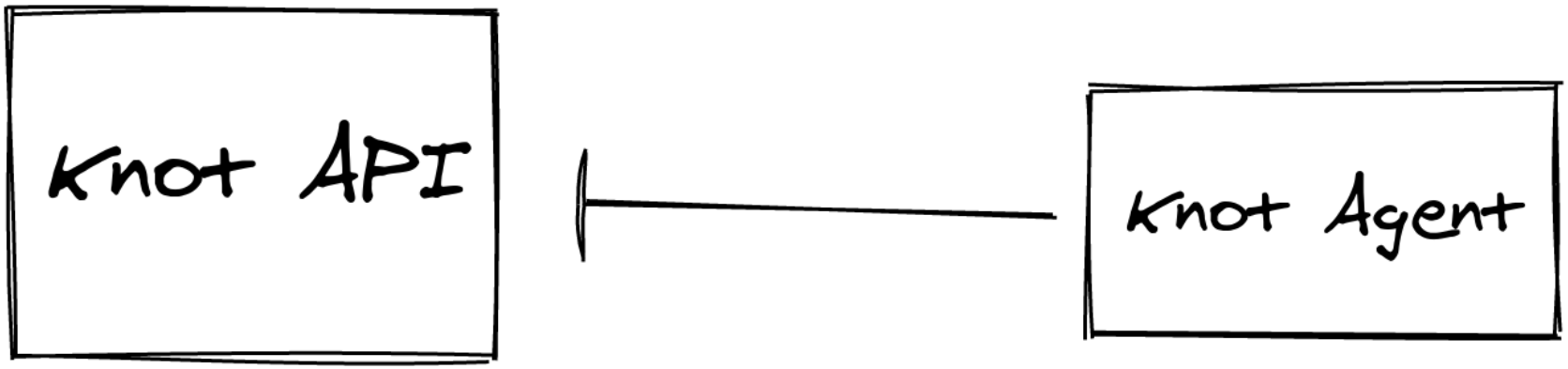
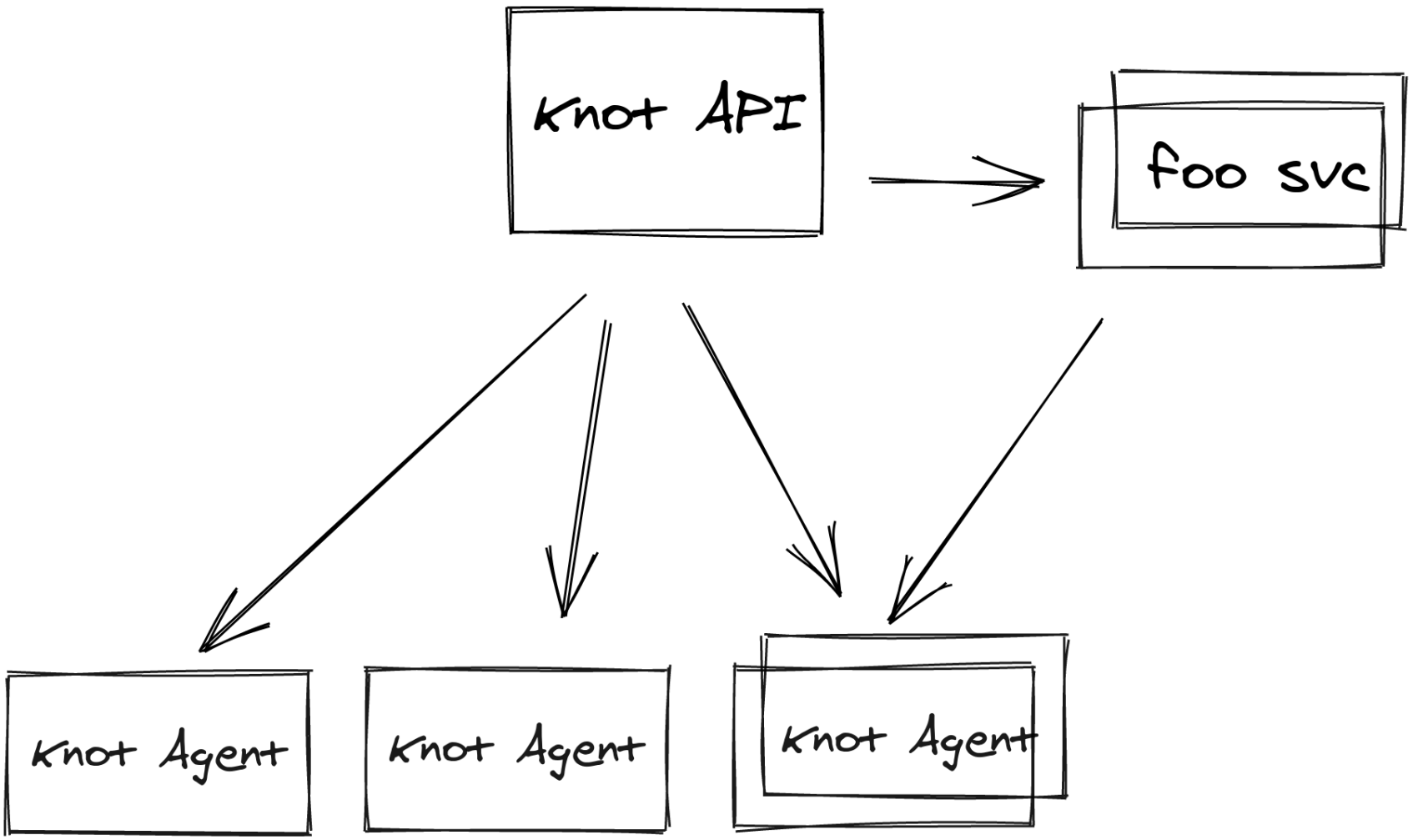
Kafka 101

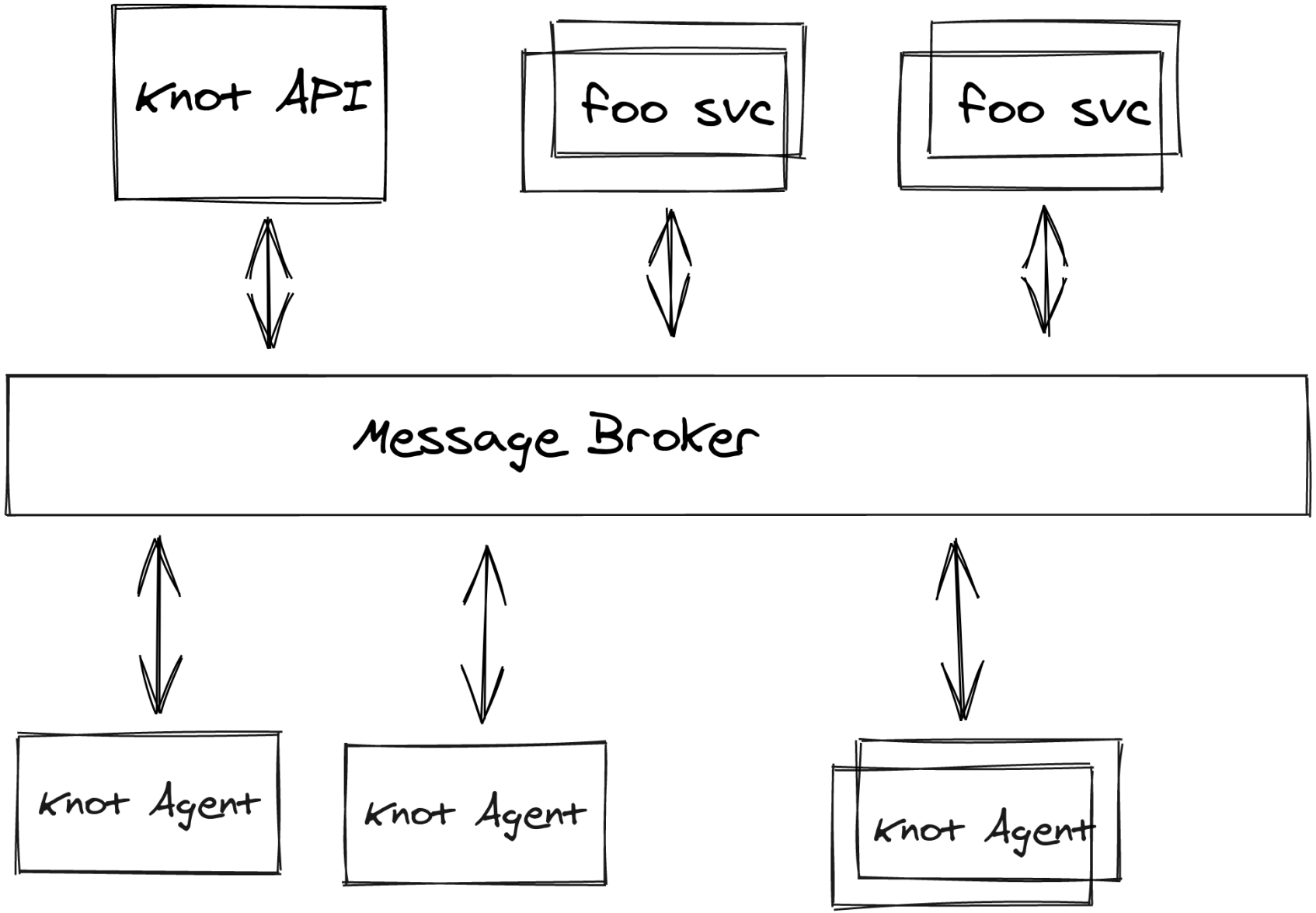
A gentle introduction

Azzam S.A

Why?




knot API
foo svc
knot Agent
knot Agent
knot Agent


Knot API
foo svc
foo svc
Message Broker
Knot Agent
Knot Agent
Knot Agent

Kafka vs RabbitMQ

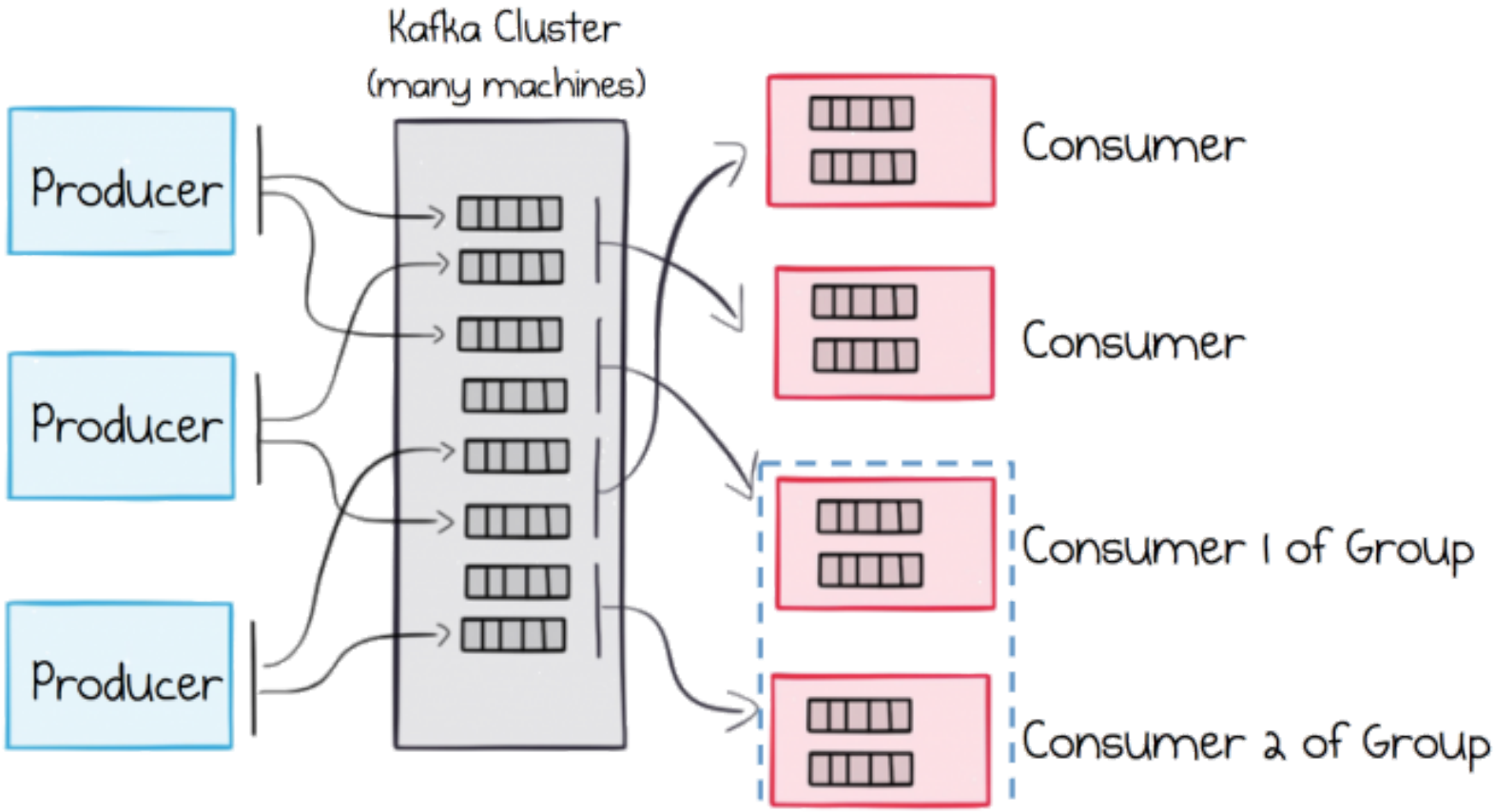
- Message ordering
- Message routing
- Message timing
- Message retention
- Fault handling
- Scaling
- Consumer complexity
  - RabbitMQ: smart-broker, dumb-consumer
  - Kafka: dump-broker, smart-consumer

When to use which

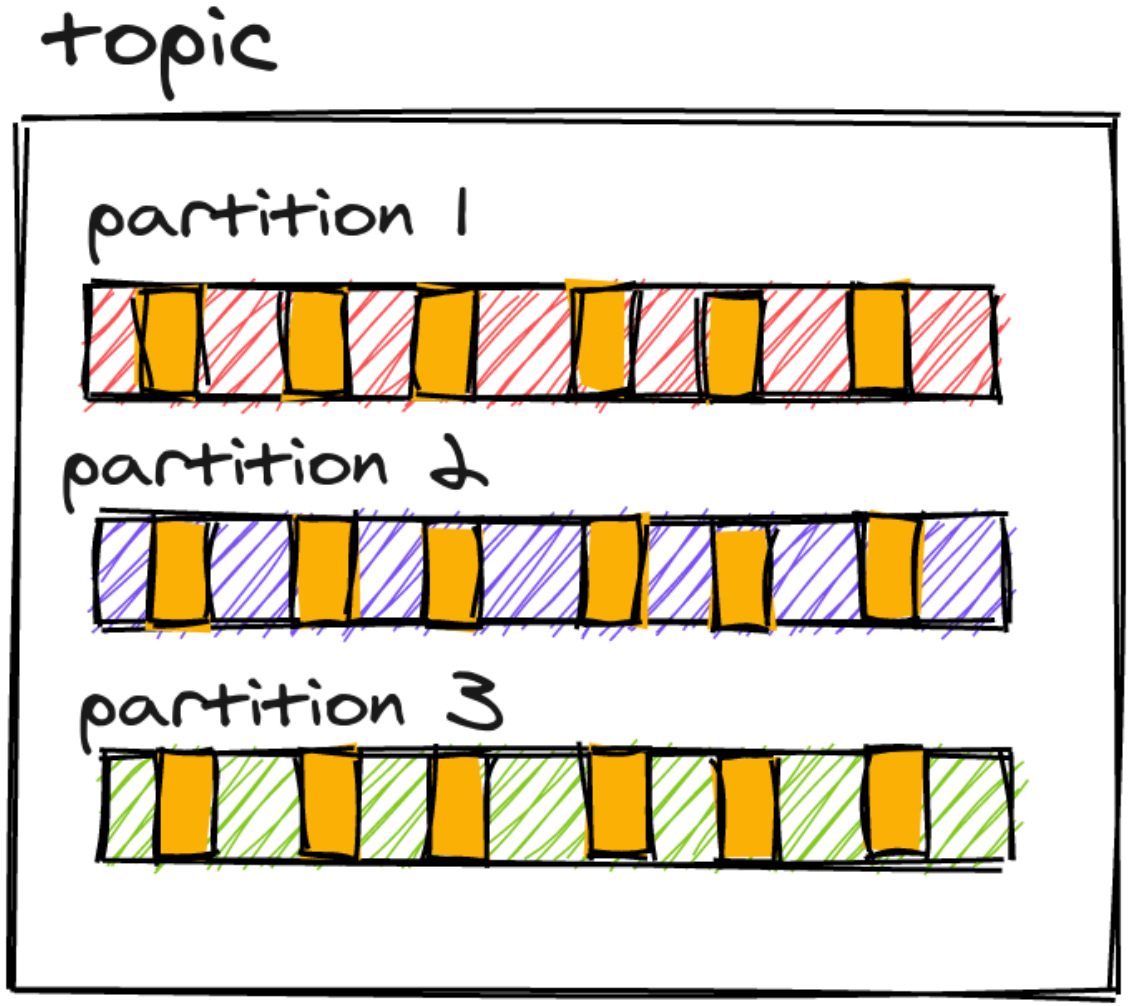
- RabbitMQ: Command
- Kafka: Event
  - Package tracking
  - Movie recommendation
  - Surge pricing

What?

Kafka Component - Bird View



confluent.io


topic
partition 1
partition 2
partition 3

Topic

- Replication
- Partition
- Offset
- Retention
- Immutable

How - Installation

YAML

```yaml
services:
    zookeeper:
        image: wurstmeister/zookeeper
    kafka1:
        image: wurstmeister/kafka:2.12-2.4.0
        ports:
            - "9092:9092"
        environment:
            KAFKA_ADVERTISED_HOST_NAME: 103.93.59.207
            KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181
```

Creating topics

BASH

```
#!/usr/bin/env bash

# kafka-topics is alias to kafka-topics.sh --bootstrap-
server localhost:9092

kafka-topics --list
kafka-topics --create --topic bus-location # --
replication-factor 1 --partitions 2
kafka-topics --describe --topic bus-location
```

Producing messages

BASH

```
#!/usr/bin/env bash

# kafka-producer is alias to kafka-console-producer.sh --
bootstrap-server localhost:9092

kafka-producer --topic bus-location
```

Consuming messages

BASH

```
#!/usr/bin/env bash

# kafka-consumer is alias to kafka-console-consumer.sh --
bootstrap-server localhost:9092

kafka-consumer --topic bus-location # --from-beginning
```

Consumer

- Consumer Group
- Load balance
- Reliability

More

BASH

```
#!/usr/bin/env bash

kafka-topics --create --topic bus-fuel --partitions 2

kafka-consumer --topic bus-fuel --group traffic # twice
```

Python

PYTHON

```python
from kafka import KafkaProducer
from kafka import KafkaConsumer

# Producer
producer =
KafkaProducer(bootstrap_servers="localhost:9092")

for count in range(100):
    producer.send("bus-location", count)

# Consumer
consumer = KafkaConsumer("bus-location")

for msg in consumer:
    print(msg)
```

> *Paths are made by walking.*
>
> *— Franz Kafka*

Thank You!

twitter @azzamsyawqi

www azzamsa.com

Biznet®
GioCloud

Credits

- excalidraw.com