

ESBD 4

Sistemas Distribuídos



INFORMAÇÃO,
TECNOLOGIA
& INOVAÇÃO

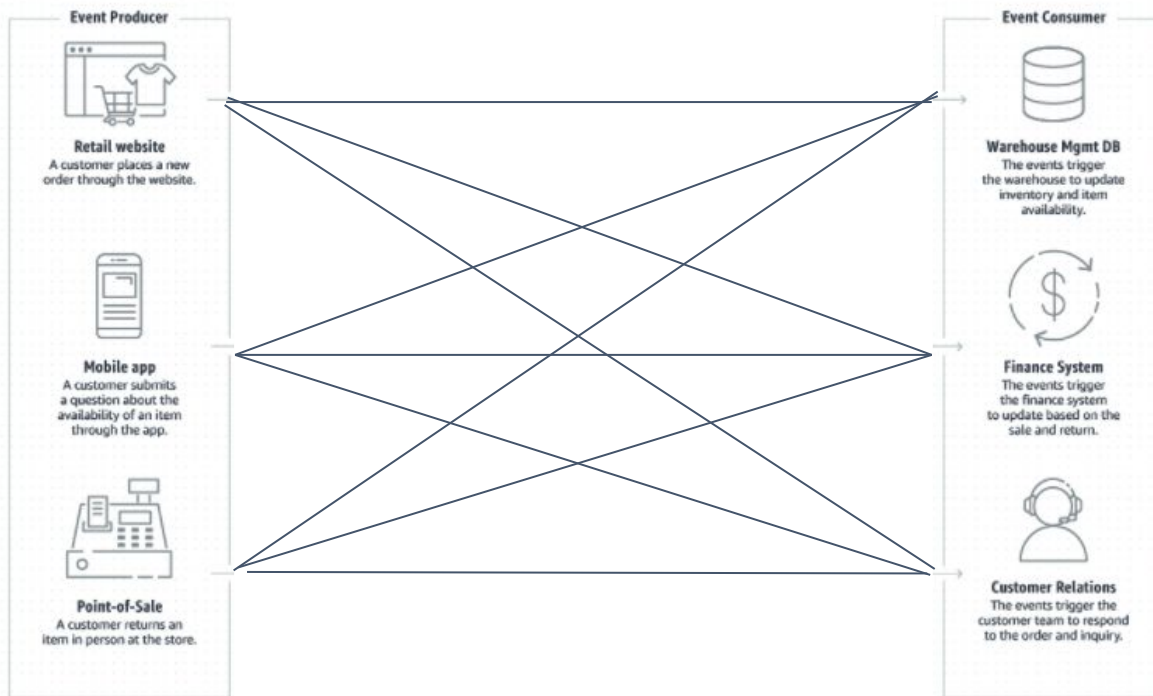
Arquiteturas Orientadas a Eventos

Arquiteturas Orientadas a Eventos



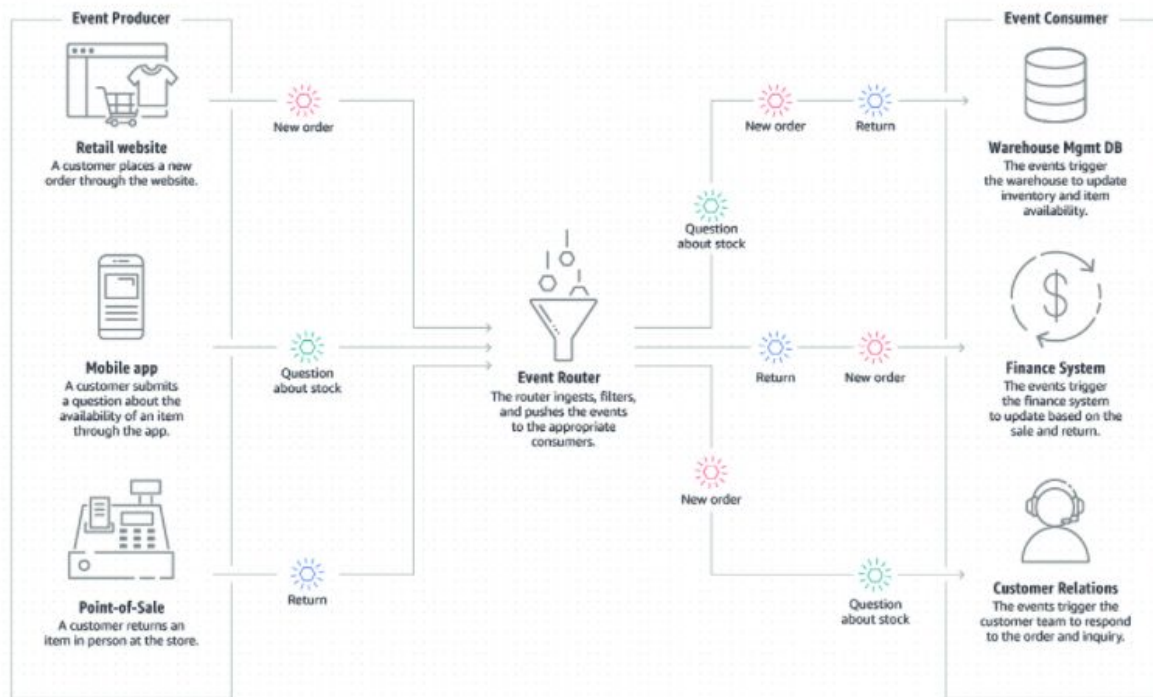
Arquiteturas Orientadas a Eventos

Arquiteturas Orientadas a Eventos



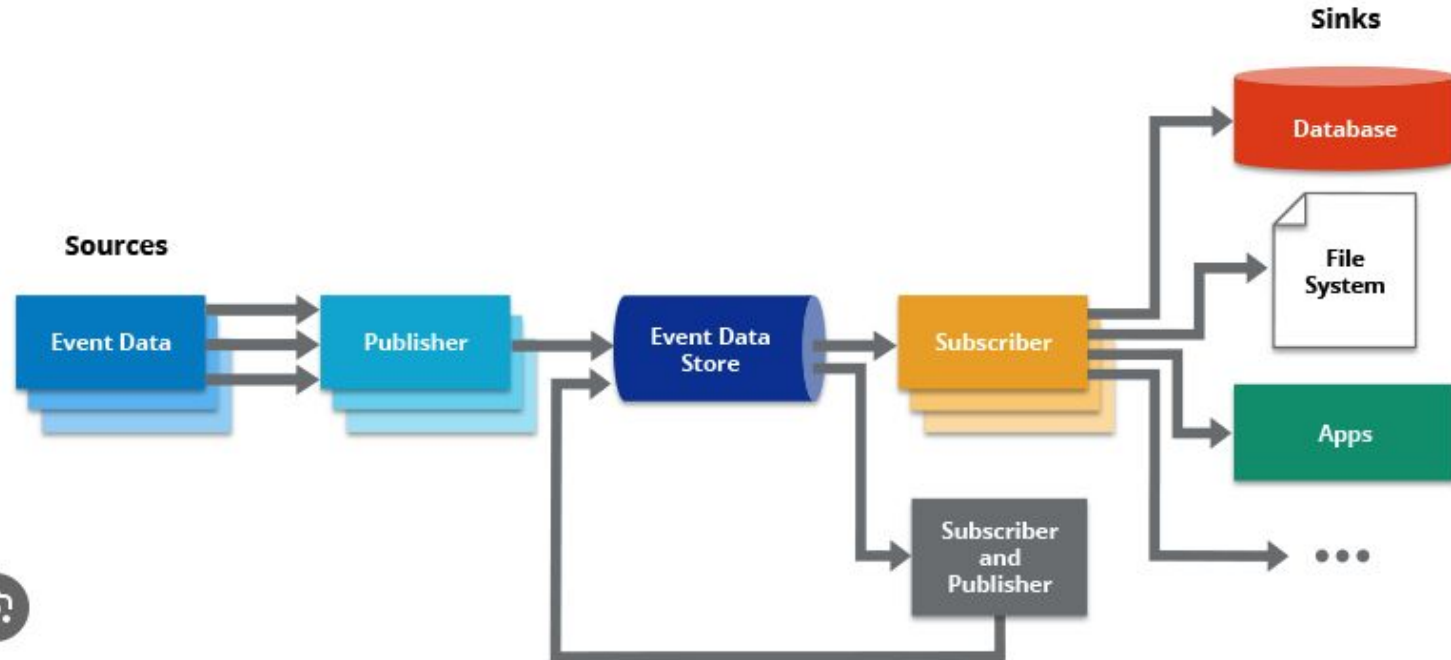
Arquiteturas Orientadas a Eventos

Arquiteturas Orientadas a Eventos



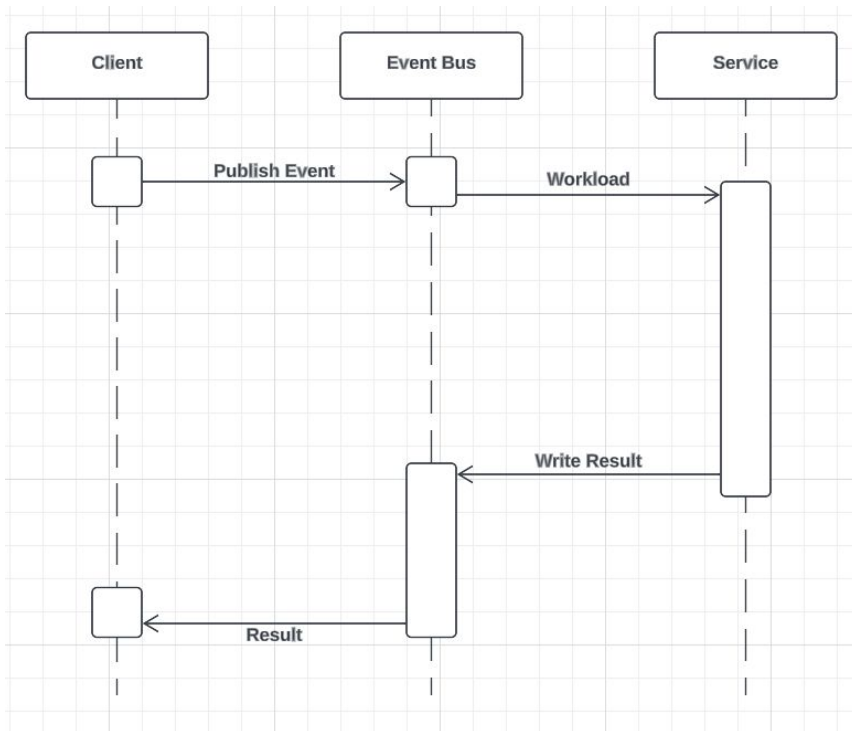
Arquiteturas Orientadas a Eventos

Arquiteturas Orientadas a Eventos



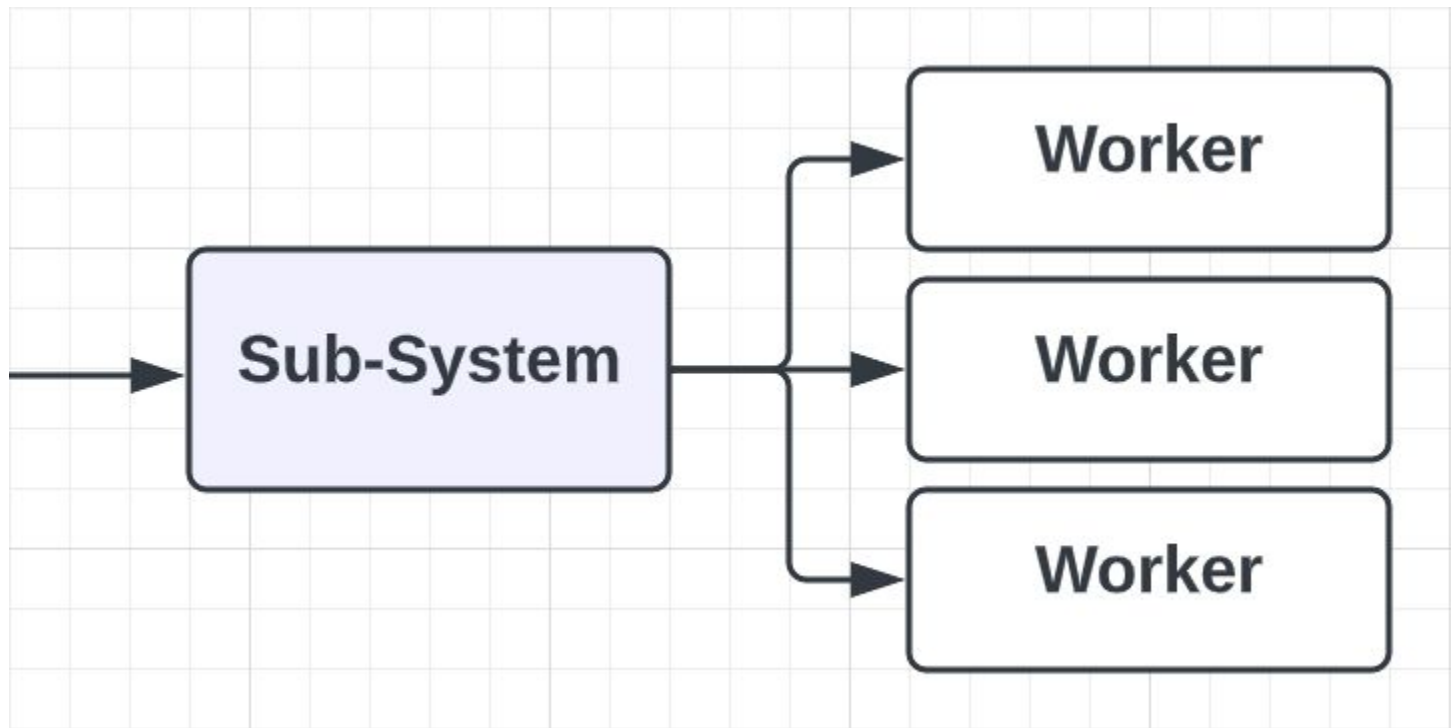
Arquiteturas Orientadas a Eventos

Modelo Assíncrono



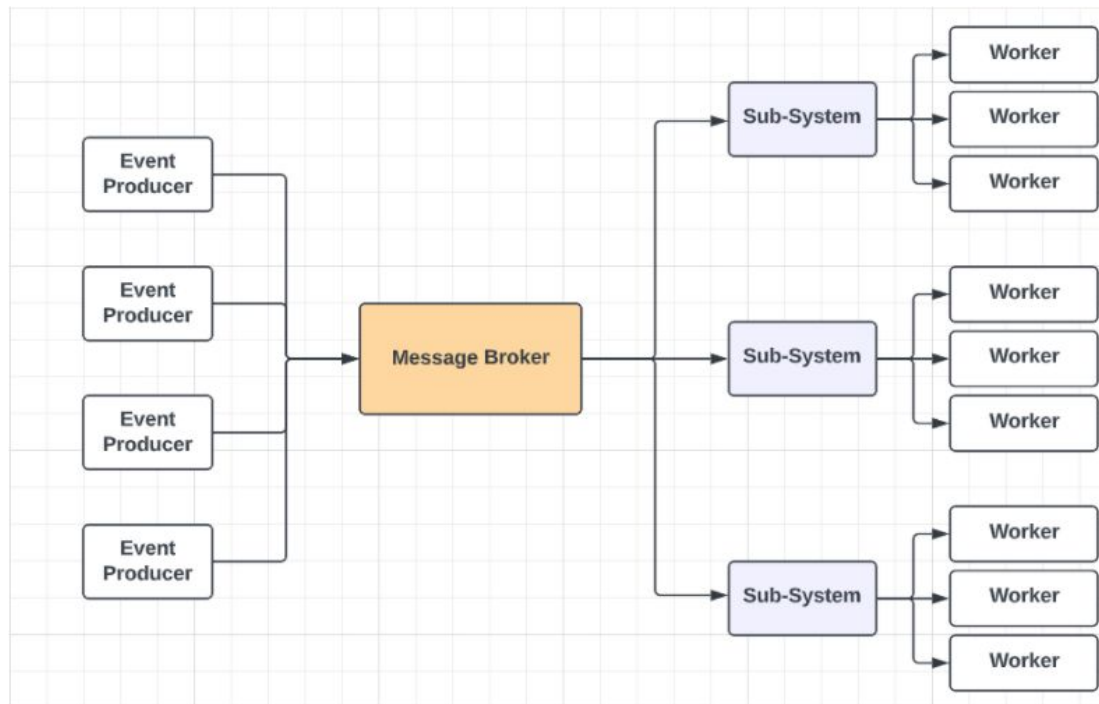
Arquiteturas Orientadas a Eventos

Arquiteturas Orientadas a Eventos



Arquiteturas Orientadas a Eventos

Arquiteturas Orientadas a Eventos



Arquiteturas Orientadas a Eventos

Redis

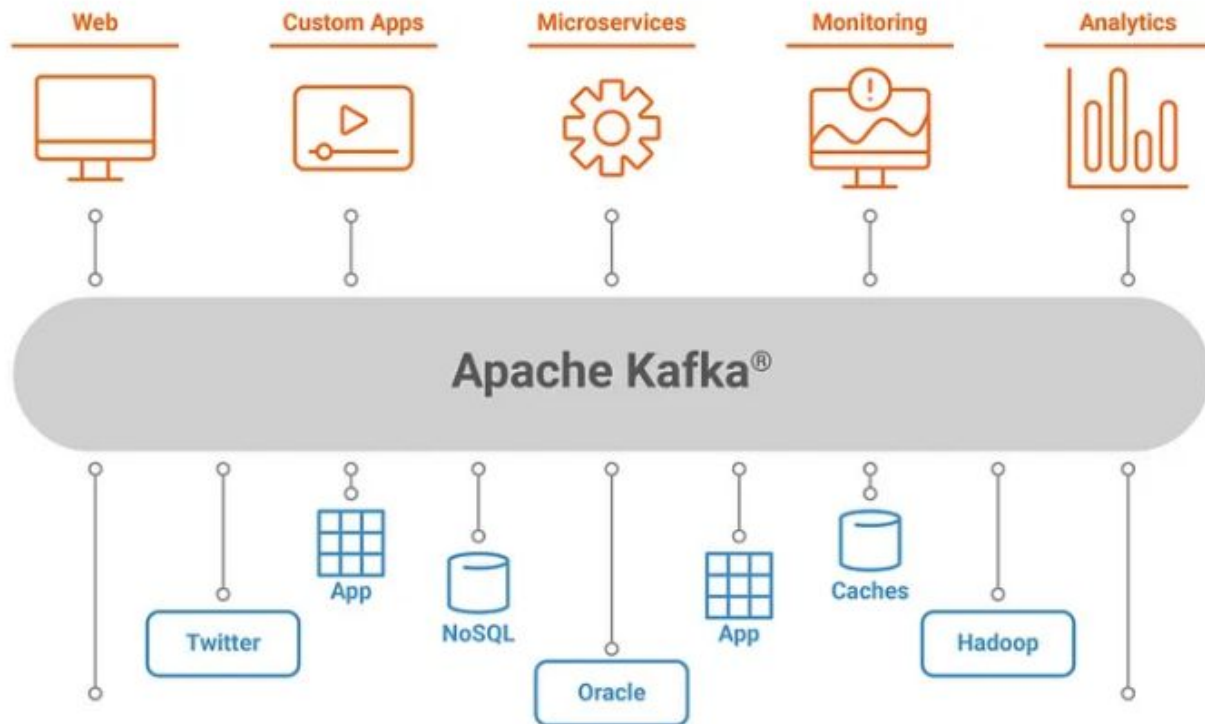
- Cache: “json em memória”
- Mensageria
- Acesso $O(1)$ em Strings
- Suporta TTL



redis

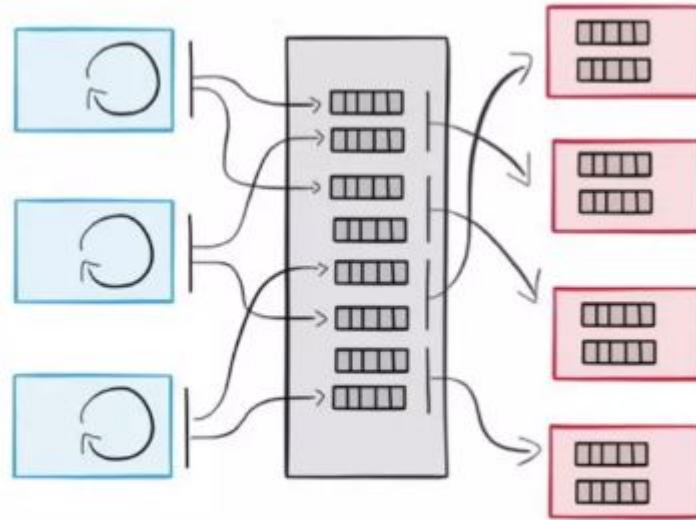
Arquiteturas Orientadas a Eventos

Kafka



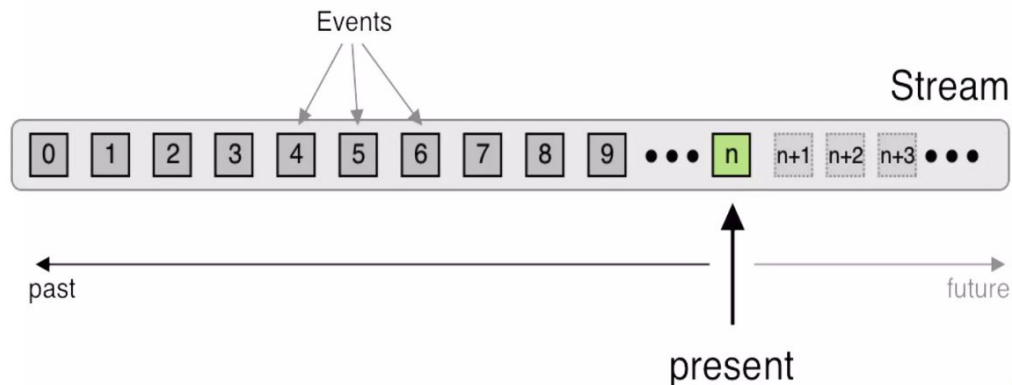
Arquiteturas Orientadas a Eventos

Kafka



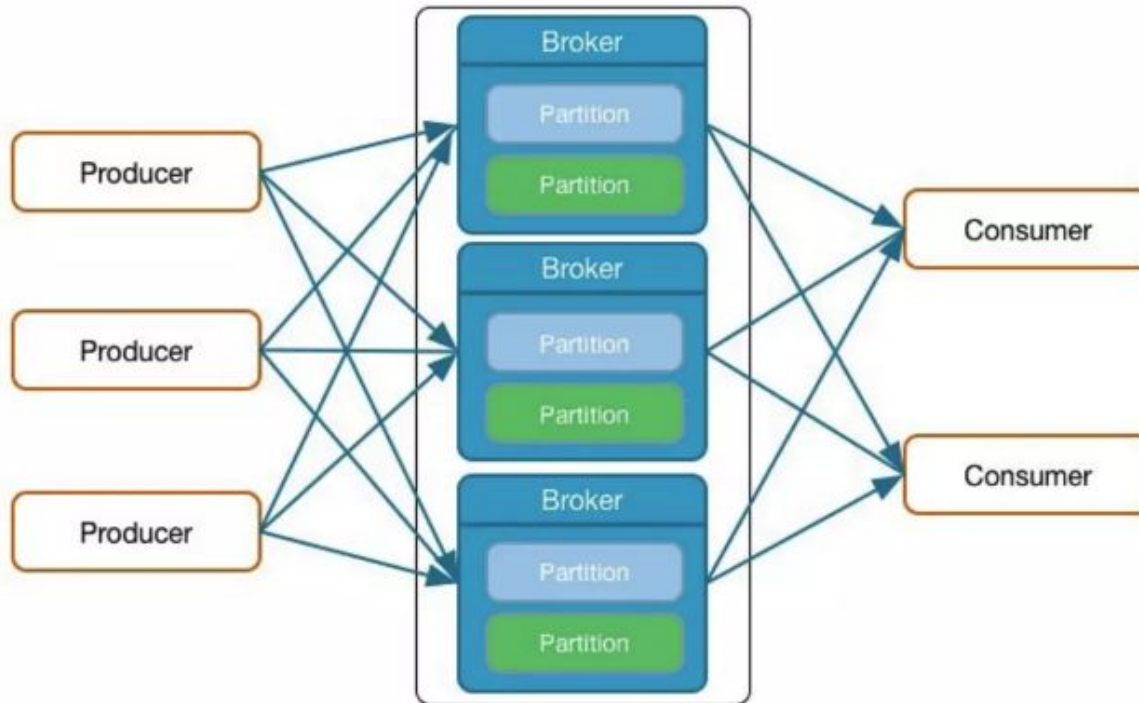
Arquiteturas Orientadas a Eventos

Kafka



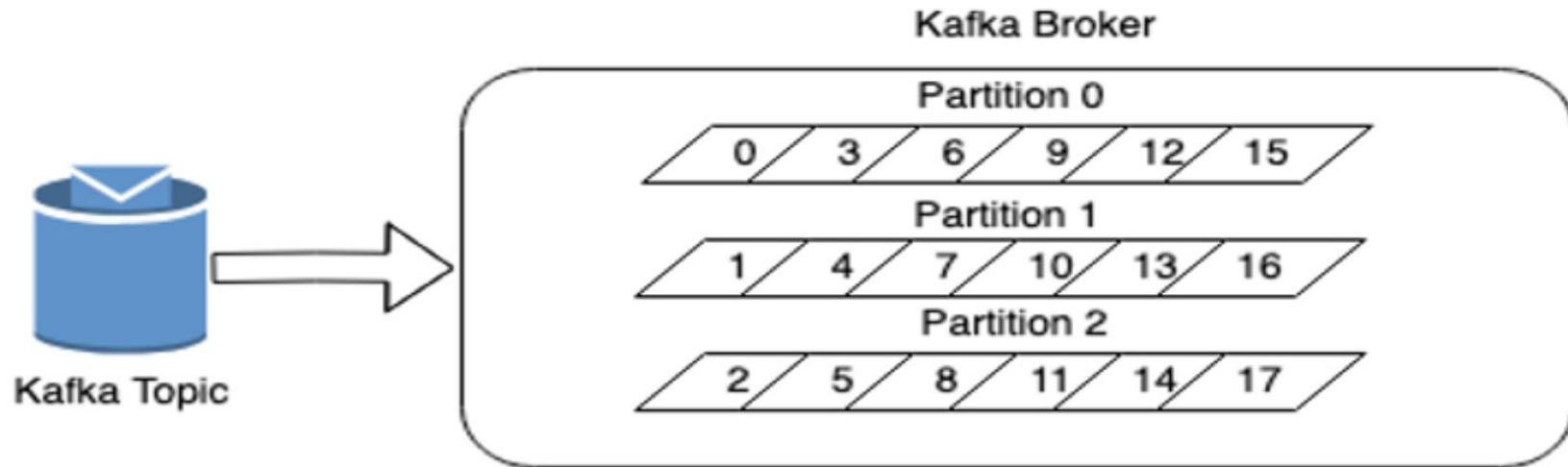
Arquiteturas Orientadas a Eventos

Kafka



Arquiteturas Orientadas a Eventos

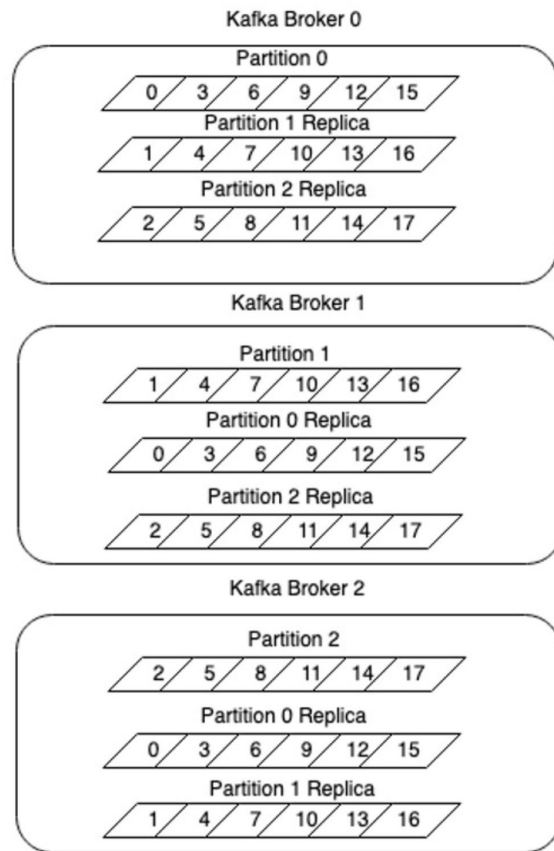
Kafka



Arquiteturas Orientadas a Eventos

Kafka

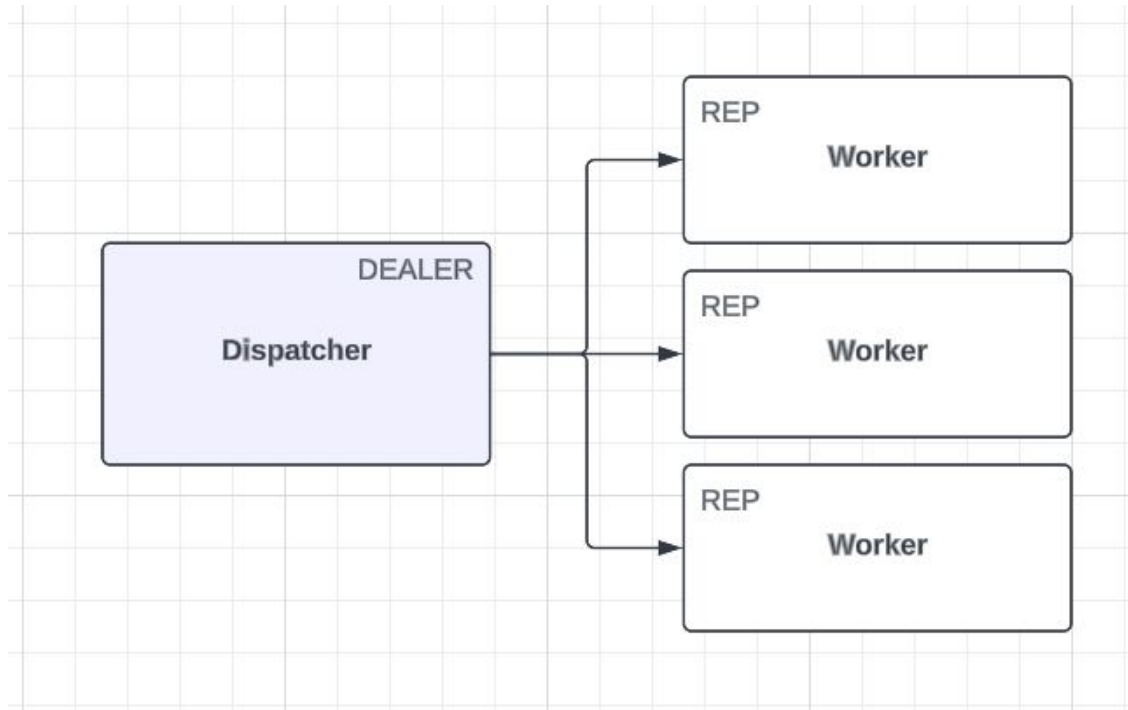
- Líderes das Partições distribuídas em Brokers (Escalabilidade Horizontal)
- Réplicas são Seguidores (Redundância)



Arquiteturas Orientadas a Eventos

ZMQ

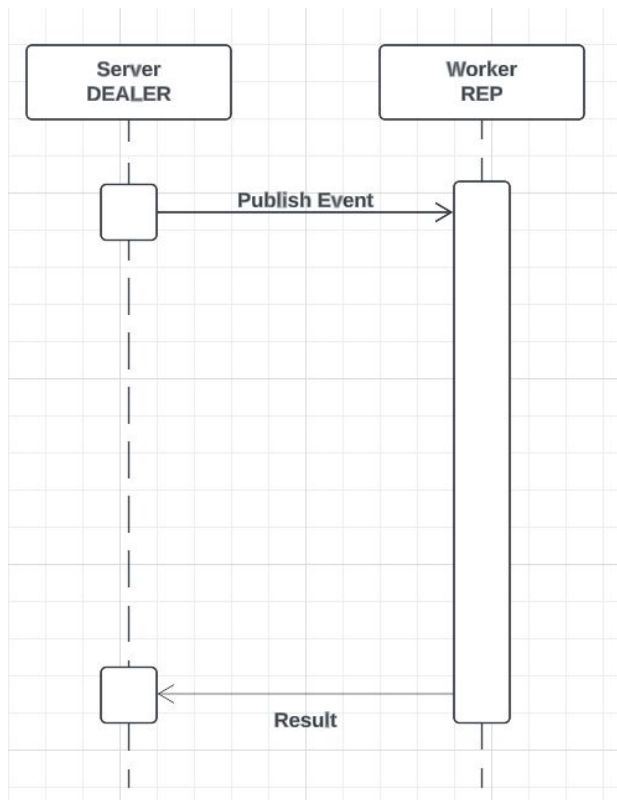
DEALER - REP



Arquiteturas Orientadas a Eventos

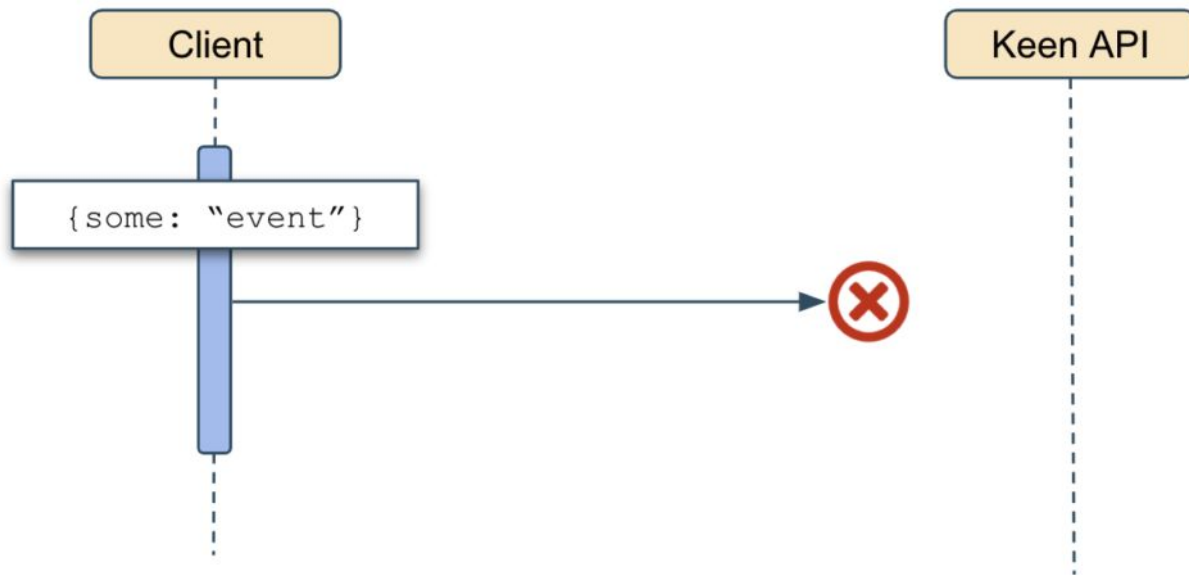
ZMQ

DEALER - REP



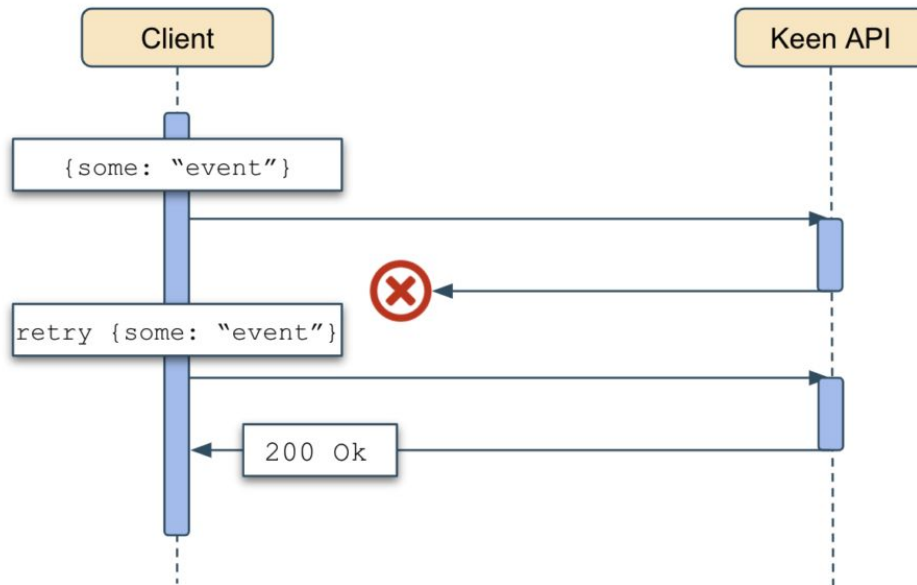
Arquiteturas Orientadas a Eventos

At-Most-Once Delivery



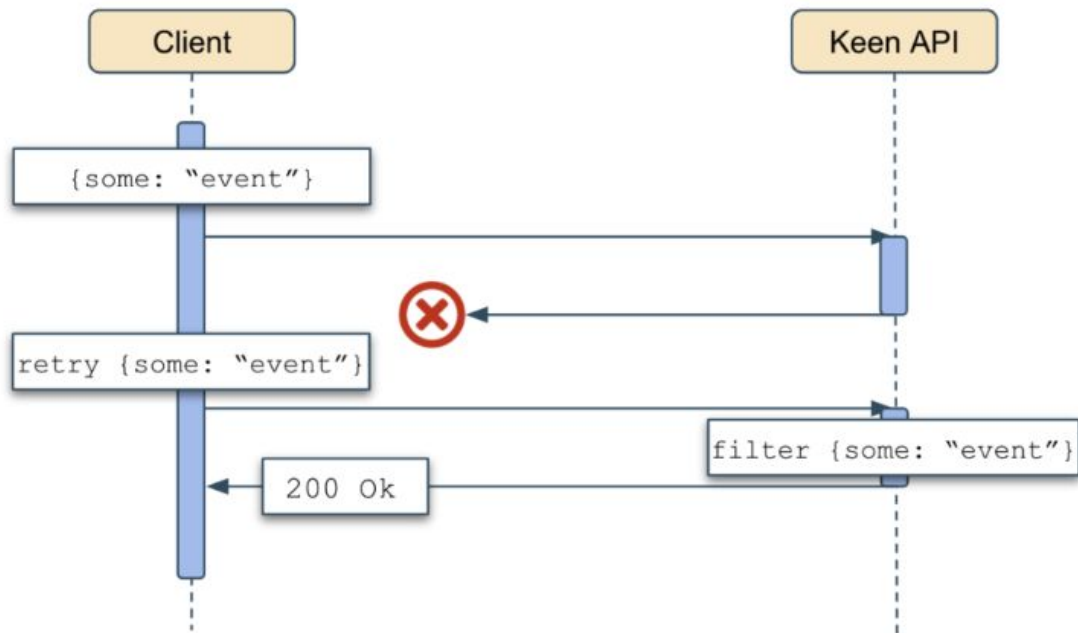
Arquiteturas Orientadas a Eventos

At-Least-Once Delivery



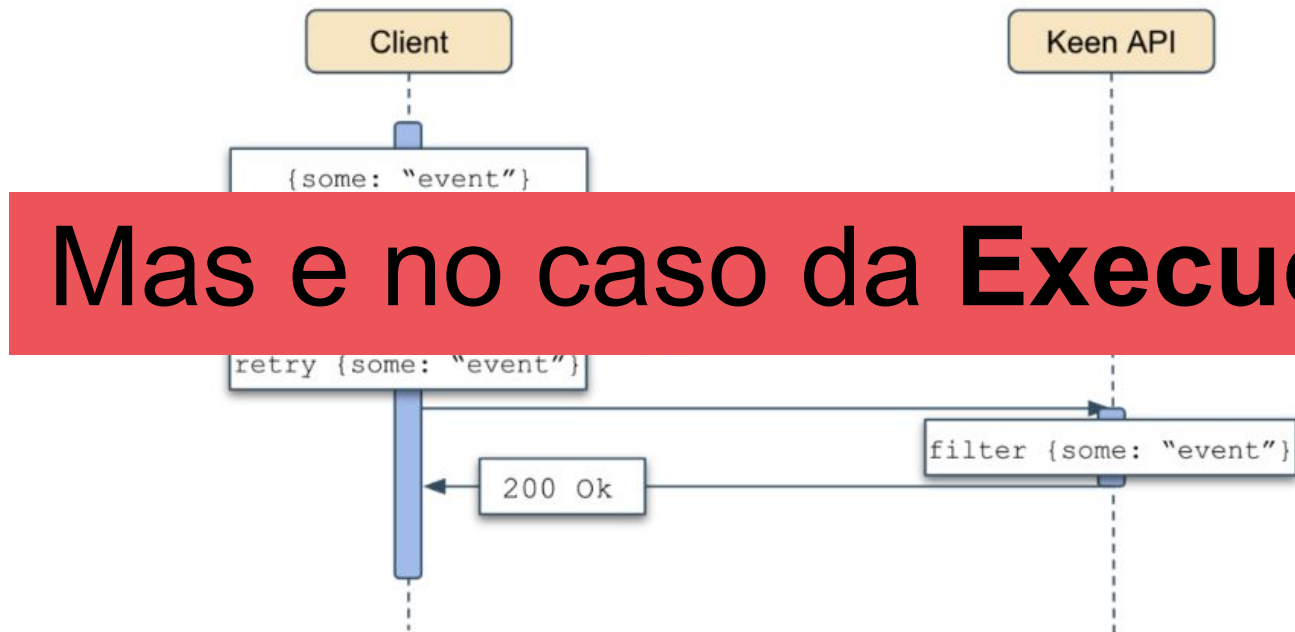
Arquiteturas Orientadas a Eventos

Exactly-Once Delivery



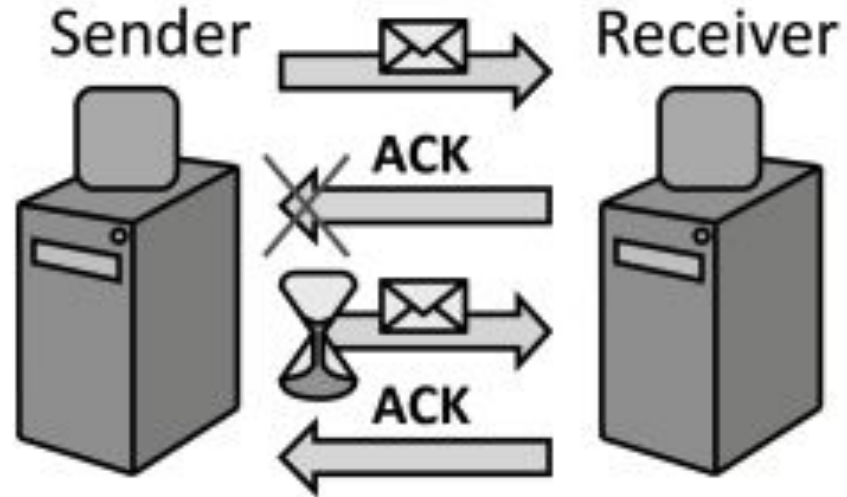
Arquiteturas Orientadas a Eventos

Exactly-Once Delivery



Arquiteturas Orientadas a Eventos

At-Least-Once Execution



Arquiteturas Orientadas a Eventos

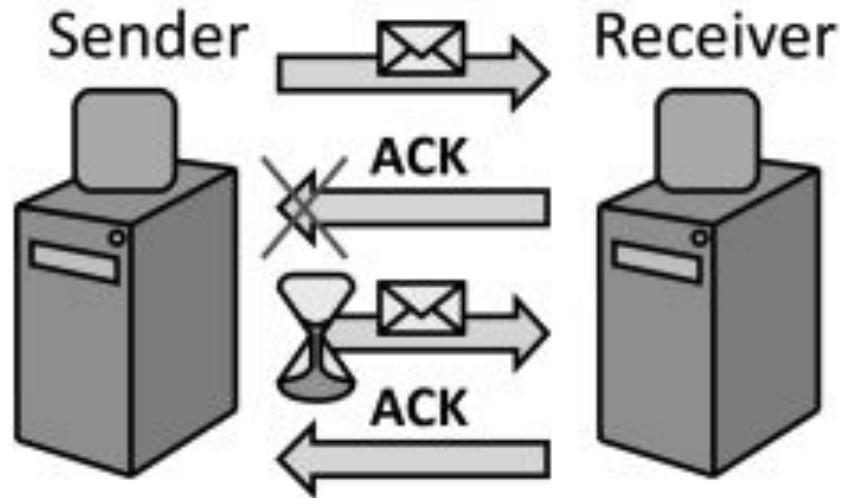
At-Least-Once Execution

E para nos aproximarmos de
Exactly-Once Execution?



+

Message ID (EPOCH number)



Arquiteturas Orientadas a Eventos

At-Least-Once Execution

E para nos aproximarmos de
Exactly-Once Execution?

```
Dispatcher

# Olhar Código 2/semantics/dealer
```

```
Worker

def main():
    m = receber_workload()

    if ja_executada(m.epoch):
        return m.epoch

    work(m)

    marcar_executada(m.epoch)

    return m.epoch
```


Arquiteturas Orientadas a Eventos

At-Least-Once Execution

E para nos aproximarmos de
Exactly-Once Execution?

```
Dispatcher

sent_msgs = []
msgs = []

def dispatcher_thread():
    while True:
        m = msgs.pop()
        send(m)
        sent_msgs.append(m)

def reciever_thread():
    while True:
        m = recieve()
        rcvd_msgs.append(m)

def comparer_thread():
    while True:
        for msg in sent_msgs:
            if msg.timeout:
                sent_msgs.remove(msg)
                msgs.append(msg)
```

Arquiteturas Orientadas a Eventos

At-Most-Once Execution

Garante **uma única** entrega de workload