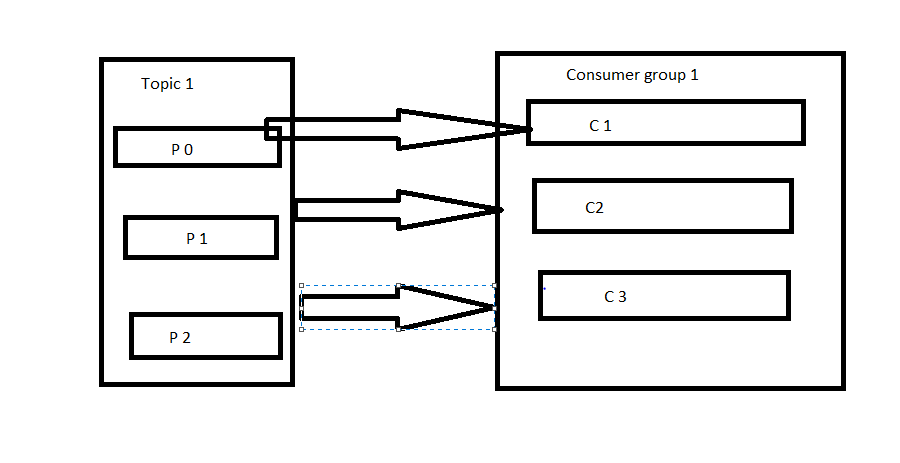
Kafka Consumer:

This component consumes the message from the kafka topic (They Need to Subscribe to that topic).

Usually number of consumers will be equal to number of partitions in the topic (Parallel Processing)



Due to multiple consumer there will be a rebalance if the consumer goes down. (This will be automatically handled by kafka)

To Scale up the application add multiple consumer groups and subscribe to the particular partition.

Consumer mauntains its membership by sending heartbeats to a kafka broker.

Can create a consumer by either using Java Code or in command line interface (Provided by kafka)

Schema registry is used in order to maintain the Serializer and Deserialize the messages. (Avro Serializer is the most common)

Config file for Consumer (consumer.properties)

# Zookeeper connection string

# comma separated host:port pairs, each corresponding to a zk

# server. e.g. "127.0.0.1:3000,127.0.0.1:3001,127.0.0.1:3002"

zookeeper.connect=127.0.0.1:2181

# timeout in ms for connecting to zookeeper

zookeeper.connection.timeout.ms=6000

#consumer group id

group.id=test-consumer-group

#consumer timeout

#consumer.timeout.ms=5000

Create topic command

Bin/kafka-topics.sh --zookeeper 127.0.0.1:2181 –create --topic my-first-topic –partitions 3 --replication-factor 3

To consume messages

Bin/kafka-console-consumer.sh –bootstrap-server 192.168.56.101:9102 –topic my-first-topic

To describe topics [Gives details about topics with Partion and ISR]

Bin/kafka-topics.sh –zookeeper 127.0.0.1:2181 –describe –topic my-first-topic