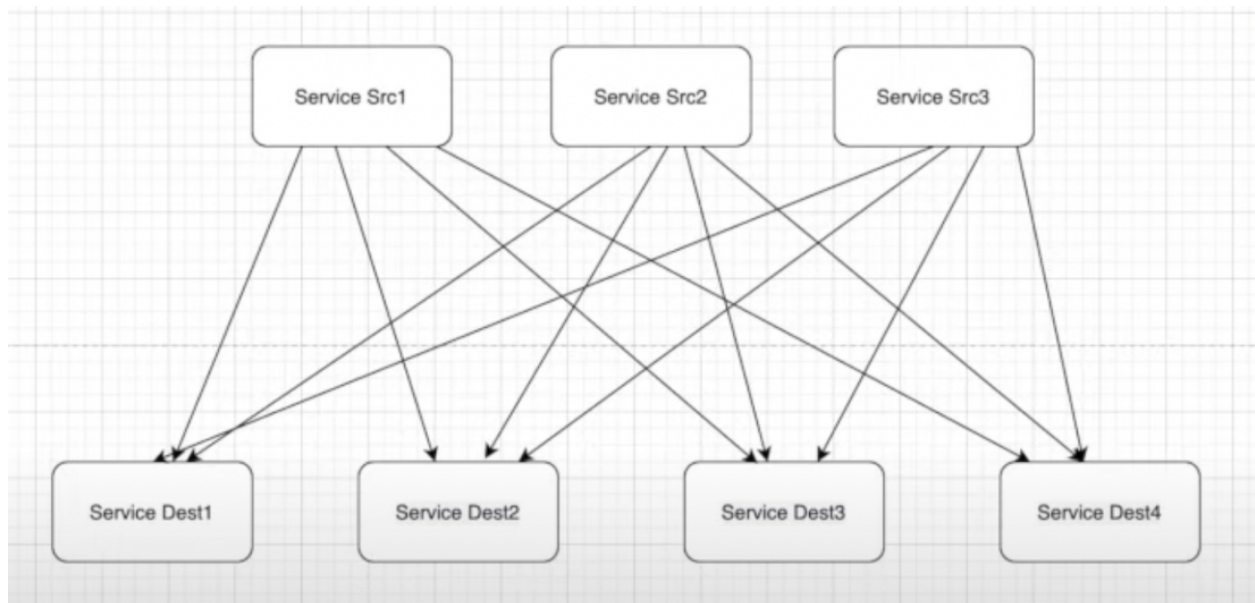
Kafka

1. Kafka is a distributed message streaming platform that uses publish and subscribe mechanism to stream the records.
2. Developer - LinkedIn => Apache
3. Open Source

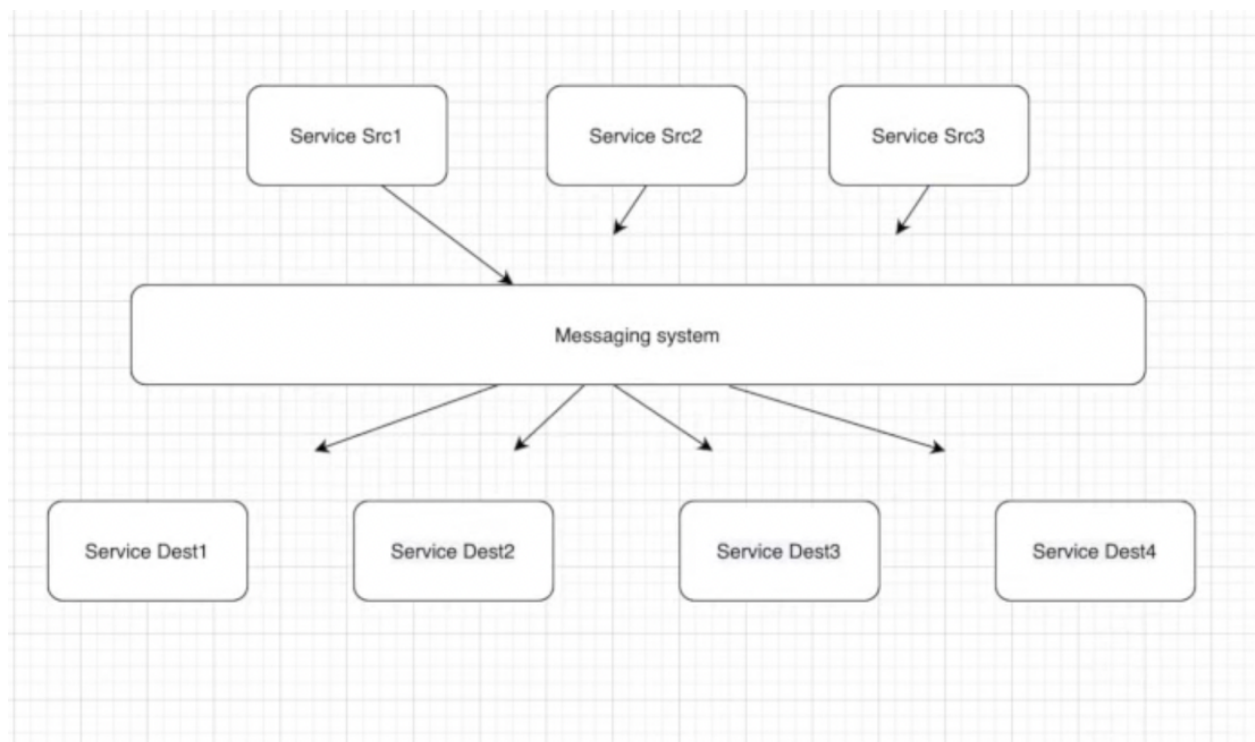
Data Availability & Replication



Sending Messages from A to B



Sending Messages from A to B using Messaging System



Messaging System

A messaging system is responsible for transferring data from one application to another so the application can focus on data without getting bogged down on data transmission and sharing.

1. Point to point
 - a. Messages are persisted in a Queue
 - b. A message can be consumed by only one receiver
 - c. There is no time dependency for the receiver to receive the message
 - d. After receiving the message from queue, receiver will send an acknowledgement back to the sender

Sender ==>> Queue ==>> Receiver

2. Publish - Subscribe
 - a. Messages are persisted in a topic
 - b. A message can be consumed by any number one receiver
 - c. There is time dependency for the receiver to receive the message
 - d. No acknowledgement

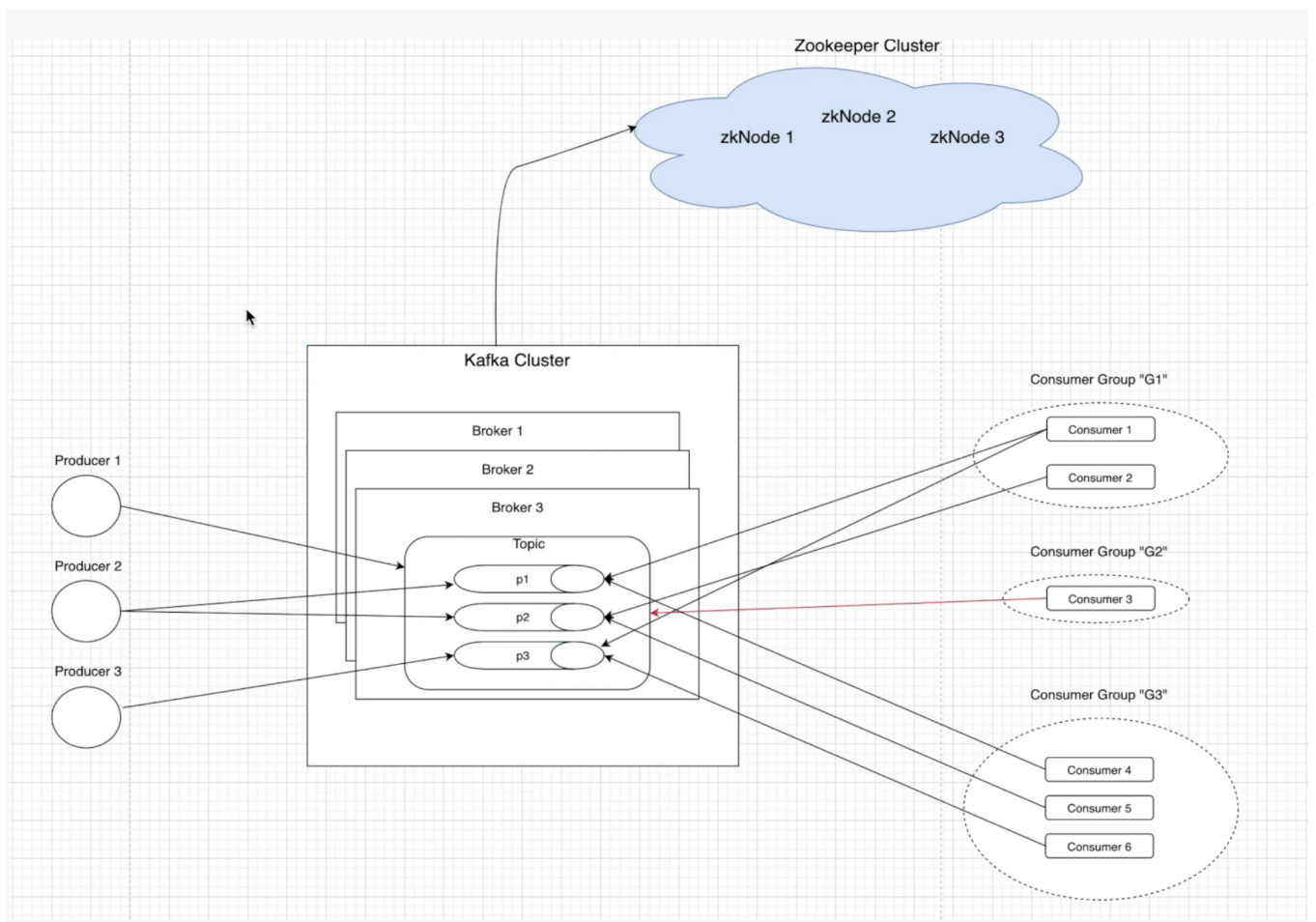
Publisher ==>> Topic ==>> Multiple Subscriber

Kafka Architecture

1. Kafka Architecture \leq Kafka Cluster
2. Kafka Cluster \leq Multiple brokers
3. Broker \leq Topic
4. Topic \leq Multiple Partitions

1. Kafka Cluster managed by Zookeeper Cluster
2. Zookeeper Cluster \leq Multiple zookeeper nodes

1. Producers can produce messages on topic level as well as partition level
2. Consumers can consume messages on topic level as well as partition level
3. Each consumer belongs to a consumer group
4. A consumer group is a group of related consumers that performs a task



Topic

1. A set of messages belonging to a particular category is called a topic.
2. Similar to a table in mysql database.
Eg - Employee Table == Employee_Topic
3. Topic names should be unique within a cluster.
4. We can create as many topics as we want.
5. Two properties
 - a. Partitions
 - i. Topics are split into partitions
 - ii. All messages within a partition are ordered and immutable
 - iii. Each message within a partition has a unique id known as offset
 - b. Replication
 - i. Replicas are backups of partitions
 - ii. Replicas are never read or write data
 - iii. Replicas are used to prevent data loss

Brokers

1. Brokers are simple software process which maintain and manage the published messages
2. Also known as kafka servers
3. Also manage consumer offset and ensures delivery of messages to the right consumer
4. A set of brokers communicating each other to perform the management and maintenance task are collectively known as kafka cluster
5. We can add more brokers in a running kafka cluster without any downtime

Zookeeper

1. Used to monitor kafka cluster and co-ordinate with each broker
2. Keeps all the metadata information related to kafka cluster in form of a key-value pair
3. Metadata includes configuration and health status of each broker
4. A set of zookeeper nodes working together to manage other distributed systems is known as zookeeper cluster

Kafka Features

1. Scalable
2. Fault tolerance
3. Durable
4. Performance
5. No data loss
6. Zero down time
7. Reliable

Kafka Provides

1. Producer APIs
2. Consumer APIs
3. Stream APIs
4. Connector APIs
5. Admin APIs

Google Docs -

<https://docs.google.com/document/d/1ERQstYGpaZpphYYZKyt2jvD-hHTJCVdRPEer1kT2fig/edit>