APACHE KAFKA

Introdução

O QUE É O APACHE KAFKA

Open Source
Sistema distribuído
Stream de dados

O QUE ELE FAZ

Central de transferência e armazenamento de dados

Importancia

EVENTOS

Presentes na maiora dos sistemas nos dias de hoje

VANTAGENS

Manipulação e armazenamento historico de dados

Como ele funciona?

KLUSTER

Conjunto de servidores funcionando em sincronia e cooperando entre sí.

BROKERS

Servidores que armazenam os dados e partições

PRODUTOR E CONSUMIDOR

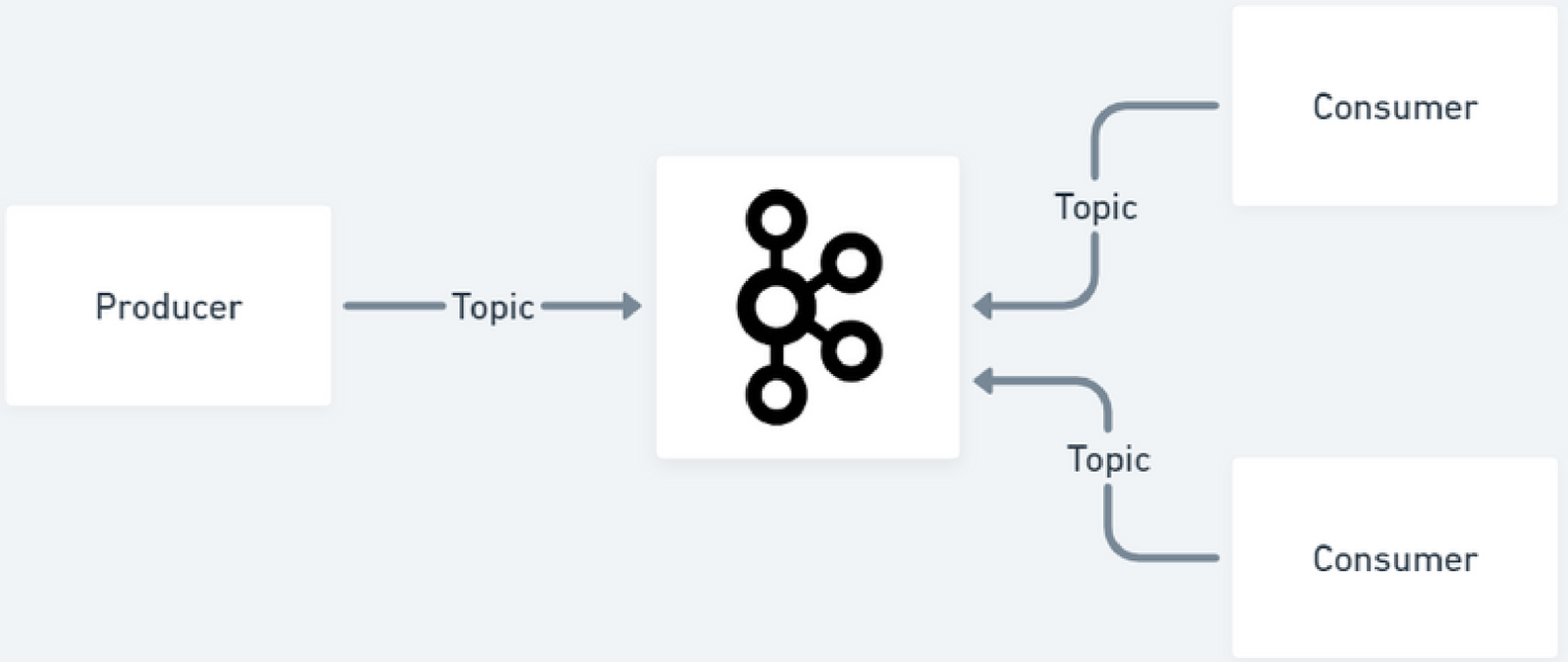
Produtores geram dados e postam nos tópicos, consumidores leem constantemente o tópico esperando por mensagens.

ZOOKEEPER

Possuem a função de service discovery e oquestragem de brokers, garantindo sua sincronicidade e cooperação.

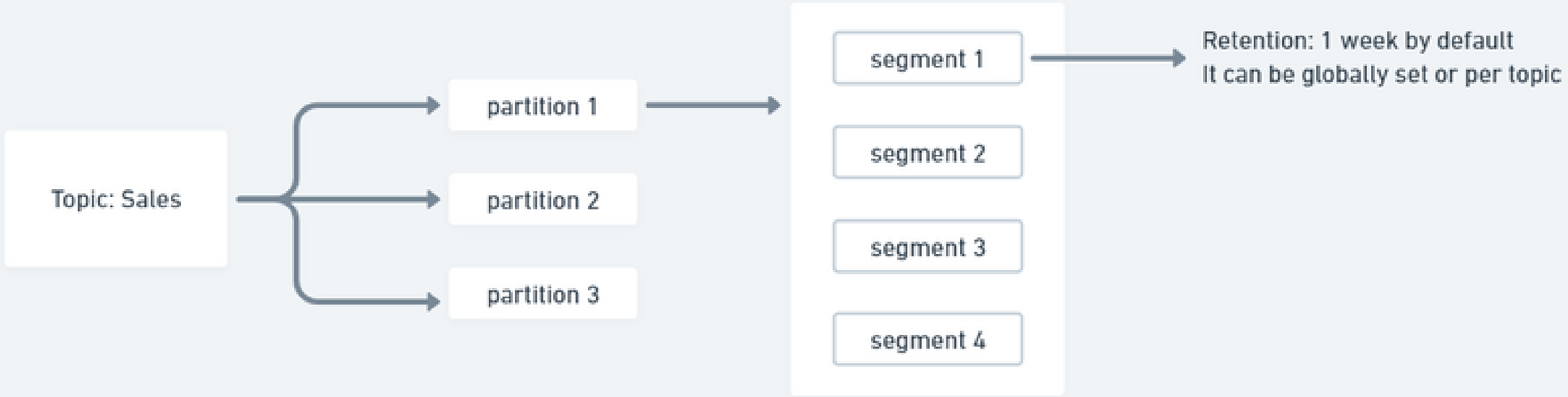
Tópicos

Conceito geral



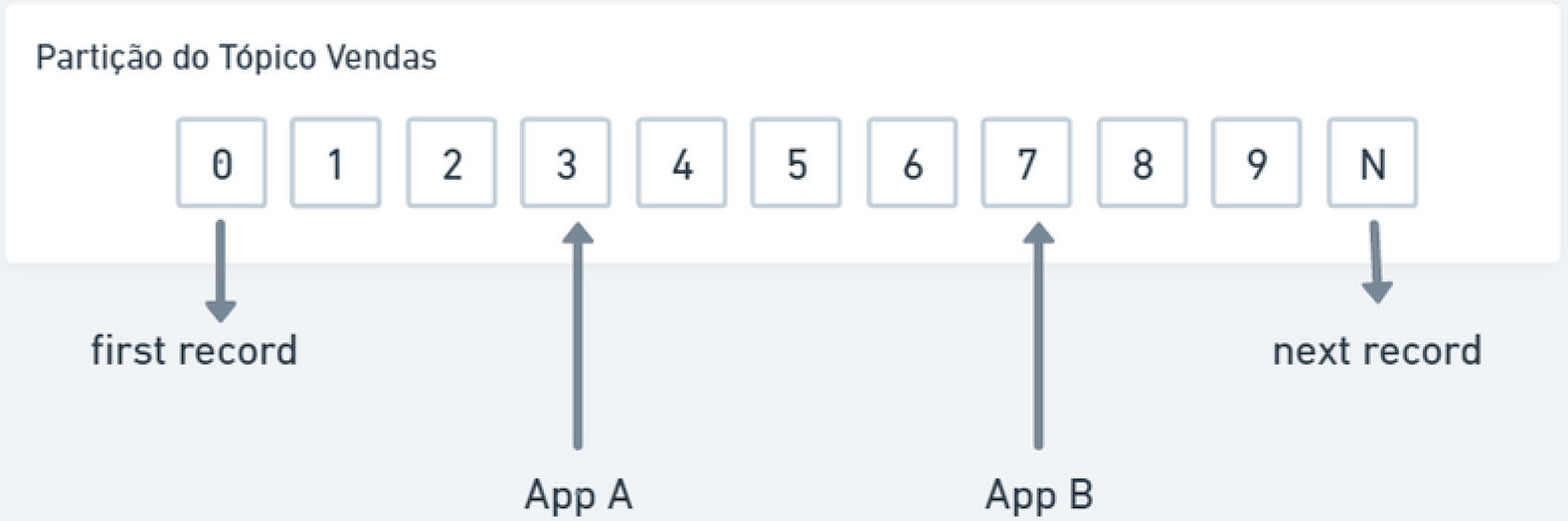
Tópicos

Partições e segmentos



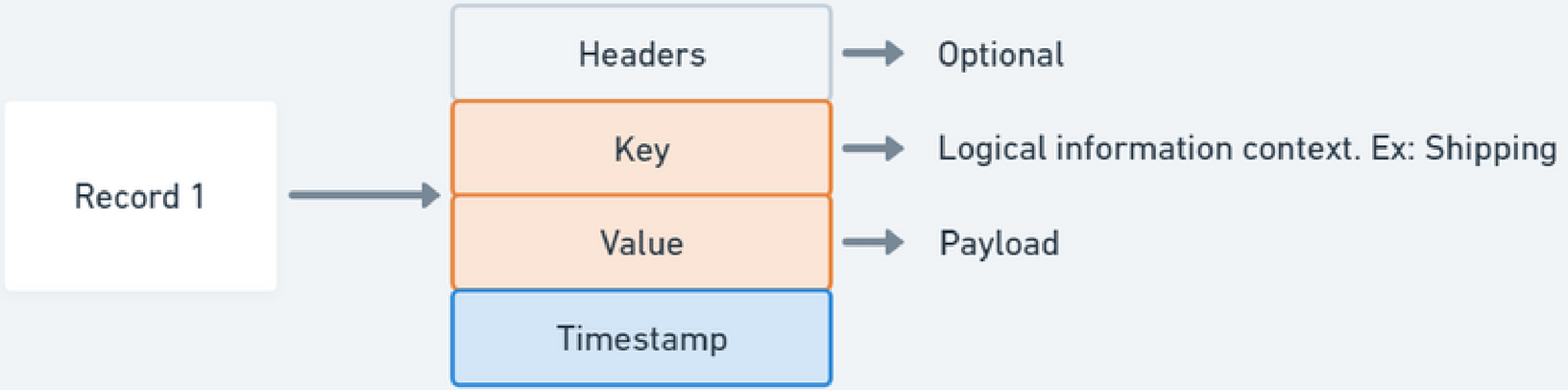
Tópicos

Logs



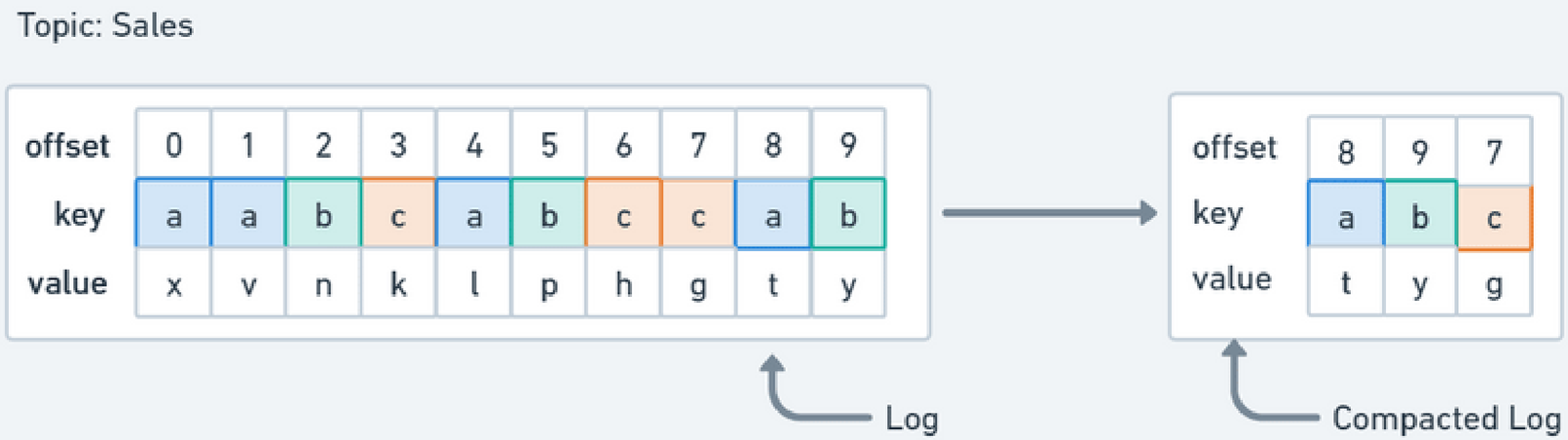
Tópicos

Anatomia dos offsets



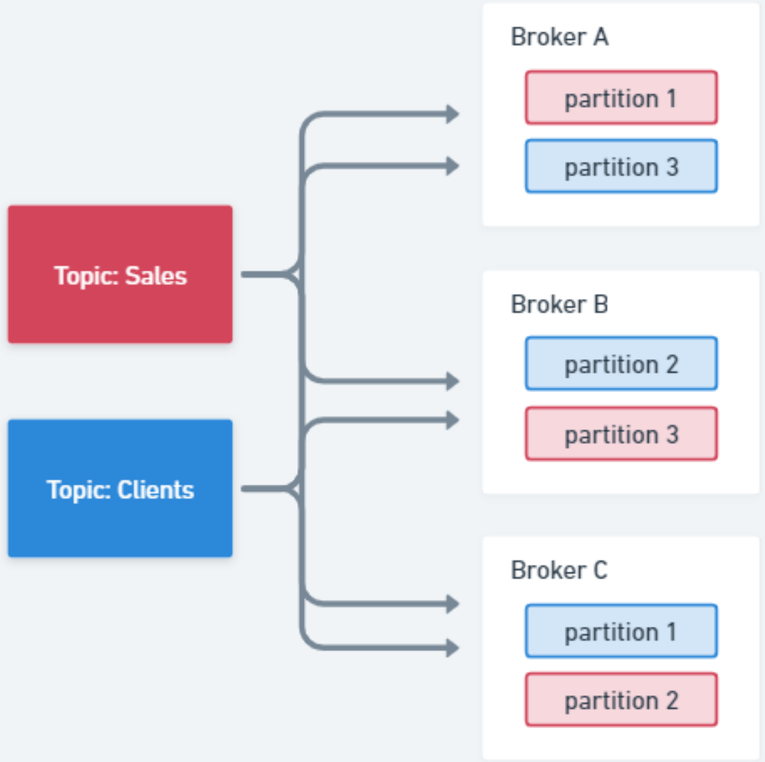
Tópicos

Tópico compacto



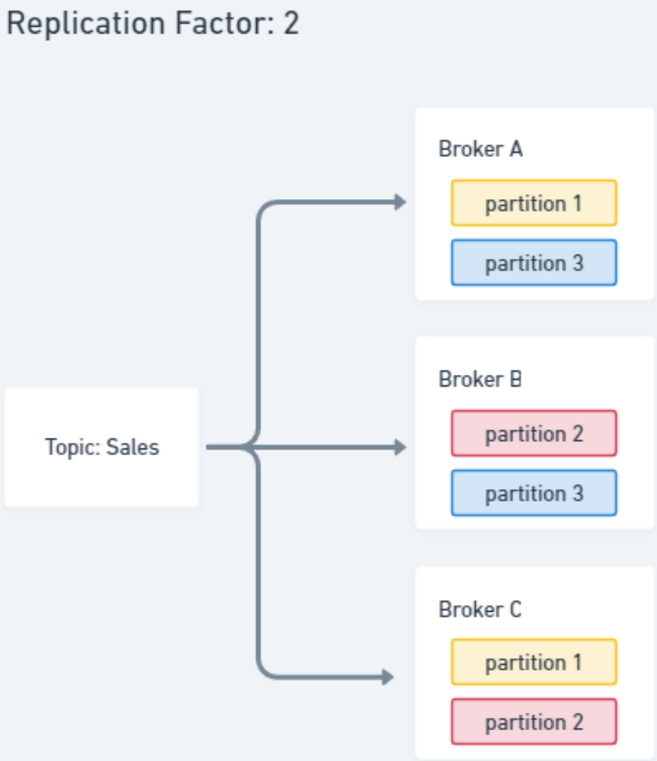
Tópicos

Partições Distribuidas



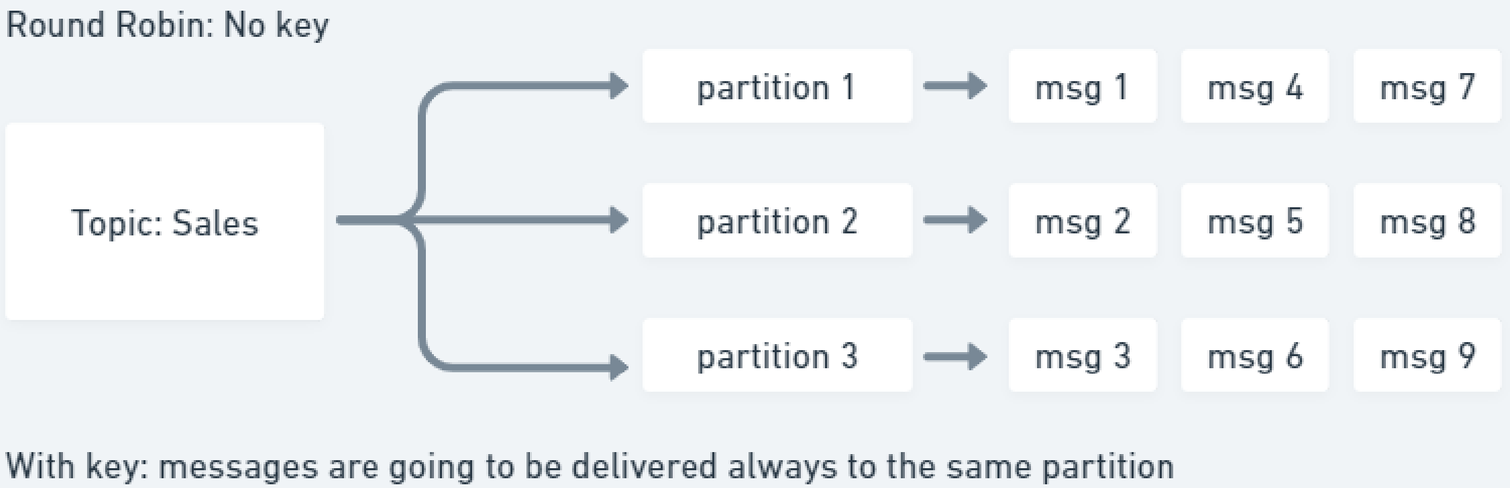
Tópicos

Replication Factor



Tópicos

Entrega de mensagens



With key: messages are going to be delivered always to the same partition

Tópicos

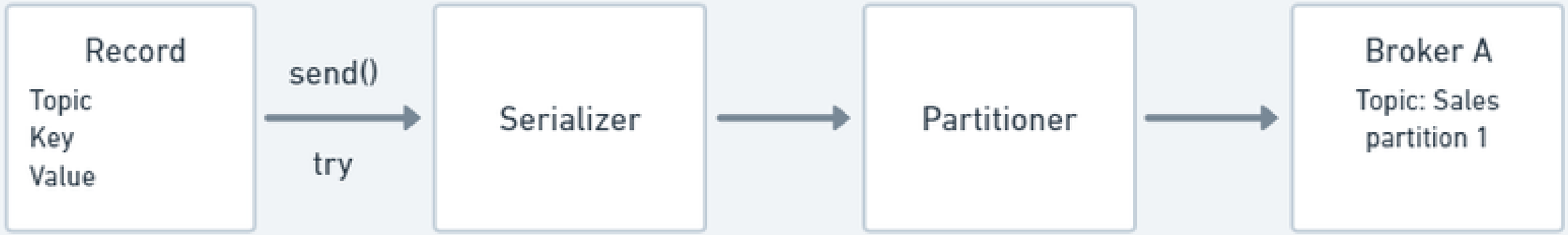
Partições Líder

Leaders = Blue
Followers = White

If there is any trouble with a broker running a leader partition, a new leader is elected.

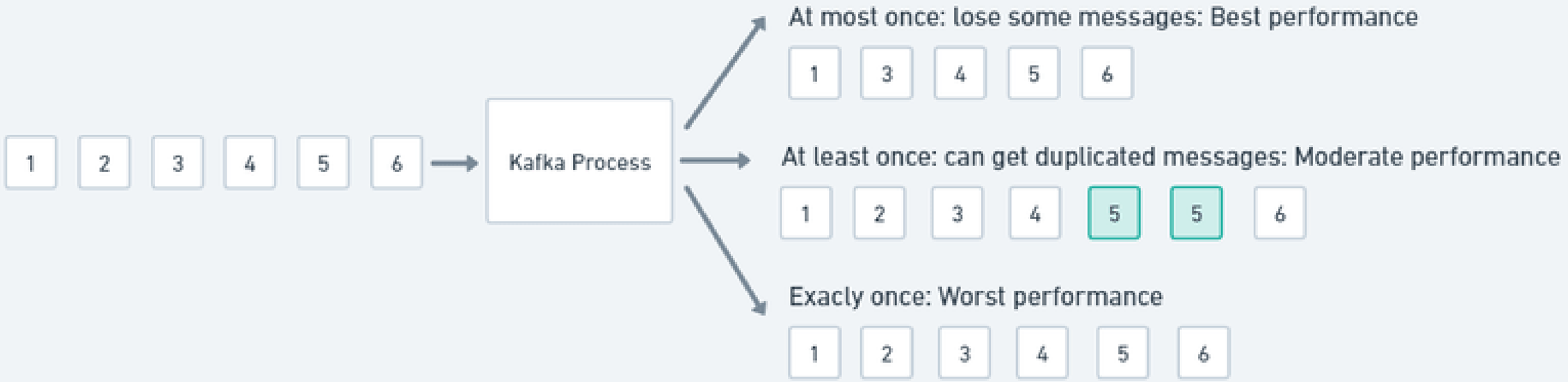
Produtores

Processo de entrega

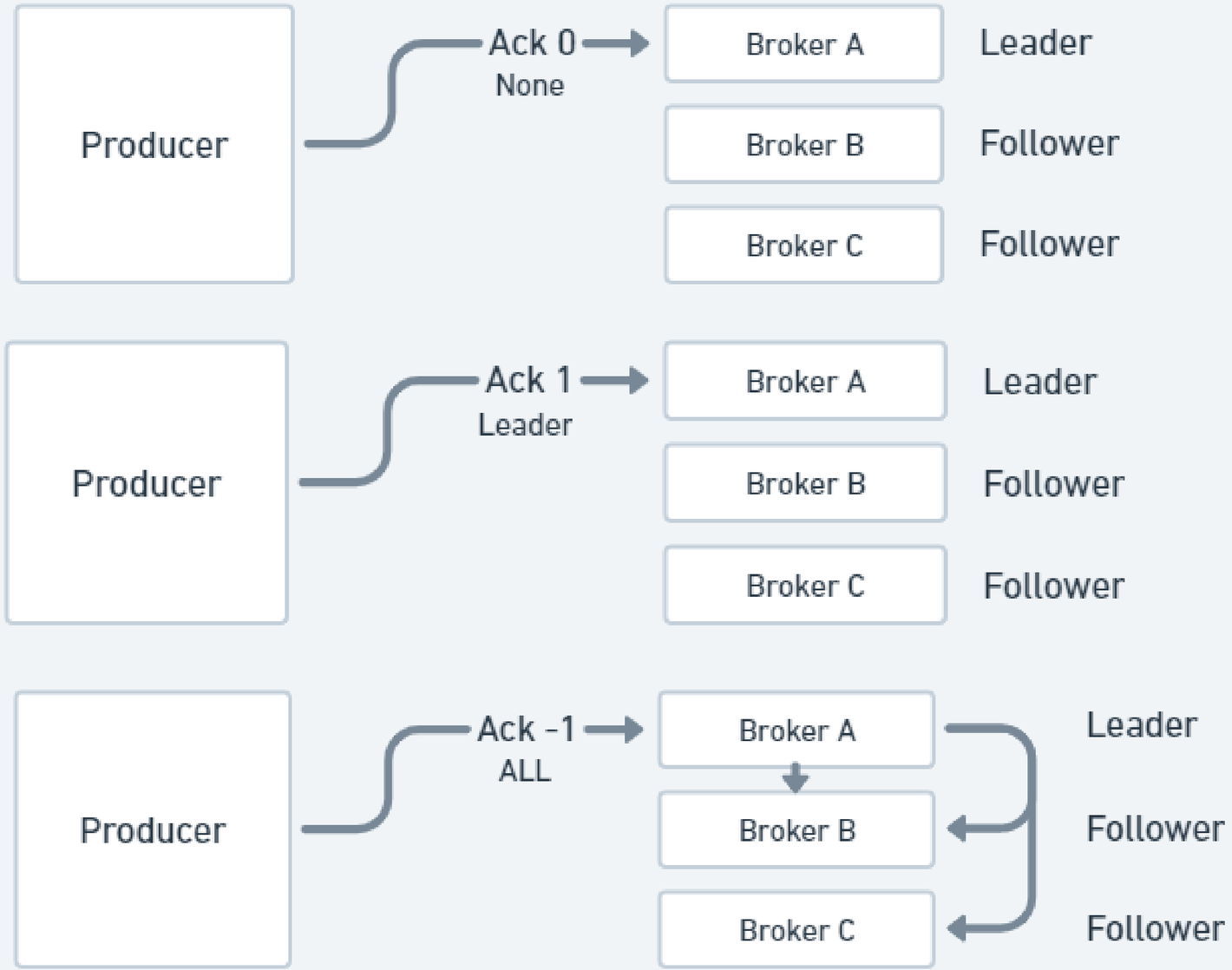


Produtores

Garantia de entrega



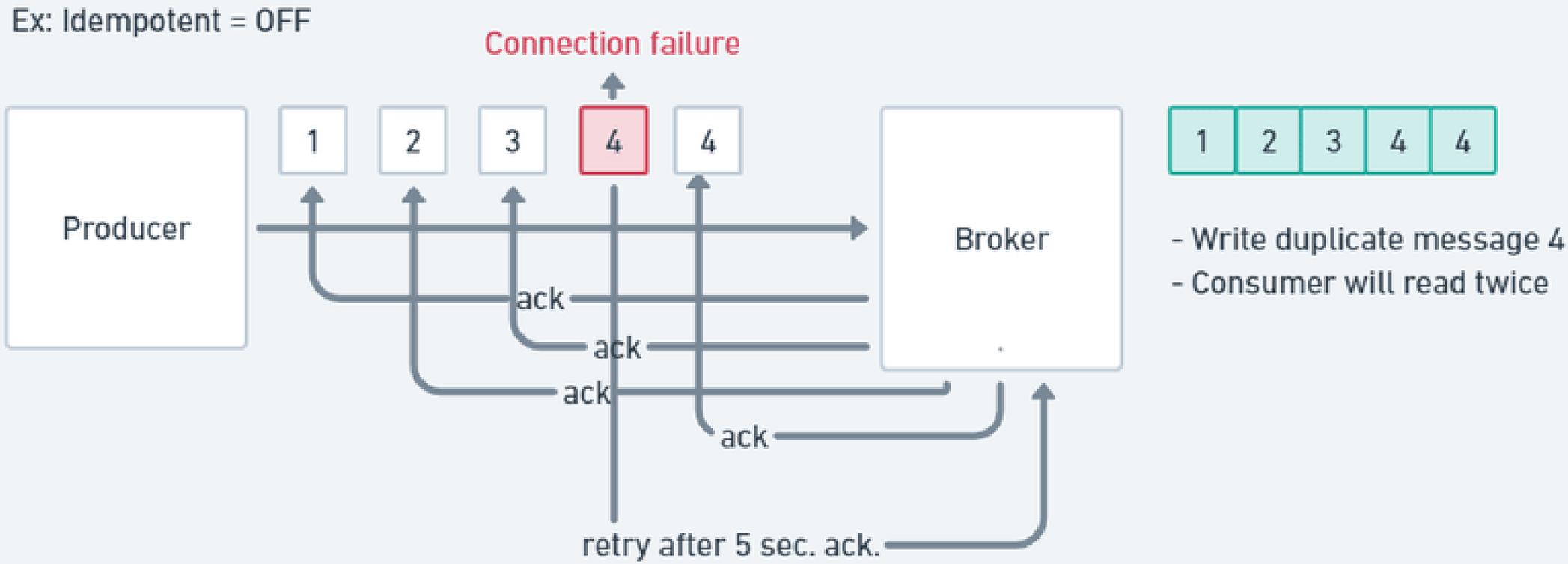
Produtores



Garantia de entrega - ACK

Produtores

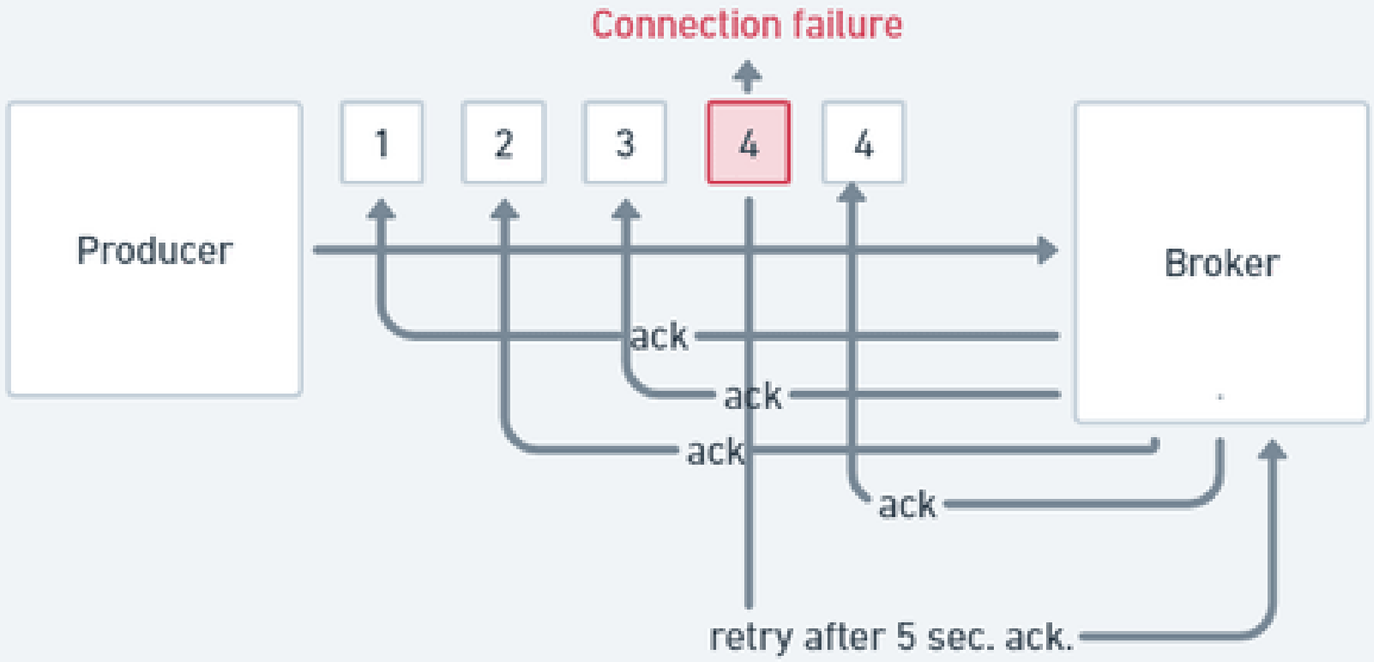
Idempotencia - OFF



Produtores

Idempotencia - ON

Ex: Idempotent = ON



- Check Producer ID and message sequence number.
- Discard if duplicate.
- Message is written in correct order

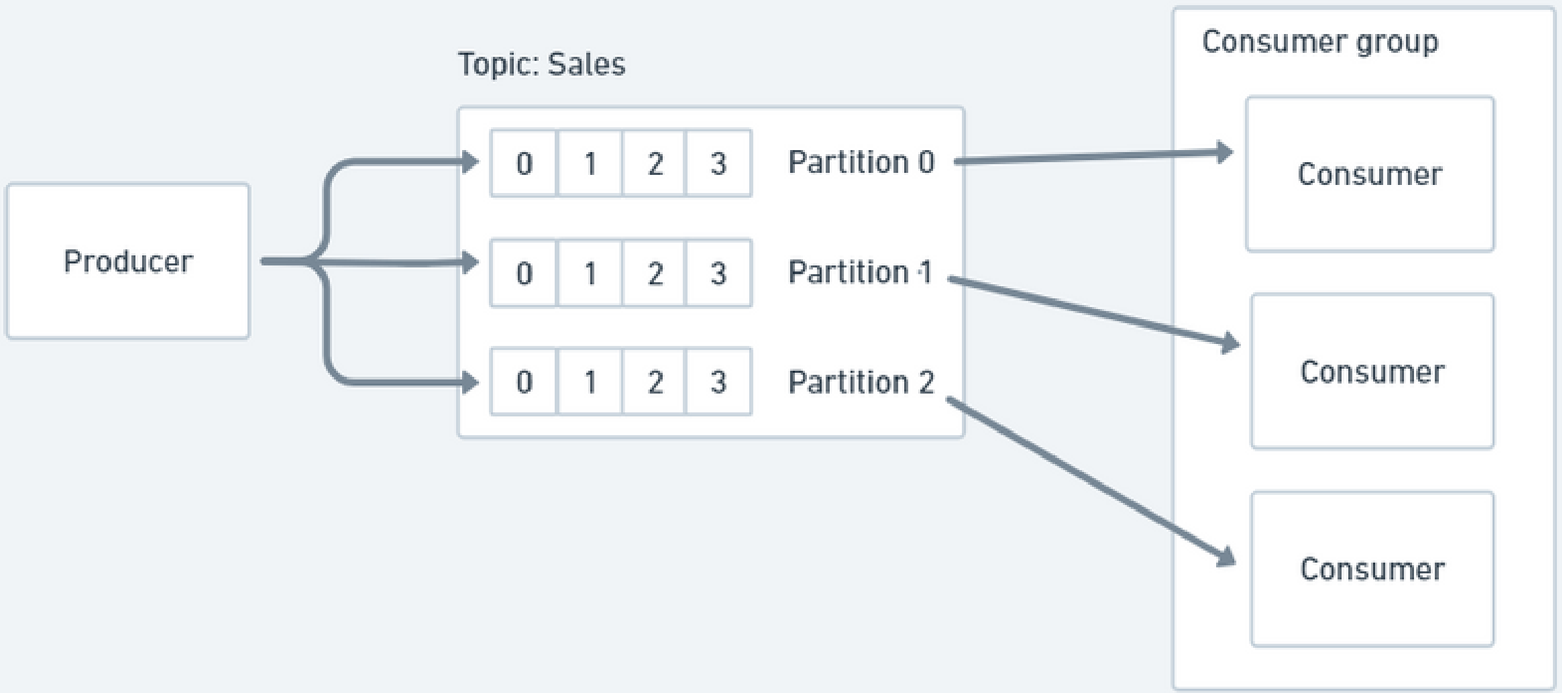
Consumidores

Consumidores

Grupo de consumidores

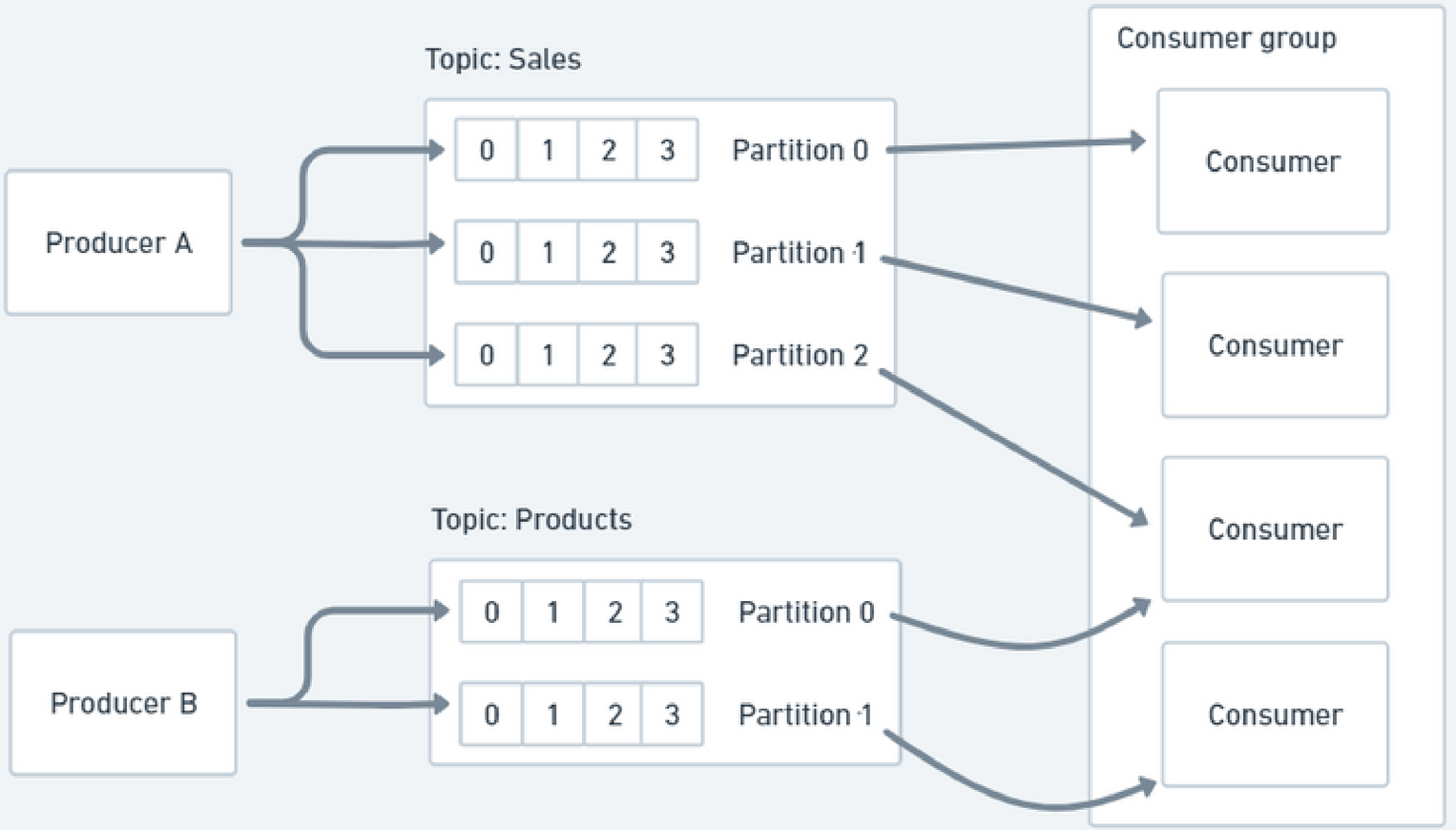
Consumidores

Grupo de consumidores



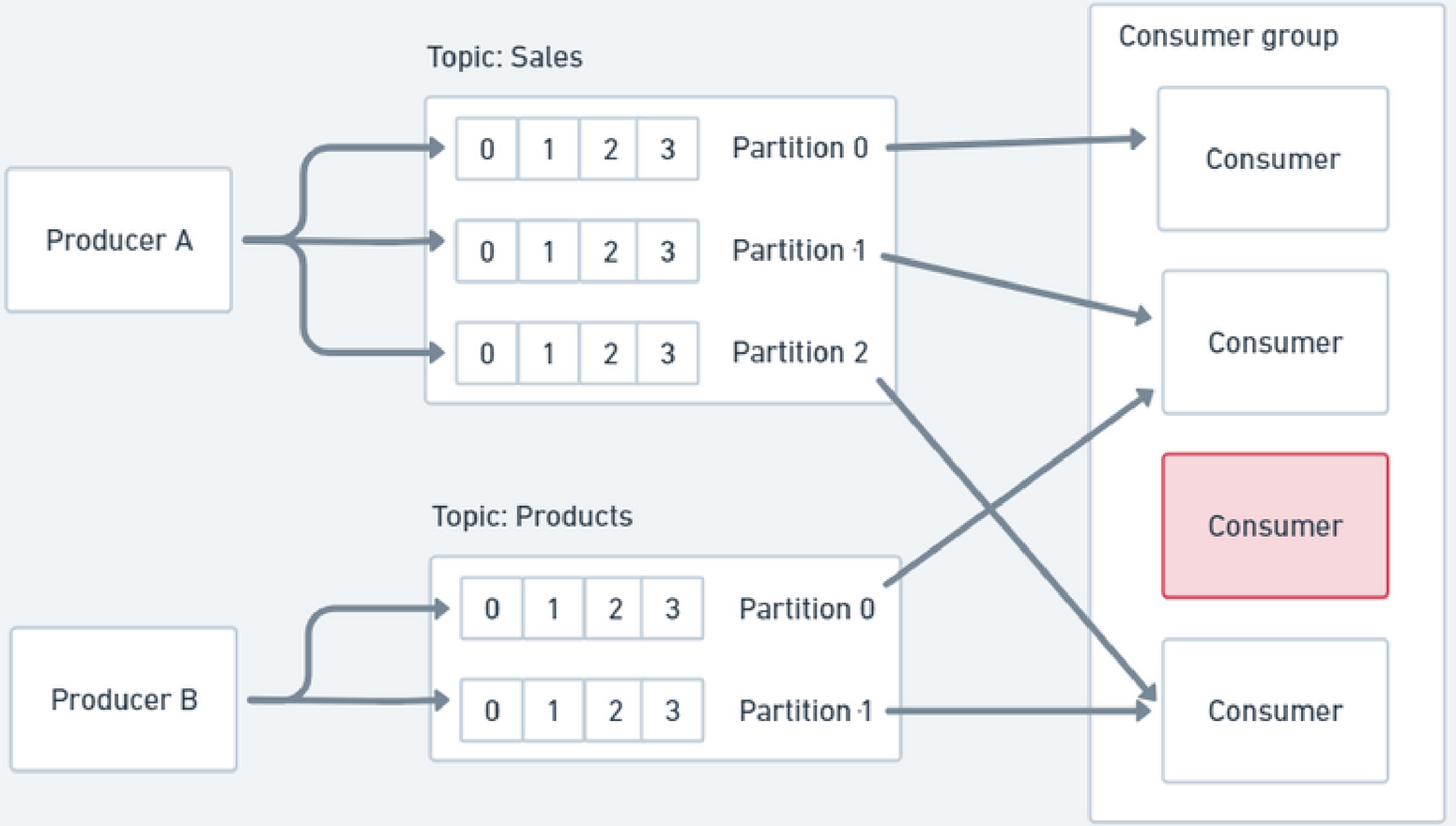
Consumidores

Grupo de consumidores



Consumidores

Reequilibrio de Consumidores

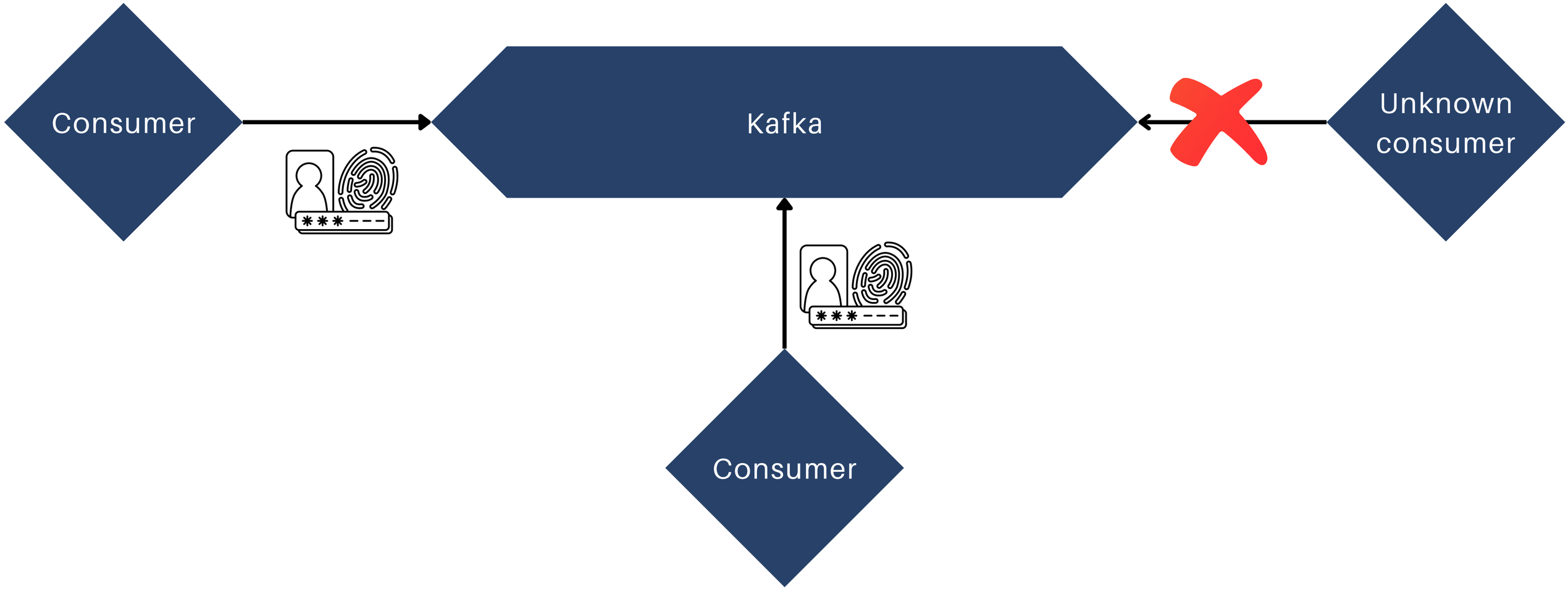


Segurança

Criptográfia

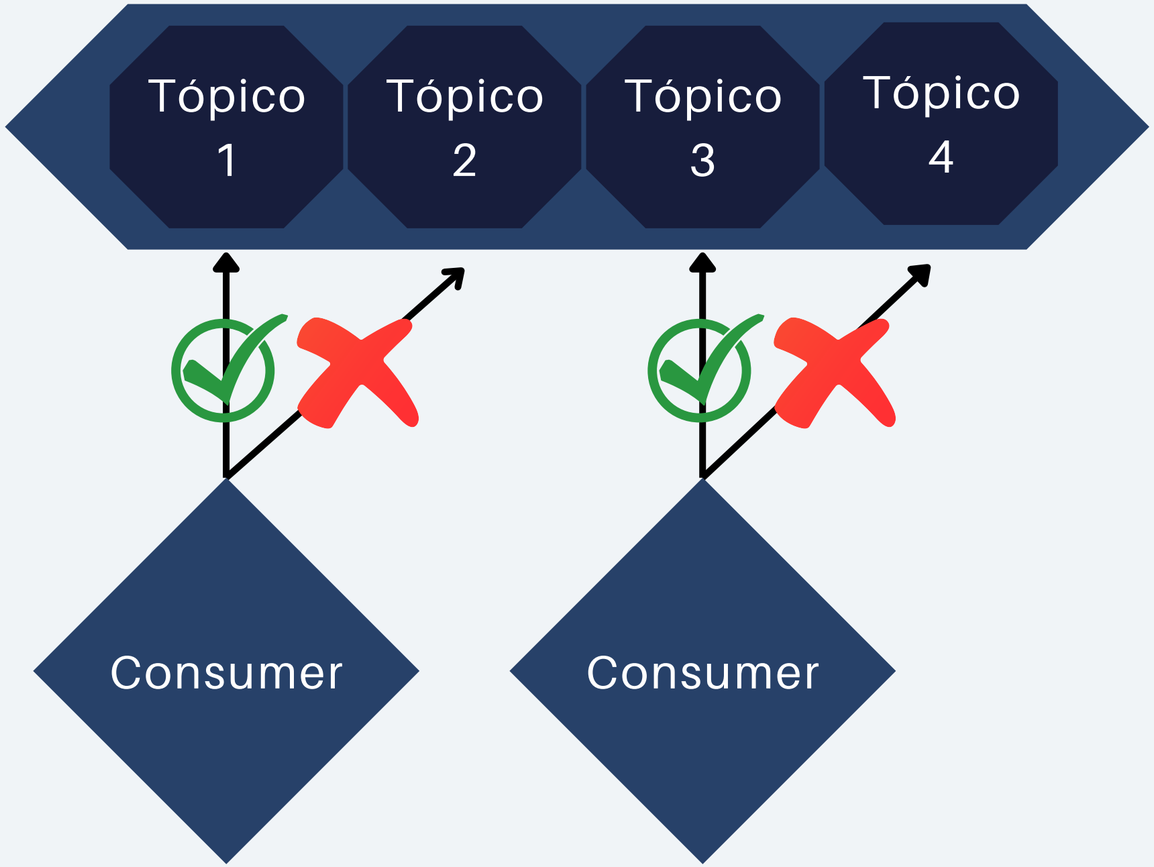
Segurança

AAA - Autentication



Segurança

AAA - Authorization



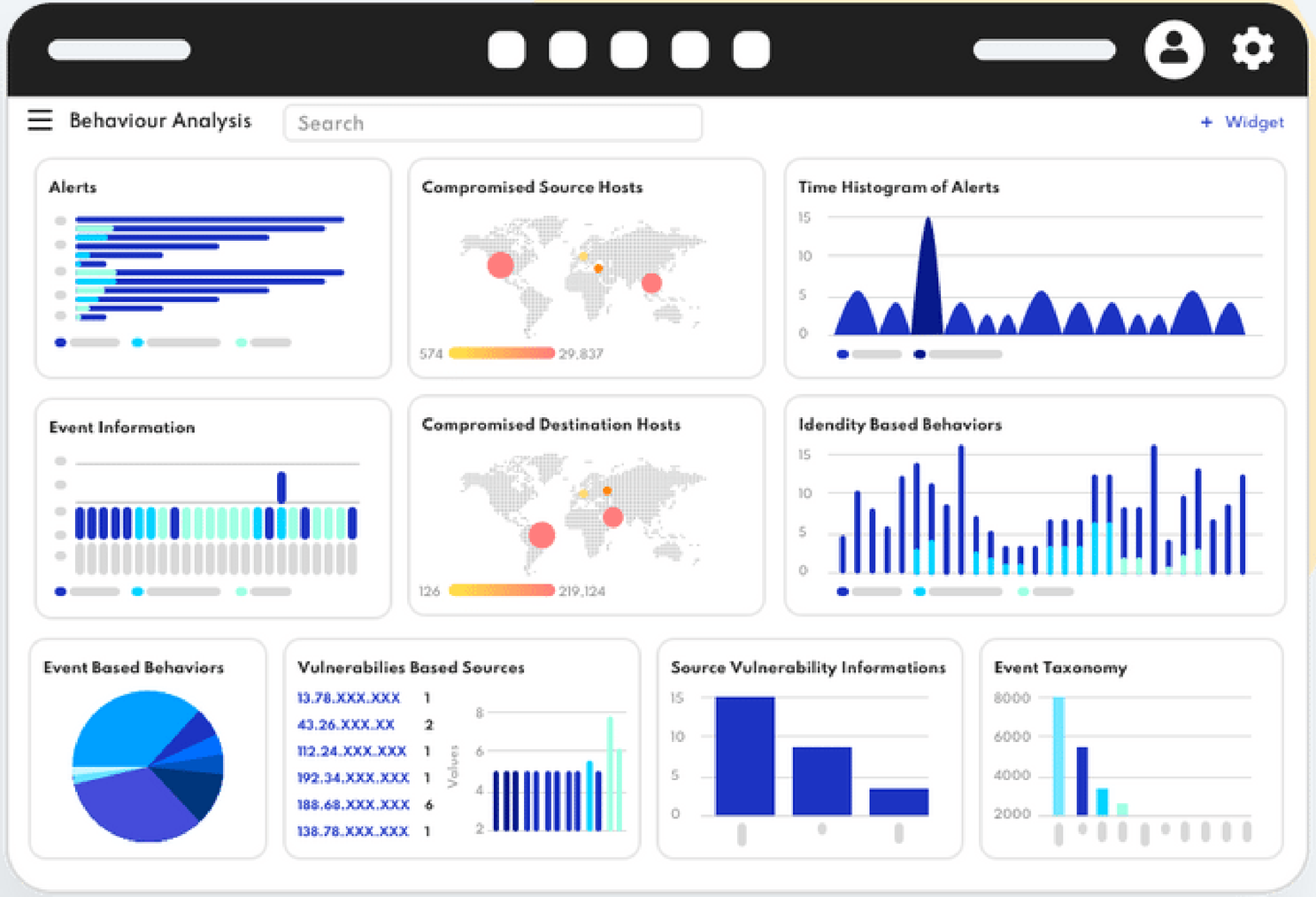
Segurança

AAA - Accounting

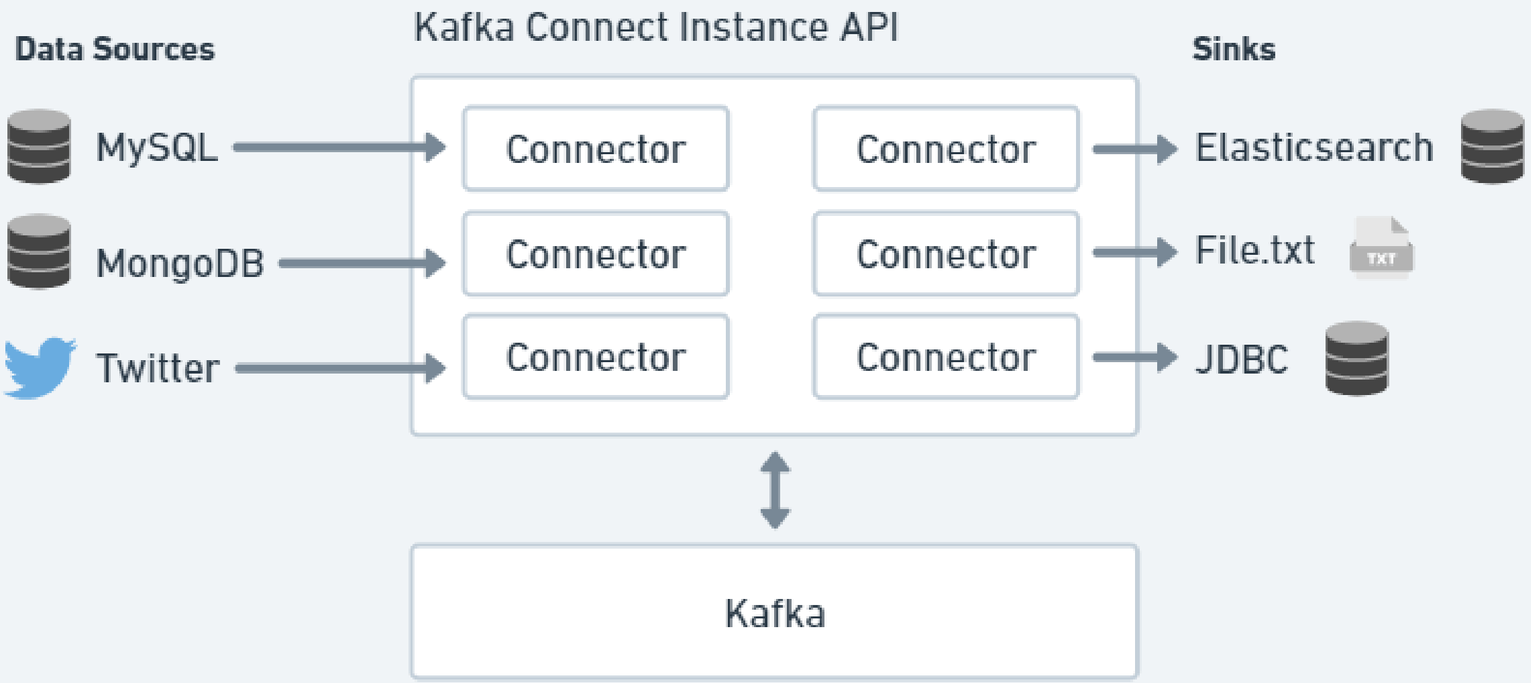
Time	Resource Name	Type	Action	Performed by	Parameters
Nov 19, 2020 5:20:21 PM	Default Threshold - HOMEPA...	Threshold Profiles	Updated	mounika.ks	{"objects_alert_broken":{"seve...
Nov 19, 2020 5:11:53 PM	Default Threshold - HOMEPA...	Threshold Profiles	Updated	mounika.ks	{"objects_alert_broken":{"seve...
Nov 19, 2020 3:07:11 PM	Site24x7	Monitors	Updated	mounika.ks	{"website":"http://www.site24...
Nov 19, 2020 1:06:41 PM	mounika.ks	Users	Logout	mounika.ks	None
Nov 19, 2020 12:39:53 PM	**Notification_XXXX**	**Notification Profiles**	**Created**	**mounika.ks**	**{profile_name=Notification_X...**
Nov 19, 2020 12:32:55 AM	cpustats_1.py - Default Thres...	Threshold Profiles	Updated	mounika.ks	{"anomaly_profile":false,"plugi...
Nov 18, 2020 10:27:19 PM	Plugin_XXXX	Threshold Profiles	Deleted	mounika.ks	None
Nov 18, 2020 10:17:00 PM	cpustats_1.py-mounika-9952	Run all rules	Run All Rules Initiated	Site24x7	{"1000000212161":{"confxml...
Nov 18, 2020 10:17:00 PM	cpustats_1.py-mounika-9952	Run all rules	Run All Rules Initiated	Site24x7	{"1000000212161":{"confxml...
Nov 18, 2020 10:16:54 PM	cpustats_1.py-mounika-9952	Monitors , Plugin	Created	Site24x7	{"heartbeat_required":true,"sc...
Nov 18, 2020 9:35:02 PM	Plugin_XXXX	Threshold Profiles	Created	mounika.ks	{"anomaly_profile":false,"plugi...

Segurança

AAA - Accounting - SIEM

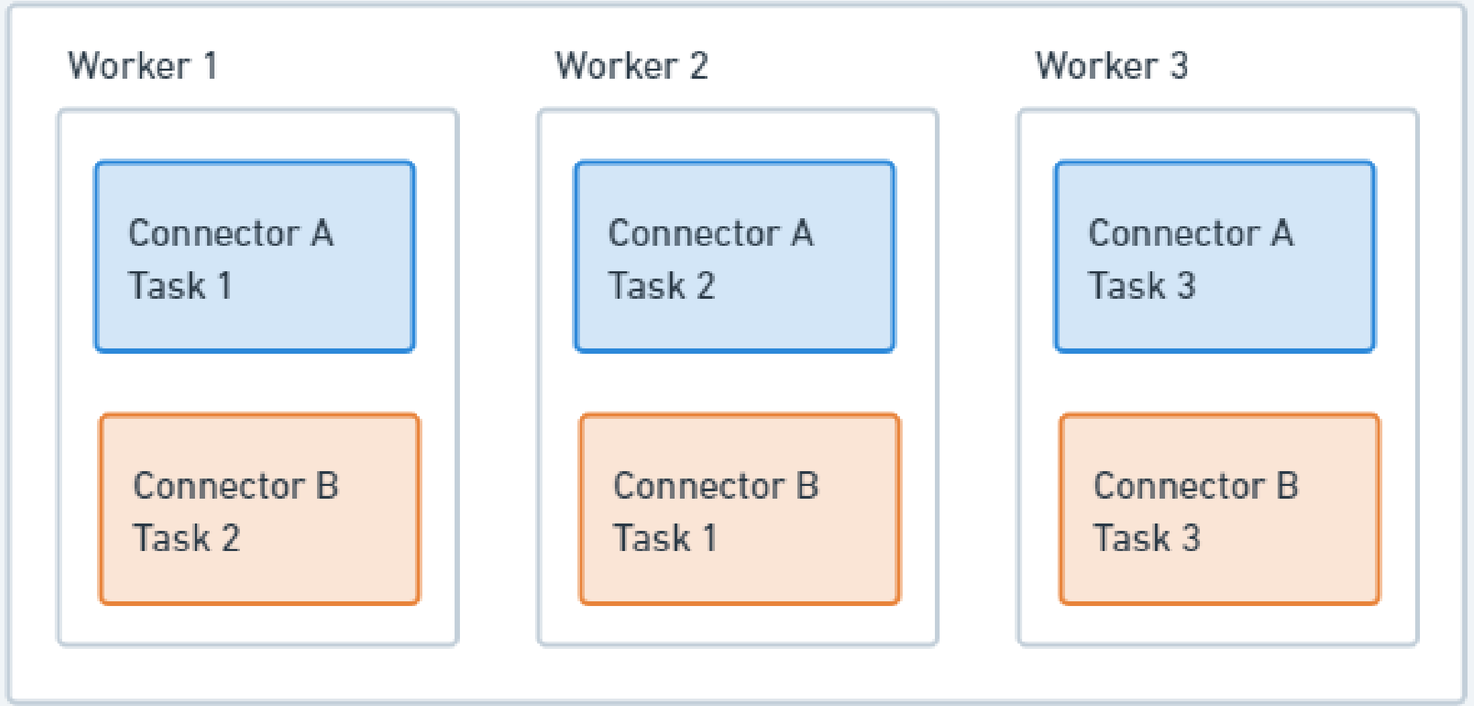


Kafka Connect



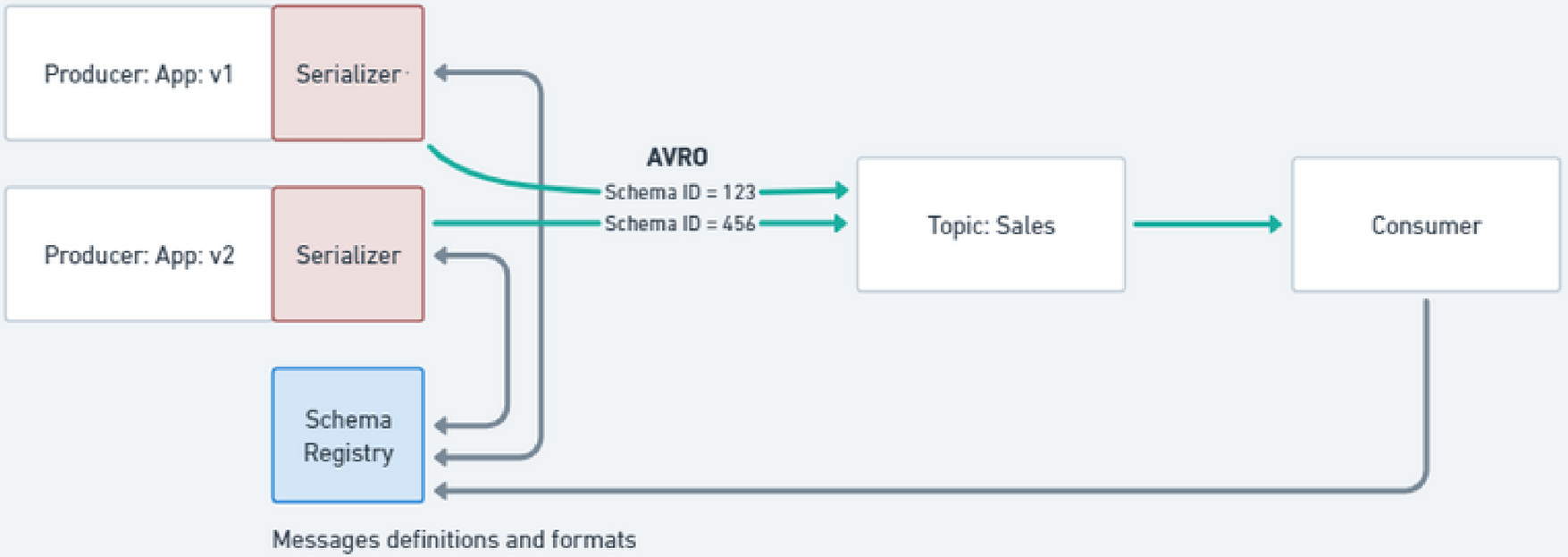
Kafka Connect

Workers e Taks



Kafka REST Proxy

Compatibilidade de dados



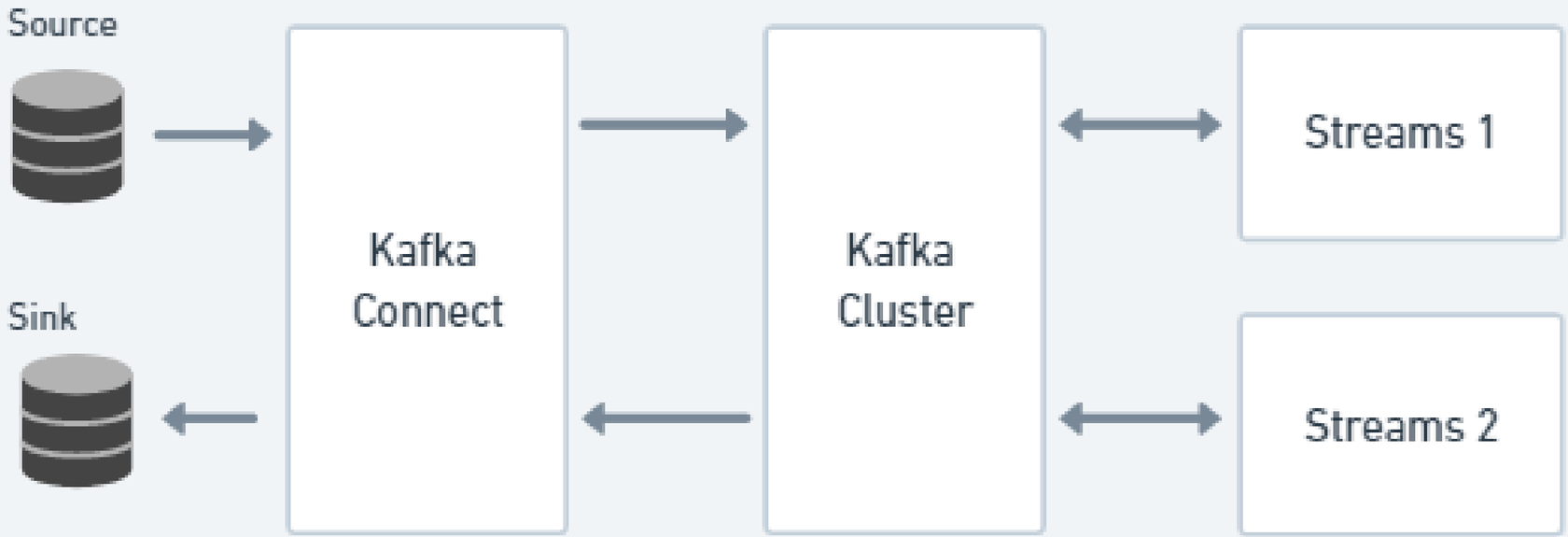
Compatibilidade de dados

Apache AVRO

```
{
  "namespace": "pnda.entity",
  "type": "record",
  "name": "event",
  "fields": [
     {"name": "timestamp", "type": "long"},
     {"name": "src",       "type": "string"},
     {"name": "host_ip",   "type": "string"},
     {"name": "rawdata",   "type": "bytes"}
  ]
}
```

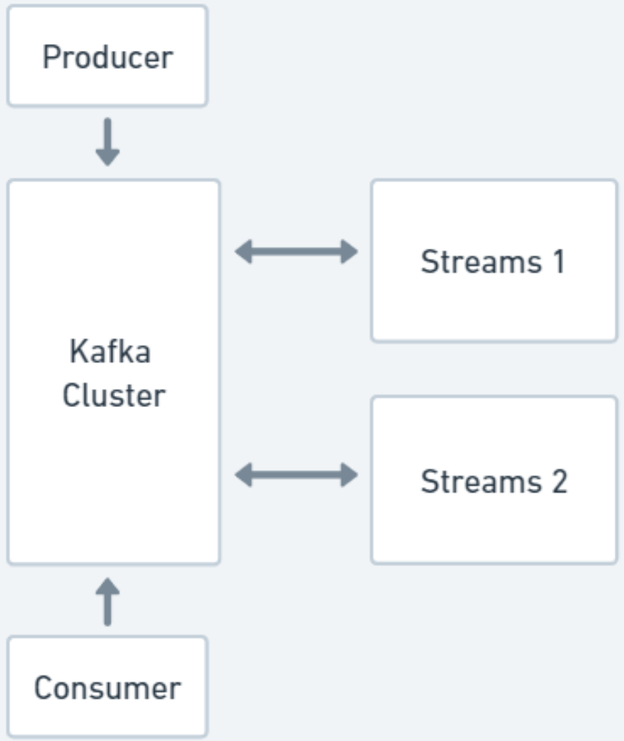
Kafka Streams

Utilizando Kafka Connect



Kafka Streams

Utilizando Produtor-Consumidor



Confluent ksqlDB

```
ksql > CREATE TABLE products WITH (KAFKA_TOPIC='products', key='ID', VALUE_FORMAT='AVRO')
```

```
ksql > SELECT id, name, price FROM products EMIT CHANGES;
```

```
ksql > CREATE table products_avg AS SELECT SUM(price) / COUNT(*) as avg FROM products;
```

```
ksql > SELECT avg FROM products_avg EMIT CHANGES;
```


REST API

Botando em Prática

- Setup do Apache Kafka
- Sistema produtor-consumidor
- Consumer Groups
- Uso do describe

MEMBROS DA EQUIPE

Breno Farias da Silva

Estudante de Ciência da Computação da UTFPR

João Henrique Gouveia Sperandio

Estudante de Ciência da Computação da UTFPR

Felipe Archanjo

Estudante de Ciência da Computação da UTFPR