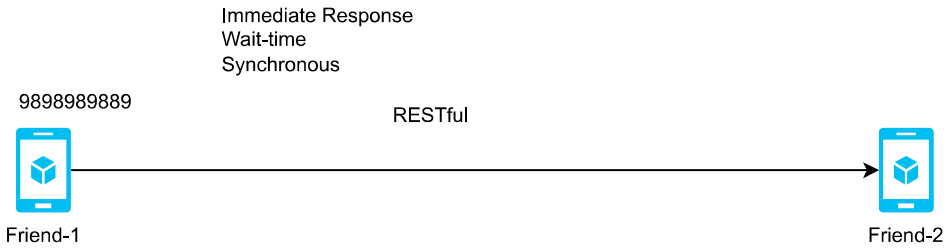
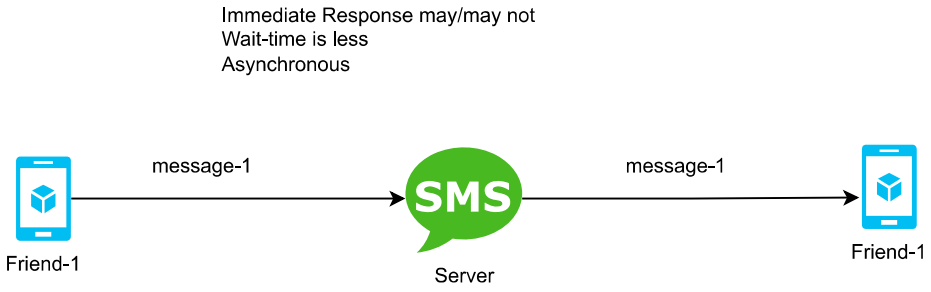
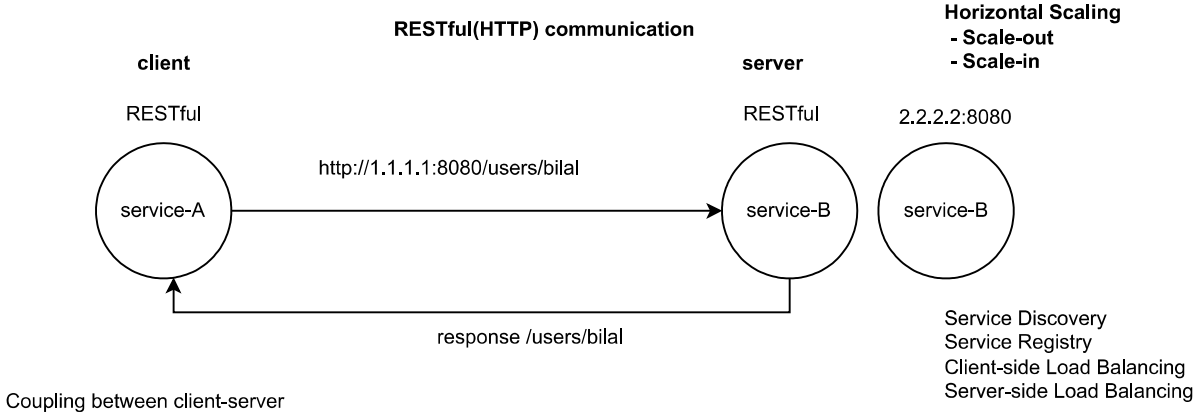
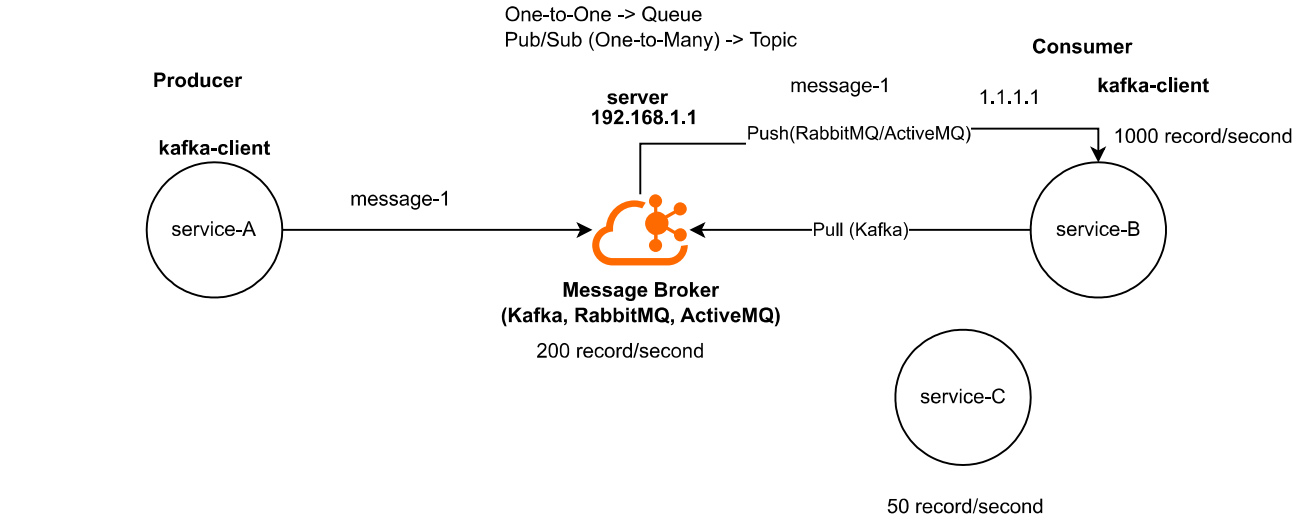
Introduction to Kafka

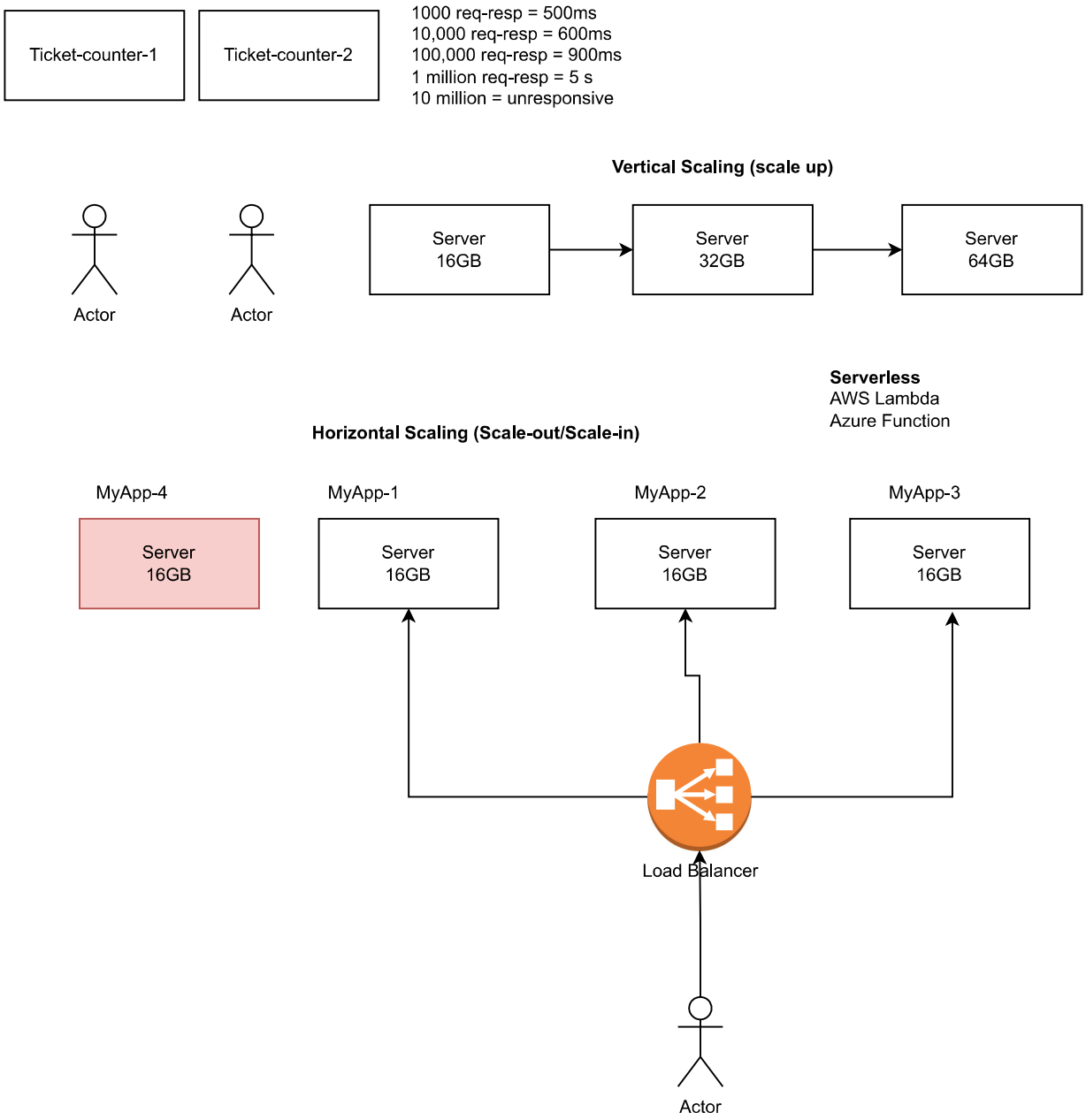




Scaling
Vertical Scaling
Horizontal Scaling
- Scale-out
- Scale-in





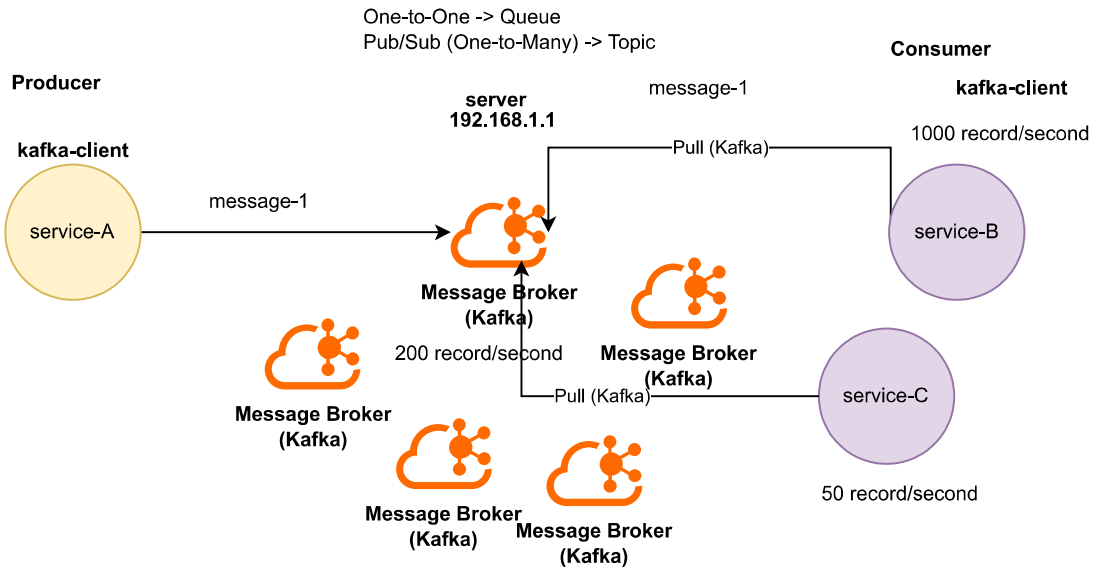

Ticket-counter-1
Ticket-counter-2
1000 req-resp = 500ms
10,000 req-resp = 600ms
100,000 req-resp = 900ms
1 million req-resp = 5 s
10 million = unresponsive
Vertical Scaling (scale up)
Actor
Actor
Server 16GB
Server 32GB
Server 64GB
Serverless
AWS Lambda
Azure Function
Horizontal Scaling (Scale-out/Scale-in)
MyApp-4
MyApp-1
MyApp-2
MyApp-3
Server 16GB
Server 16GB
Server 16GB
Server 16GB
Load Balancer
Actor

Kafka

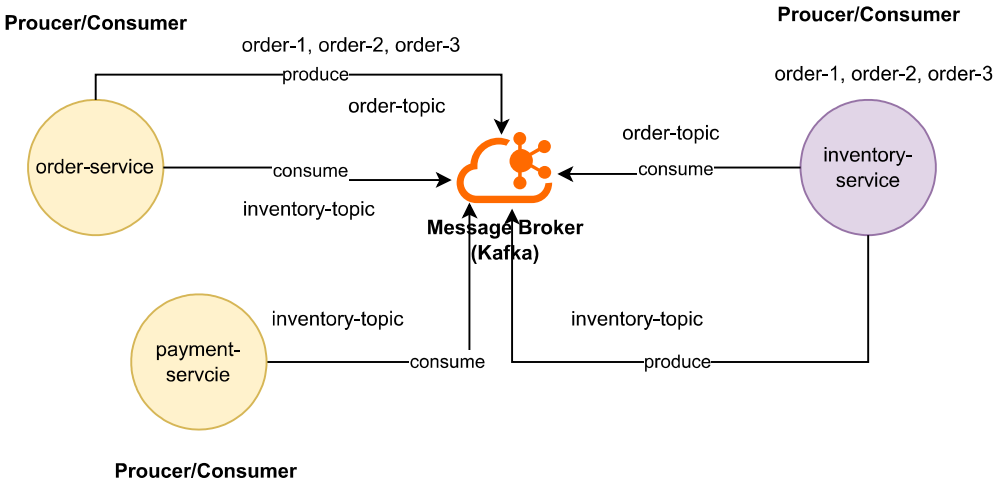
Message Retention : **7 days**

1000events x 3600second x 24hour x 7 days x 500 bytes = 302 GB

Kafka is distributed message broker with high throughput, low latency, resiliency, performant



Event Driven Architecture



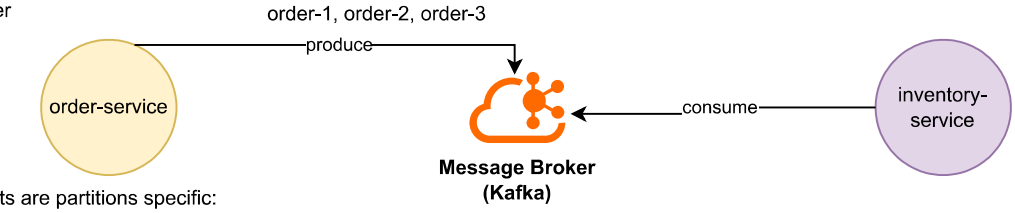
Kafka Terminologies:

Producer
Consumer
Broker
Record(Event)
Key
Topic
Partition - scalable, performance improvement
Consumer Group
Rebalance
Offset
Replication Factor
Zookeeper
KRaft

Kafka Internals

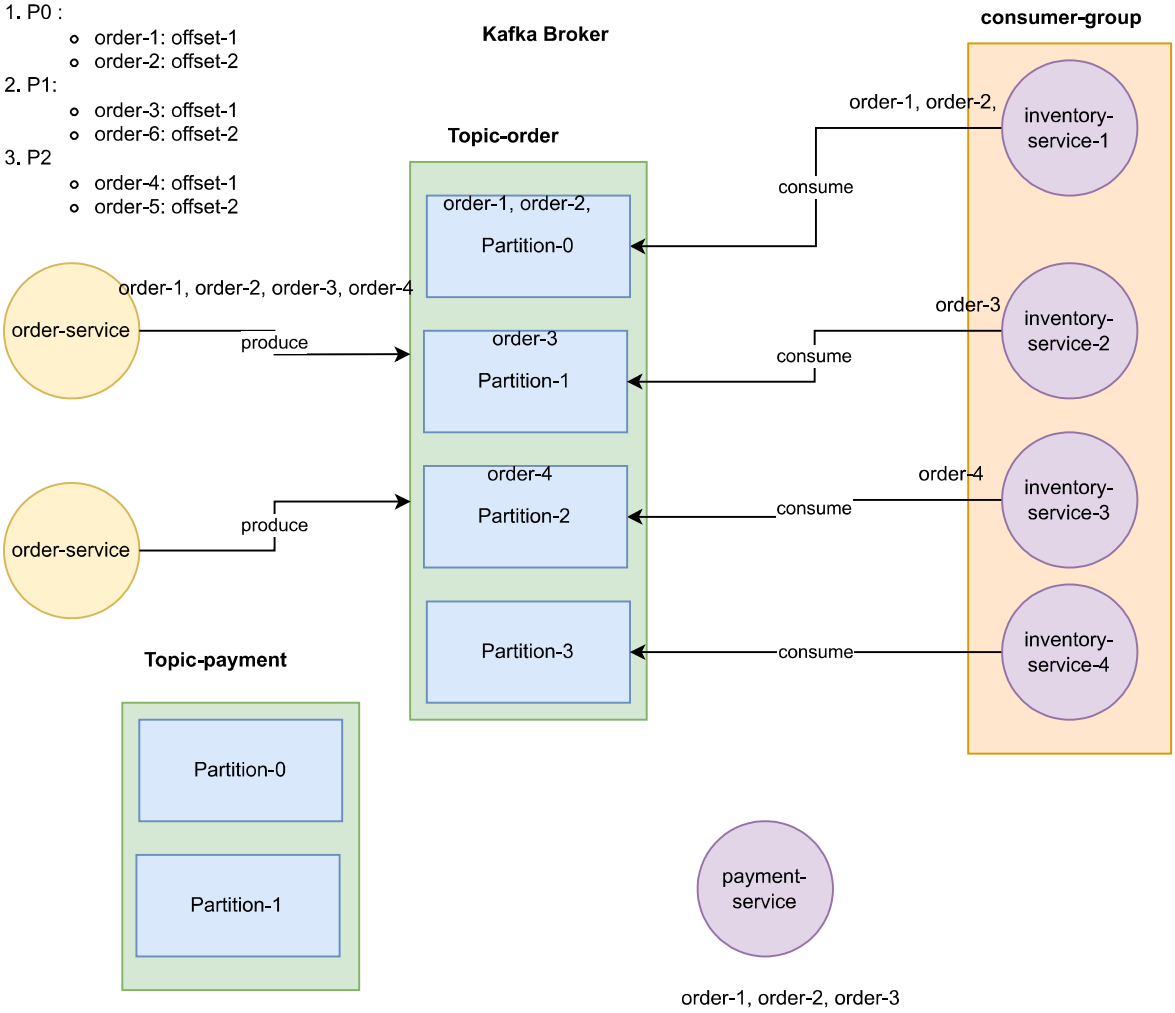
Types of calls:

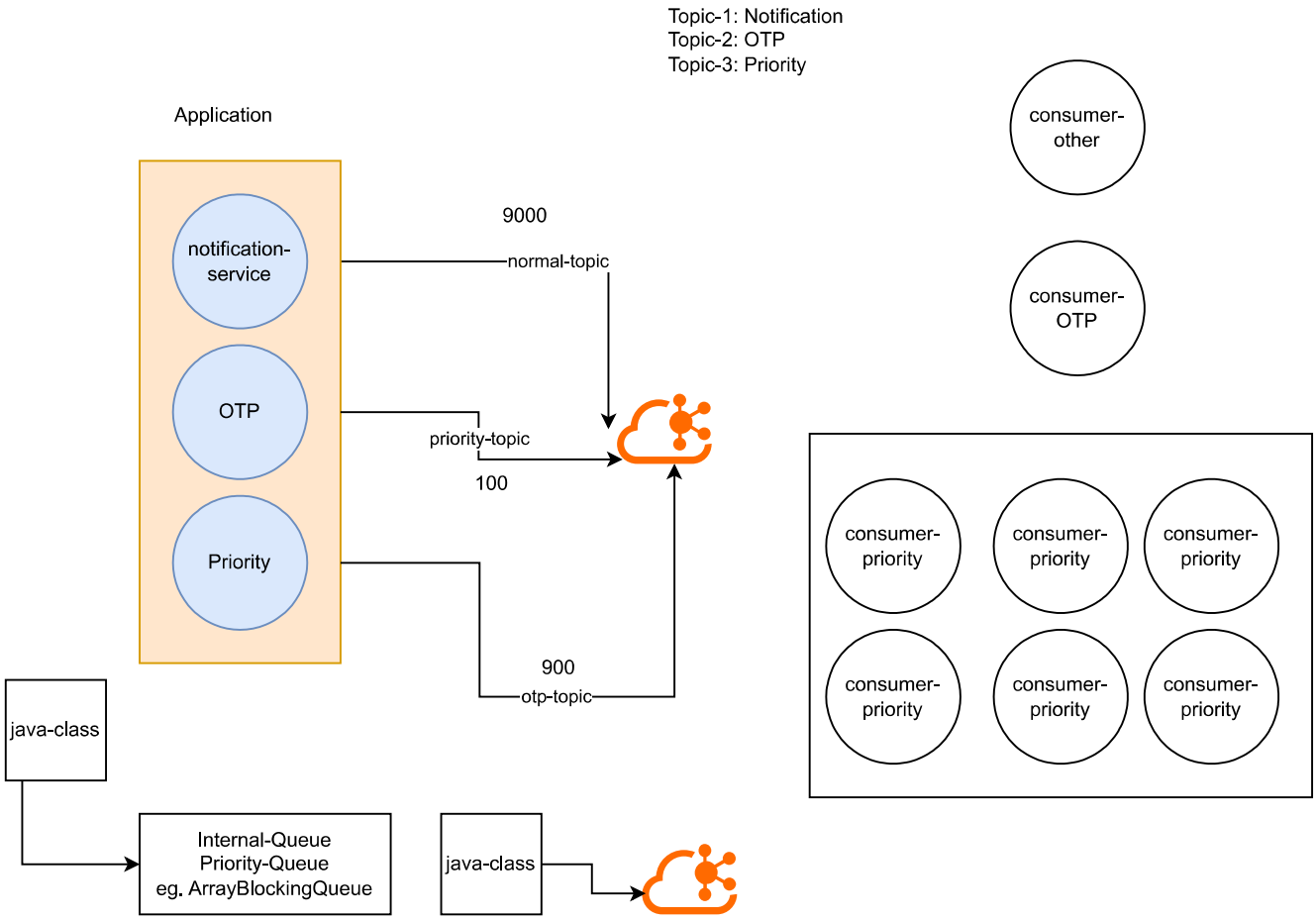
1. Producer
2. Fetch
3. metadata

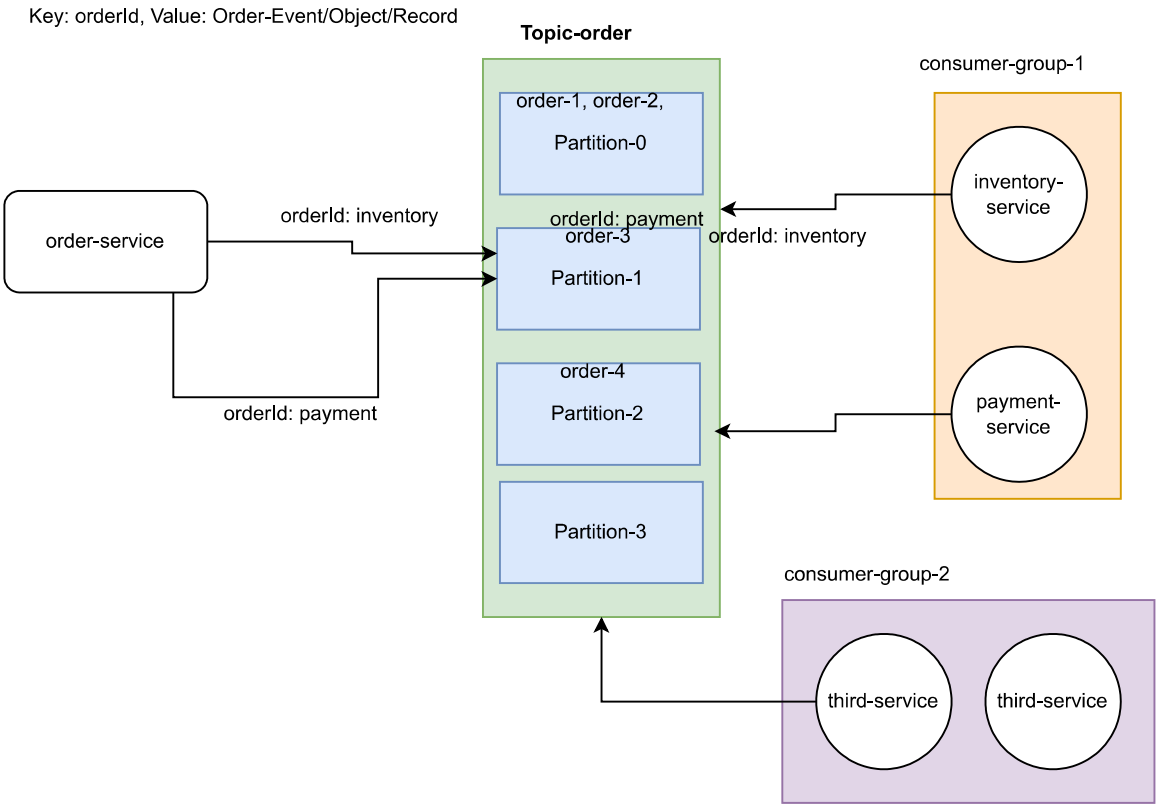


Offsets are partitions specific:
Offsets:

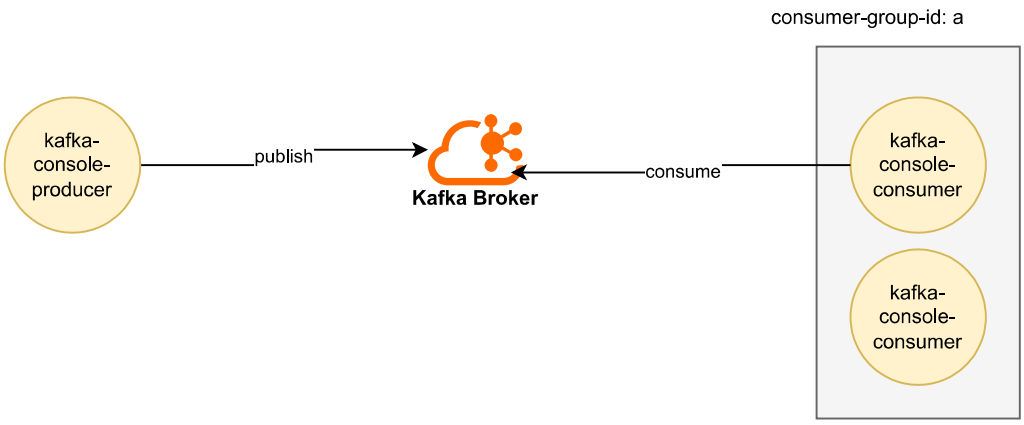
1. P0 :
   - order-1: offset-1
   - order-2: offset-2
2. P1:
   - order-3: offset-1
   - order-6: offset-2
3. P2
   - order-4: offset-1
   - order-5: offset-2




Topic-1: Notification
Topic-2: OTP
Topic-3: Priority
Application
notification-service
OTP
Priority
9000
normal-topic
priority-topic
100
900
otp-topic
consumer-other
consumer-OTP
consumer-priority
consumer-priority
consumer-priority
consumer-priority
consumer-priority
consumer-priority
java-class
Internal-Queue
Priority-Queue
eg. ArrayBlockingQueue
java-class


Key: orderId, Value: Order-Event/Object/Record
Topic-order
order-1, order-2,
Partition-0
order-3
Partition-1
order-4
Partition-2
Partition-3
consumer-group-1
inventory-service
payment-service
consumer-group-2
third-service
third-service
order-service
orderId: inventory
orderId: payment
orderId: payment
orderId: inventory

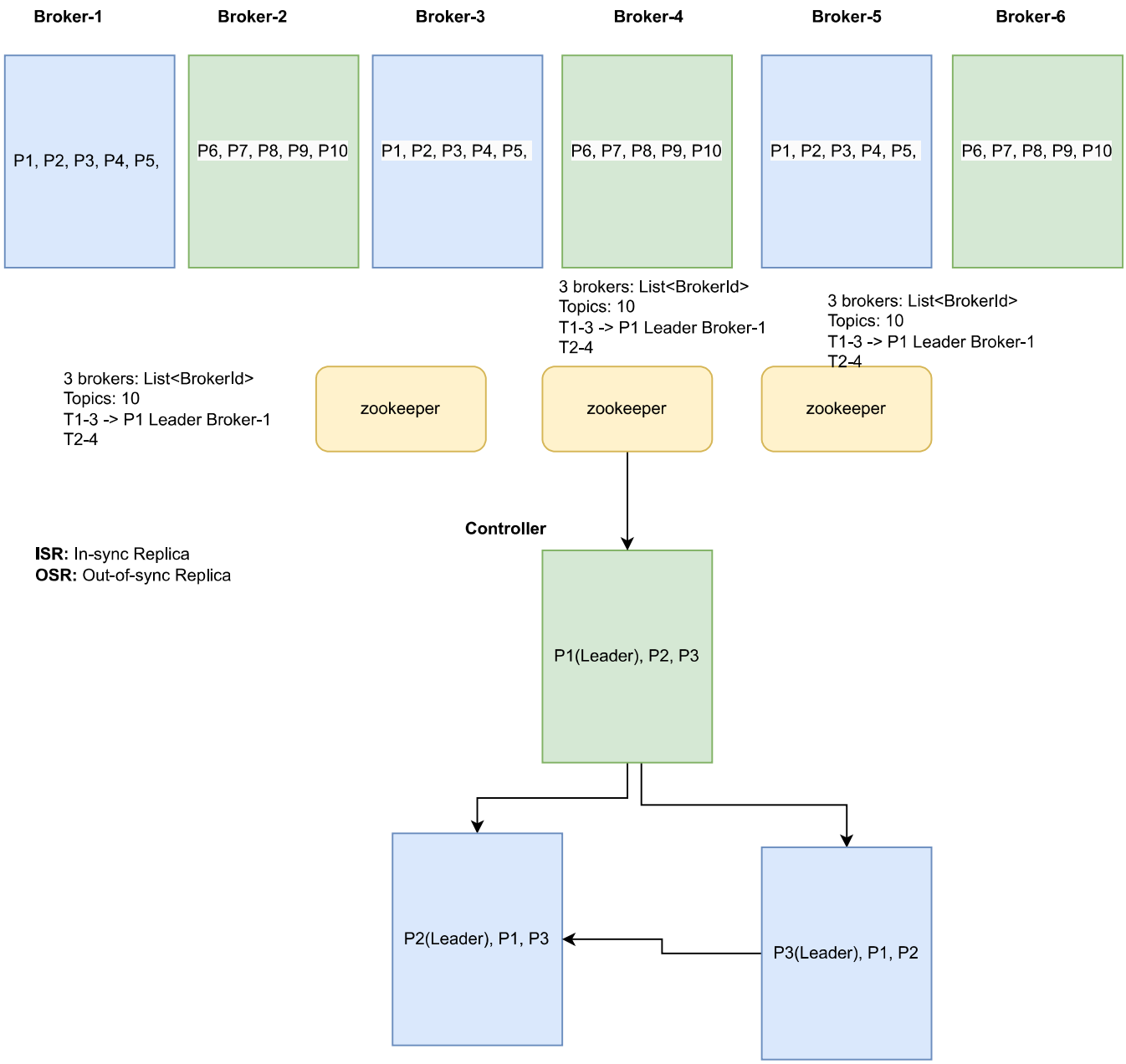
Kafka Handson



Replication Factor

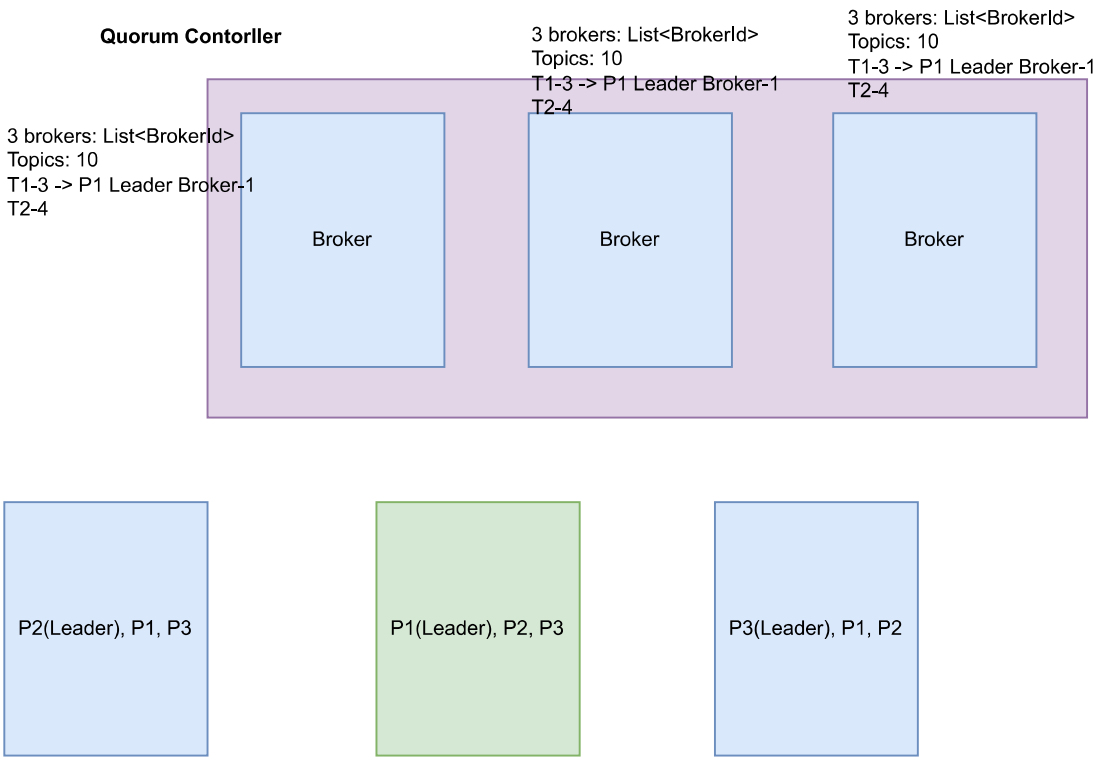
Brokers# 6; Topic: mytopic; Partitions: 10; Replication Factor: 3

Parition- 10 x 3 / 6-Broker = 5 replicas per broker



ISR: In-sync Replica
OSR: Out-of-sync Replica

KRAFT


Quorum Contorller
3 brokers: List<BrokerId>
Topics: 10
T1-3 -> P1 Leader Broker-1
T2-4
3 brokers: List<BrokerId>
Topics: 10
T1-3 -> P1 Leader Broker-1
T2-4
3 brokers: List<BrokerId>
Topics: 10
T1-3 -> P1 Leader Broker-1
T2-4
Broker
Broker
Broker
P2(Leader), P1, P3
P1(Leader), P2, P3
P3(Leader), P1, P2