**Kafka Notes**

**Section 1 - Kafka Ecosystem - The Big Picture.**

Kafka adopted Pub Sub messaging system architecture.

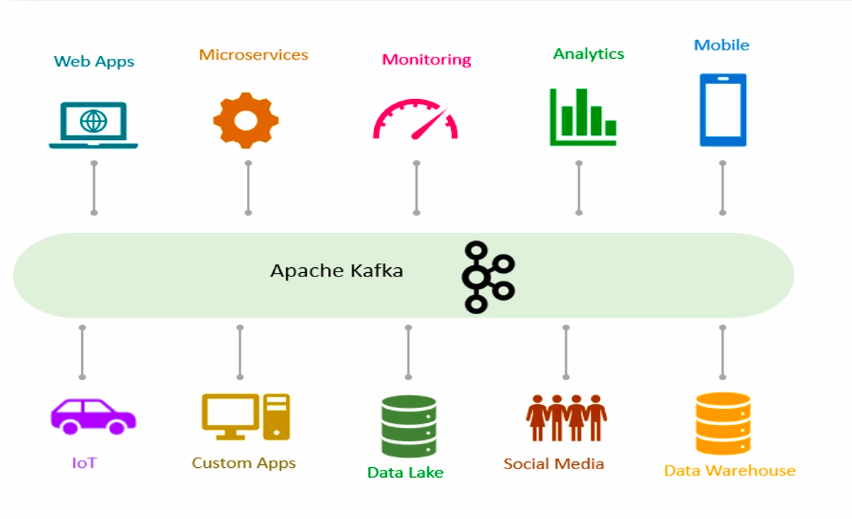
typical messaging system has 3 components:

* Message Producer
* Message Broker
* Message Consumer

Origin of Kafka: https://www.linkedin.com/pulse/kafkas-origin-story-linkedin-tanvir-ahmed

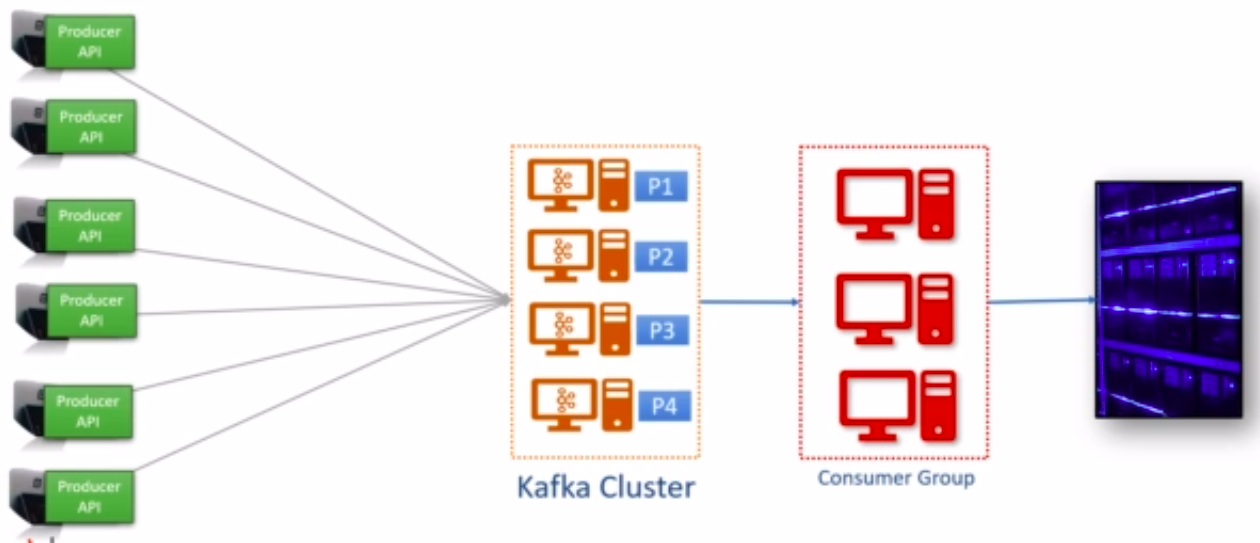
Apache Kafka Ecosystem:

* Kafka Broker
* Kafka Client API
* Kafka Connect
* Kafka Streams
* KSQL



Apache Kafka Core Concepts:

* Producer
* Consumer
* Broker
* Cluster
* Topic
* Partitions
  + an architect would take decision on no.of topic.
  + Partition is the core idea of making kafka distributed and scalable system.
* Offset
* Consumer Groups
  + Maximum possible parallel consumer is limited by no.of partitions in the topic.
  + kafka does not allow more than one consumer to read from same partition simultaneously.

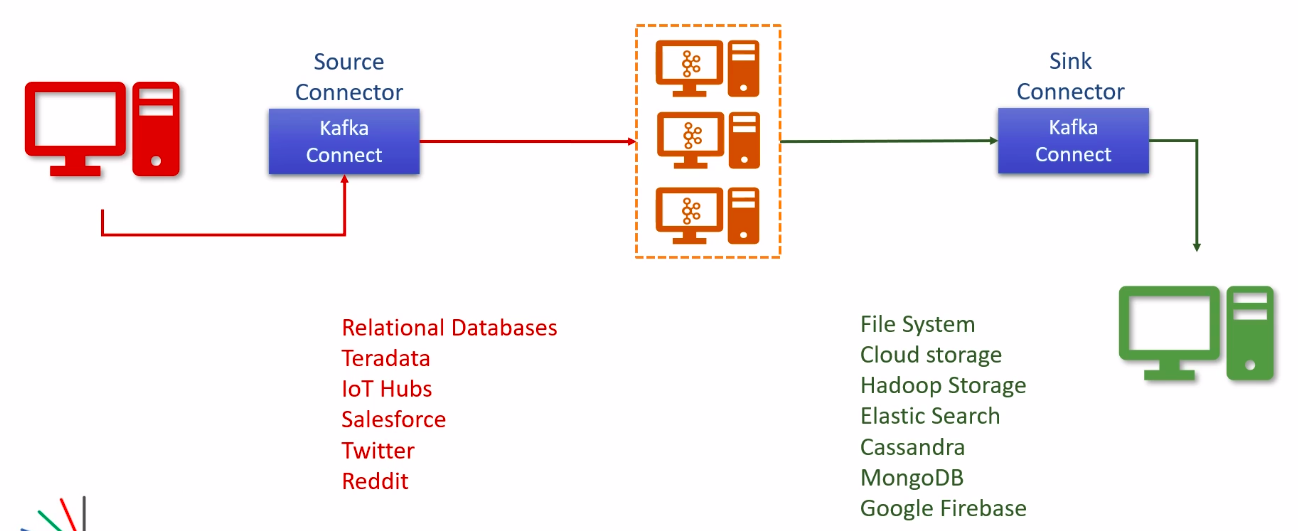


What is **Kafka Connect** ?

Instead of writing a Kafka client to read from DB and push to Kafka, Kafka connect is used.

Source >> Kafka connect (source connector) >> Kafka Broker >> Kafka connect (sink connector) >> Target.

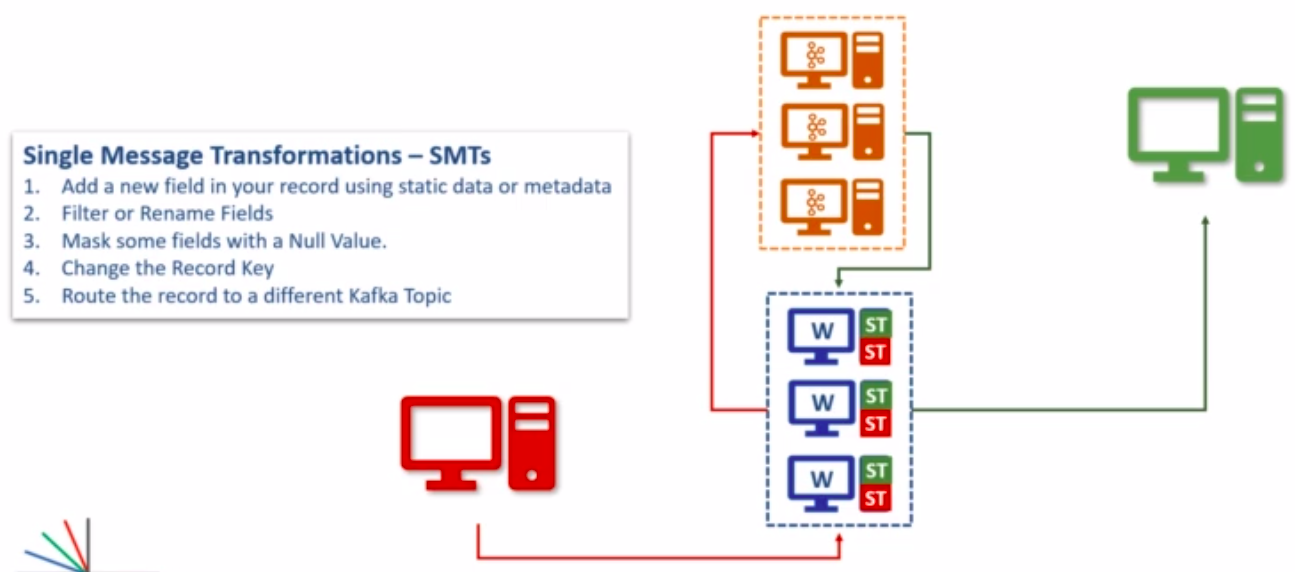
Source/Target : Relational DB's, Teradata, IoT Hubs, Salesforce, Twitter, Reddit, File system, cloud storage, hadoop storage, Elastic search, cassandra, MongoDB, google firebase.



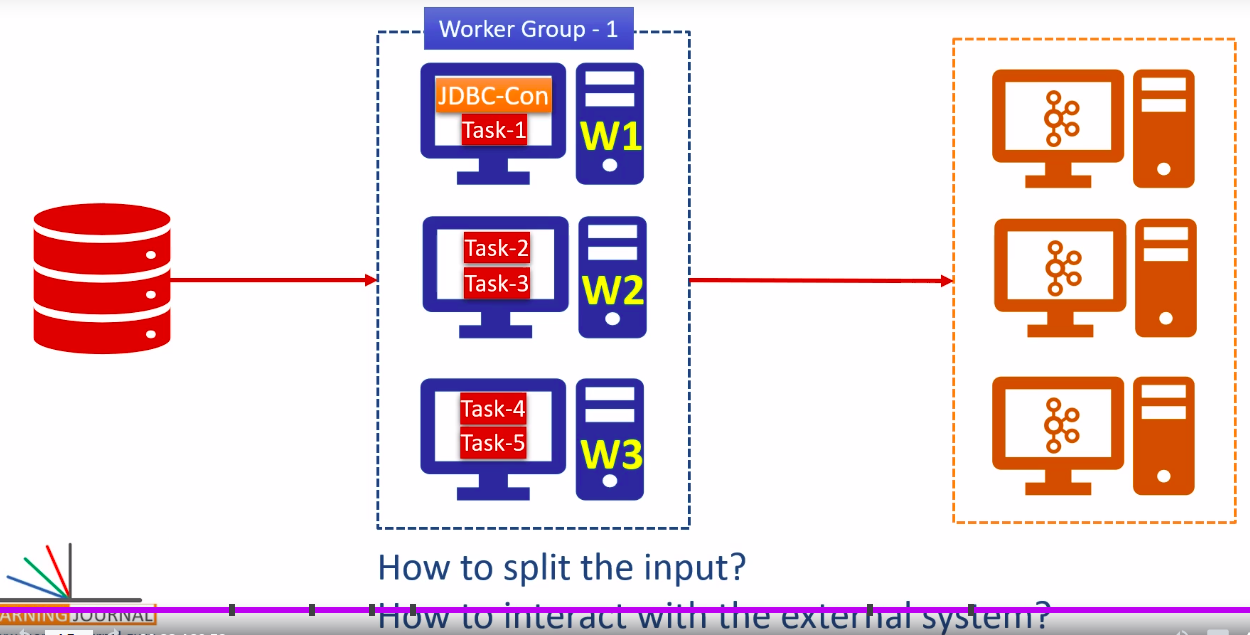
How Kafka Connect works ?

Kafka Connect scalability

kafka Connect Transformations



Kafka Connect Architecture



**Kafka Streams:**

* What is real-time stream processing ?
  + Data streams are unbounded (No definite starting or ending), Often infinite and ever growing, Sequence of data in small packets (kb), eg: sensors, log entries, click streams, transactions, data feeds.
* What is Kafka ?
* Kafka streams architecture ?

TODO:

* Do estimation and decide on no.of partition for each topic, and also check on consumer group and no.of consumers.