Enterprise Architecture

Messaging

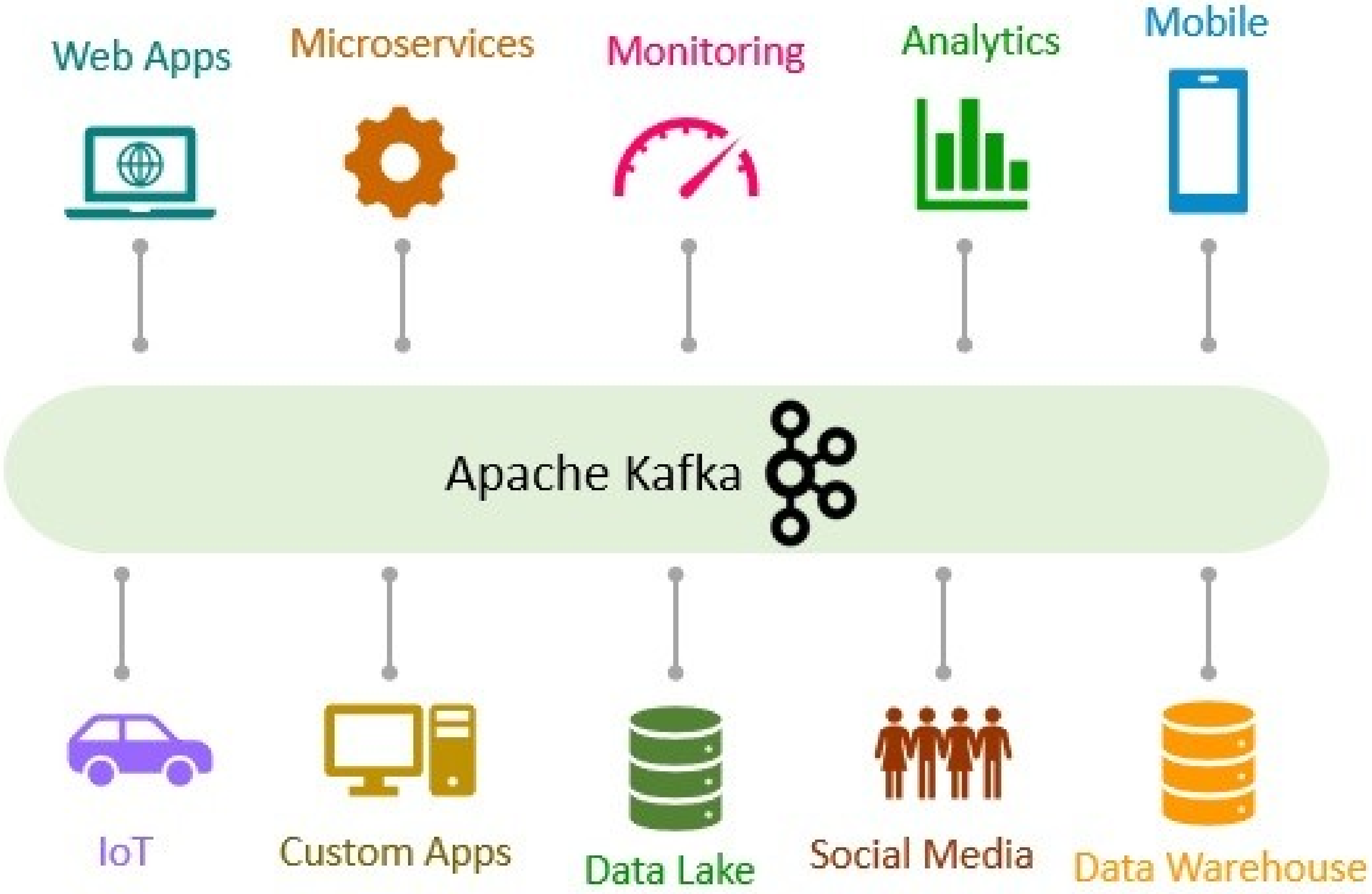
Messaging
Case Study: Kafka

Why we need it

- Let’s watch :
  - Apache Kafka in 5 minutes
    - https://www.youtube.com/watch?v=PzPXRmVHMxI

Kafka vs ESB

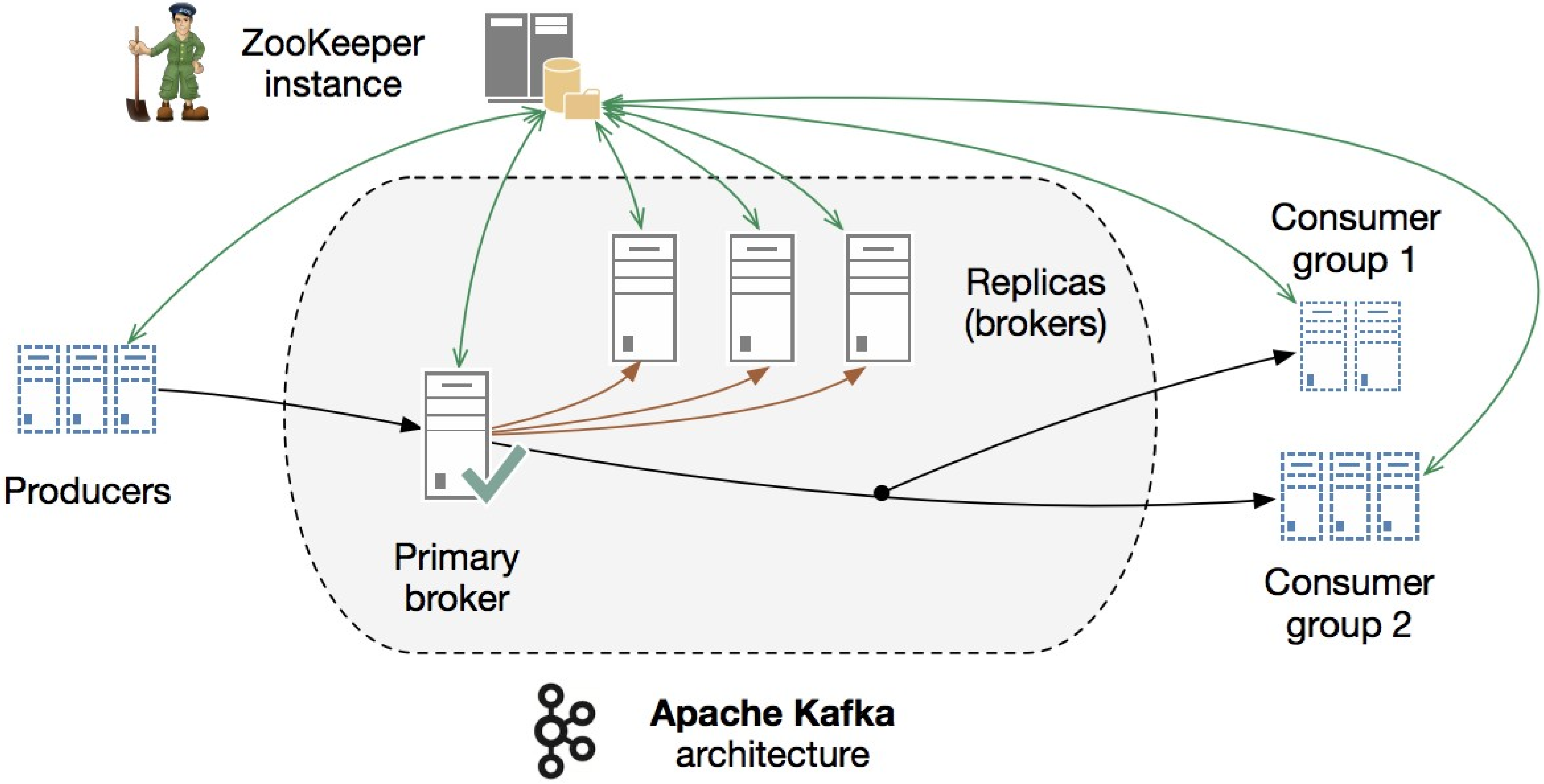
- Legacy Enterprise service bus (e.g. rabbitmq)
  - Heavy logic inside a broker, point-point ,..
- Let’s read:
  - https://content.pivotal.io/blog/understanding-when-to-use-rabbitmq-or-apache-kafka
- Make your own call!

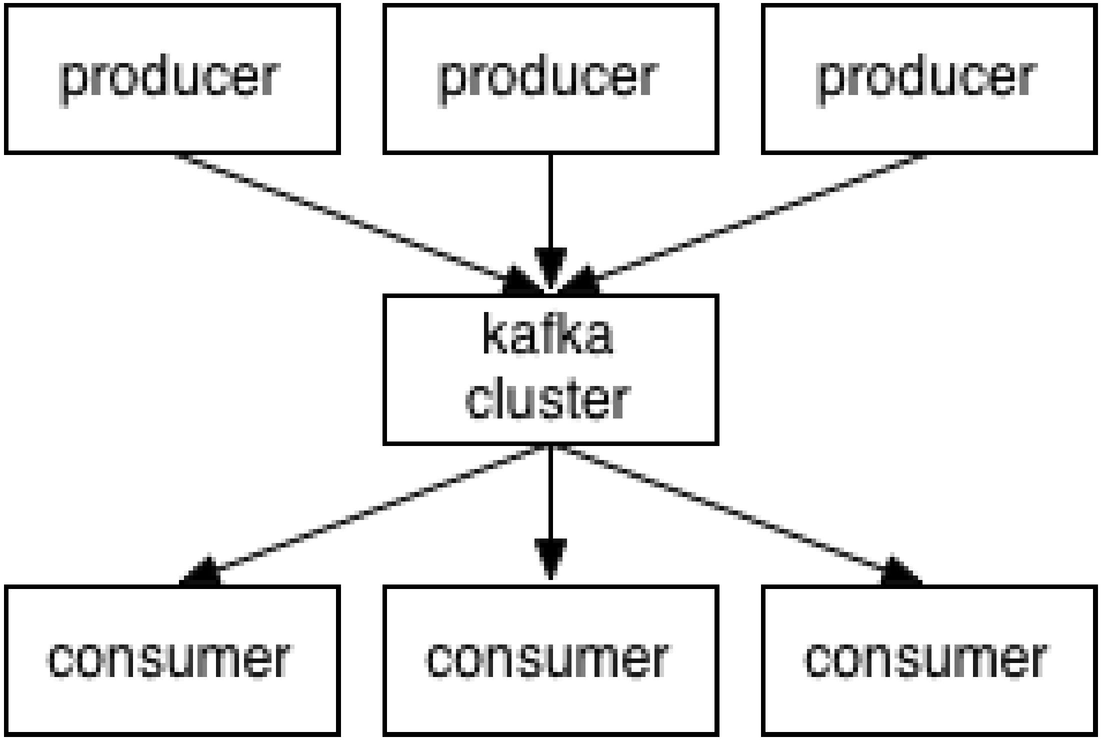


https://www.learningjournal.guru/ebook/kafka-streams-real-time-stream-processing/chapter-5-why-messaging-system/

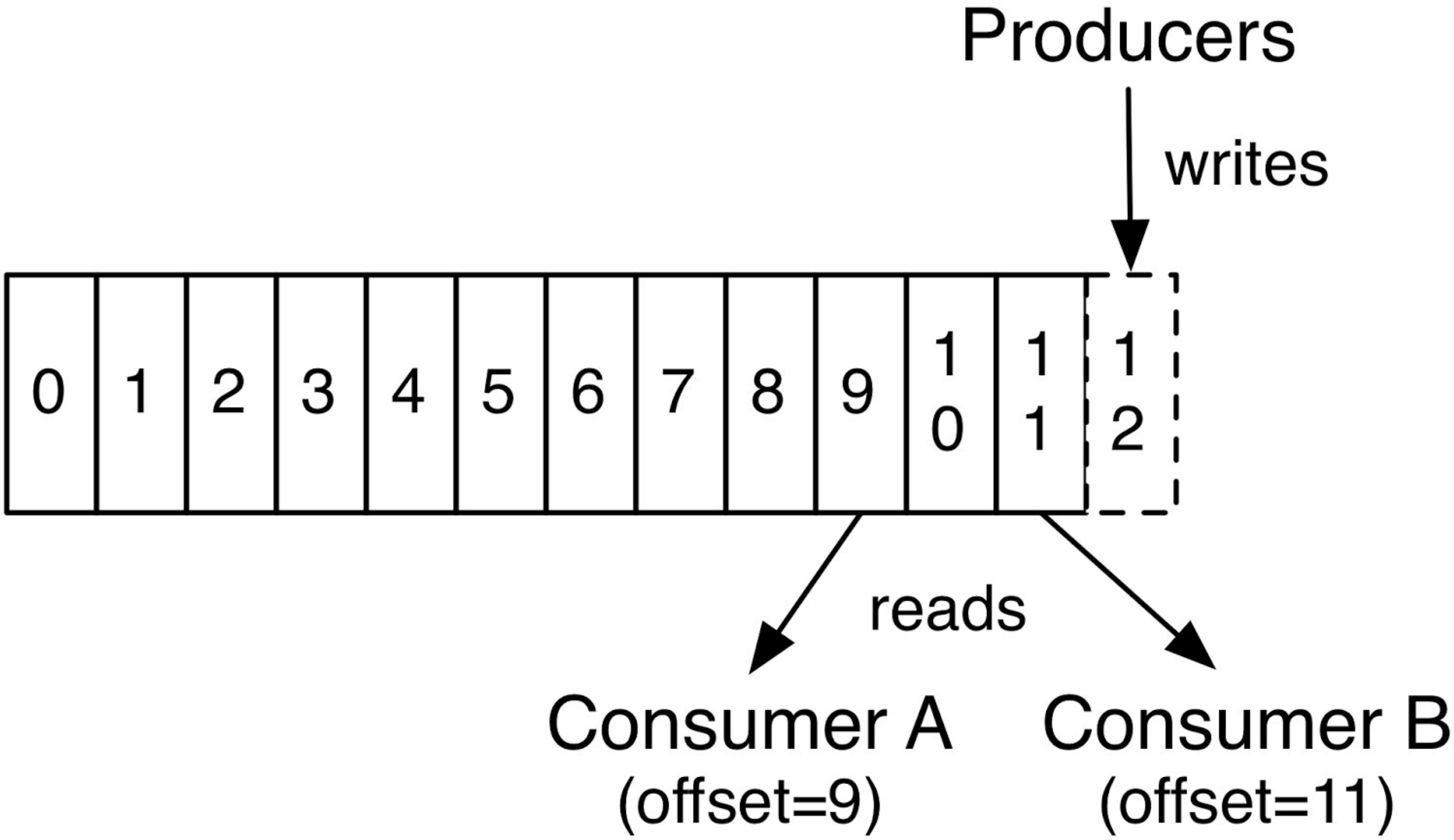
Understanding

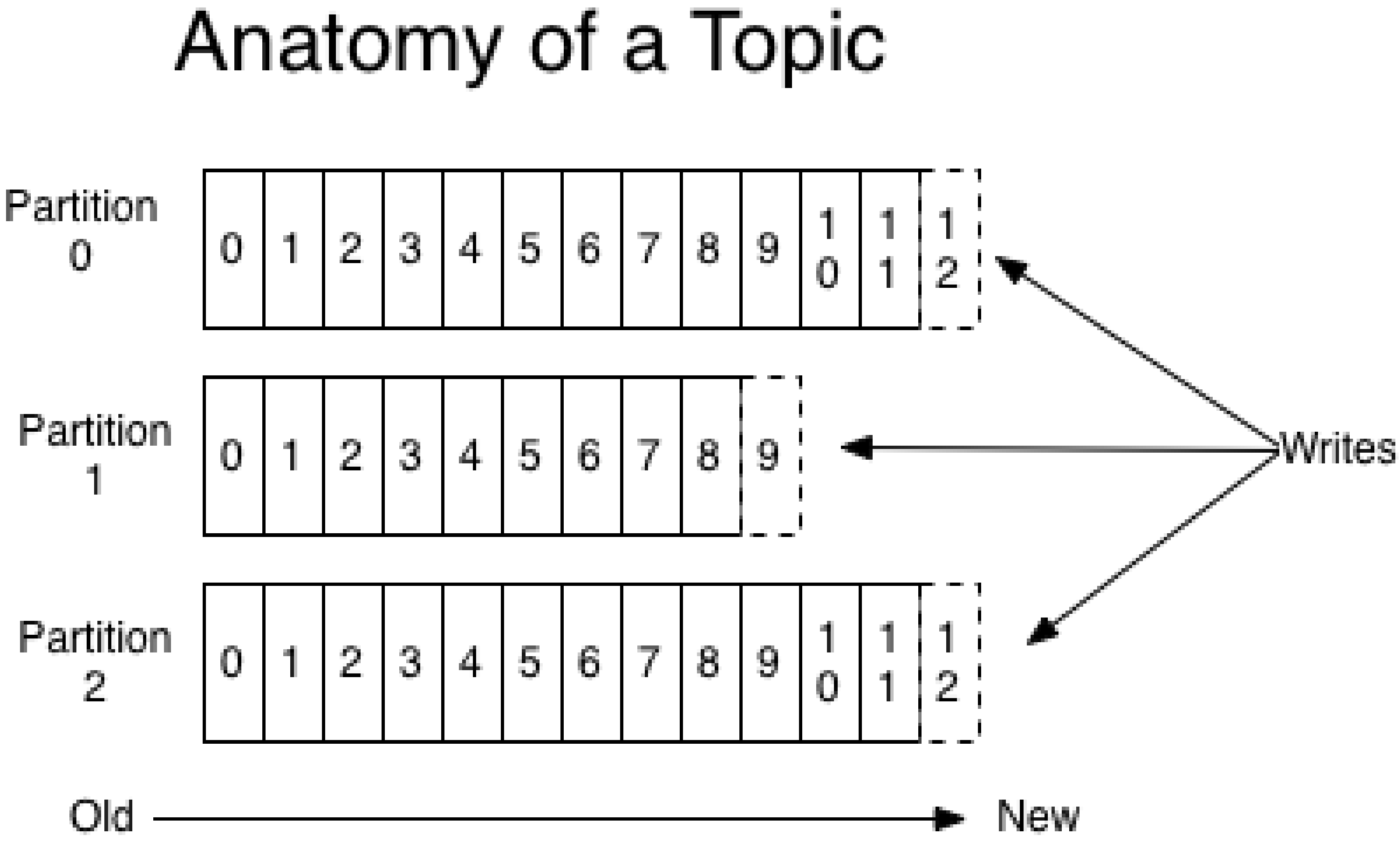
- Understanding Kafka with Lego
  - https://www.youtube.com/watch?v=Q5wOegcVa8E


ZooKeeper
instance
Replicas
(brokers)
Consumer
group 1
Producers
Primary
broker
Consumer
group 2
Apache Kafka
architecture


producer
producer
producer
kafka cluster
consumer
consumer
consumer

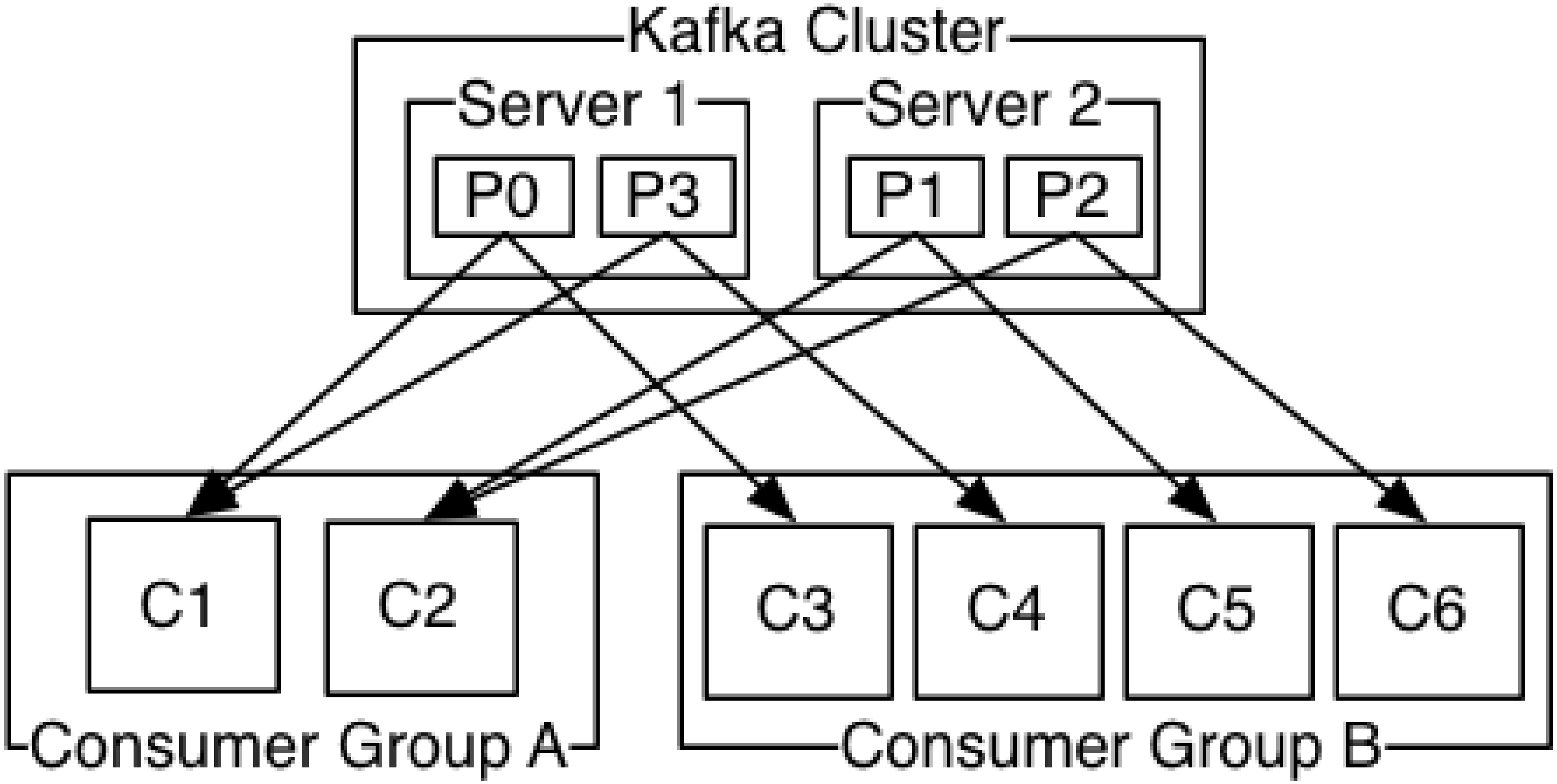
Distributed log





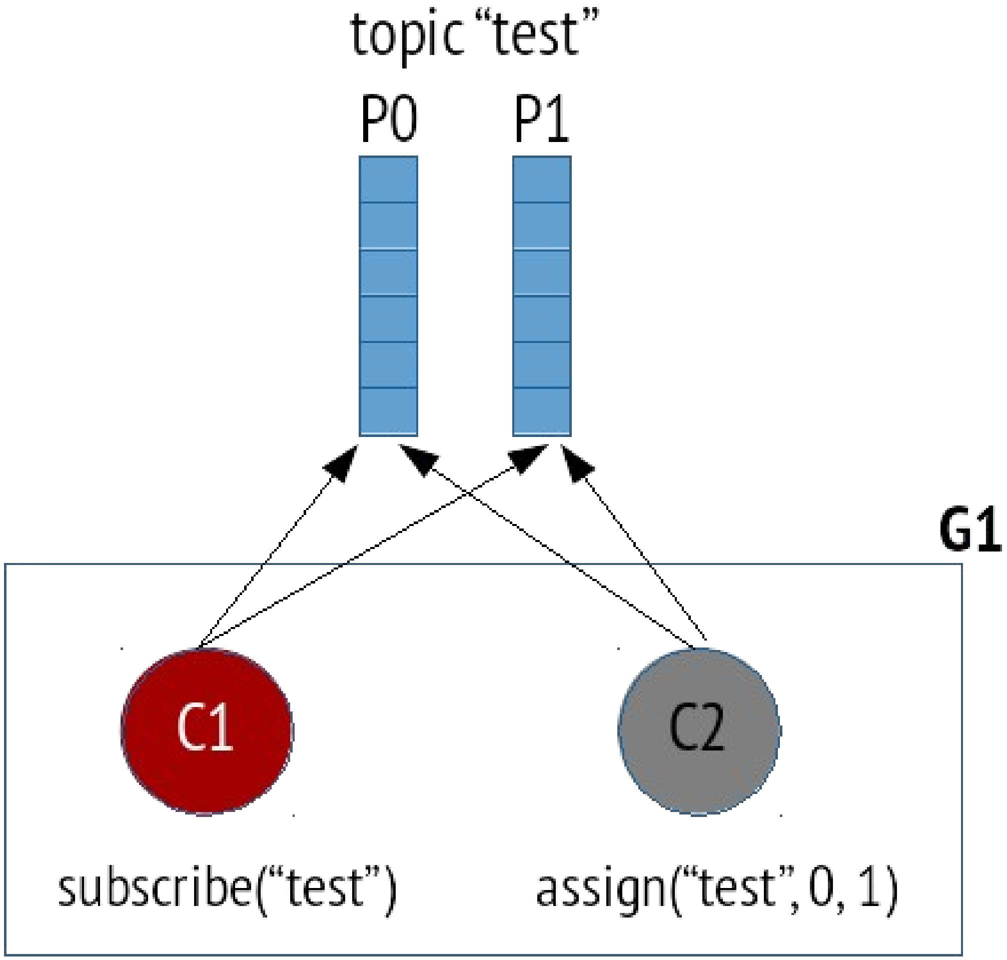
https://developer.ibm.com/streamsdev/docs/improving-application-throughput-consuming-kafka/

Consumer Groups



https://kafka.apache.org/090/documentation.html

Topic Partitions



https://paolopatierno.wordpress.com/2017/07/27/apache-kafka-consumer-groups-dont-use-them-in-the-wrong-way/

Implications

- Messages are only sorted within the same partition
  - Topic as a whole is not sorted
- Duplicate message , out of order ,..
- How to keep track of the offsets ?

Got time ?

- Hear from the creator
- How Does Apache Kafka Work | Jay kreps
  - https://www.youtube.com/watch?v=EiWsPd6JDoo

Final Thoughts

- There’s a need to rethink our patterns around Microservices messaging
  - DDD, Event sourcing, CQRS, …
- Having a messaging platform is important
- Kafka offers scalability and allows for stream processing

Tradeoffs we covered so far

- Four Distributed Systems Architectural Patterns by Tim Berglund (50 min)
  - https://www.youtube.com/watch?v=tpspO9K28PM