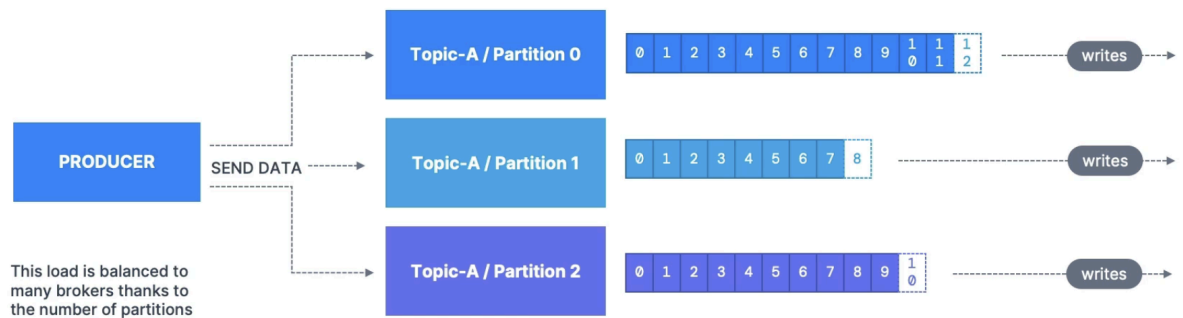


## Producers:

- Producers write data to topics(which are made up of partitions)
- Producers know to write to which partition(and which kafka broker, that is server has it)
- In case of Kafka broker failures, Producers will automatically recover

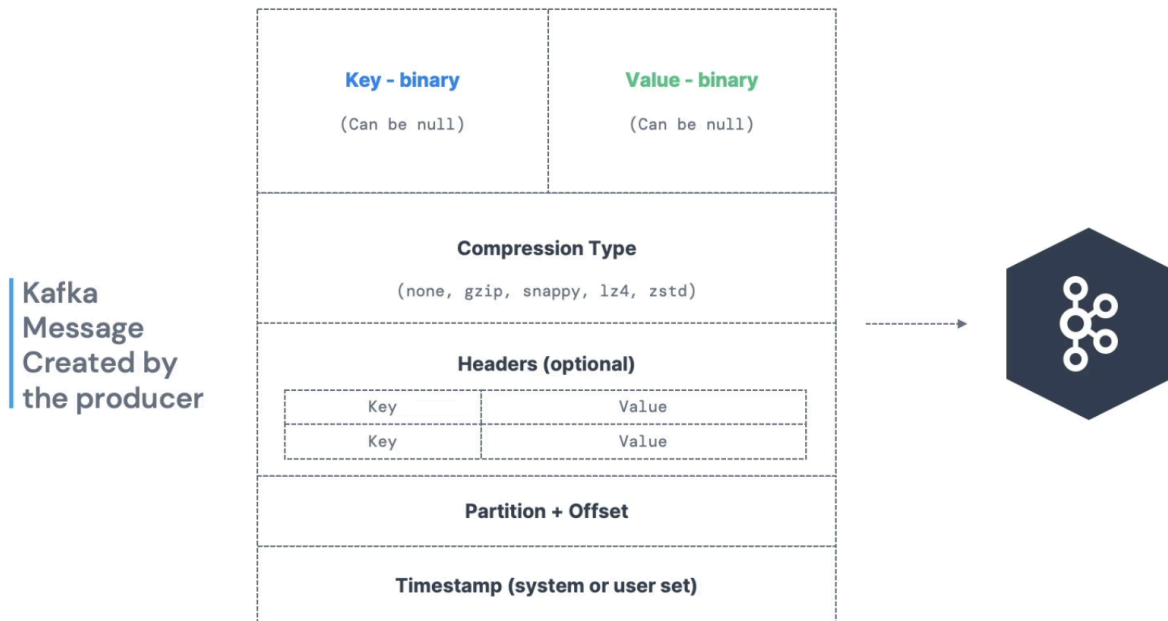


## Producer: Message Keys

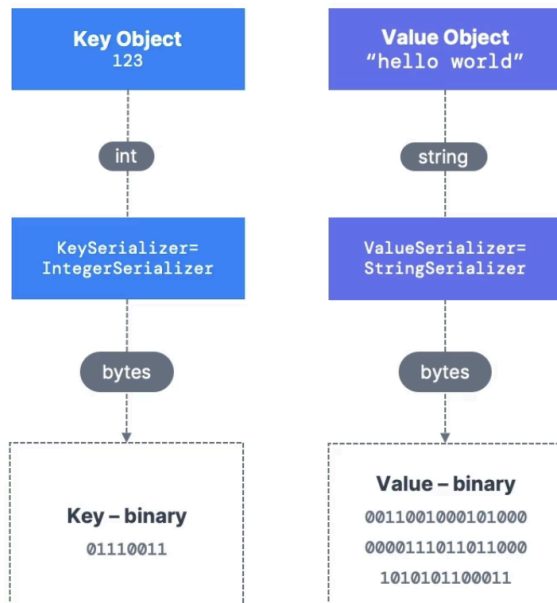
- Producers can choose to send a key with the message (string, number, binary, etc.)
- If key = null, data is sent in round robin manner (partition 0, then 1, then 2...)
- If key != null, then all the messages for that key will always go to the same partition (hashing)
- A key is typically sent when you need message ordering for a specific field (e.g - truckId)



## Kafka Messages Anatomy...



## Kafka Message Serializer:



- Kafka only accepts bytes as an input from producers and sends bytes as an output to consumers.
- Message serialization means converting objects/data into bytes.
- They are used on the value and the key
- Common Serializers:
  - String (incl. JSON)
  - Int, Float
  - Avro
  - Protobuf

### For the curious: Kafka Message Key Hashing

- A kafka partitioner is a code logic that takes a record and determines to which partition to send it into.



- **Key Hashing** is the process of determining the mapping of a key to a partition.
- In the default partitioner, the keys are hashed using the **murmur2** algorithm with the below formula:

`targetPartition = Math.abs(Utils.murmur2(keyBytes)) % (numPartitions - 1)`