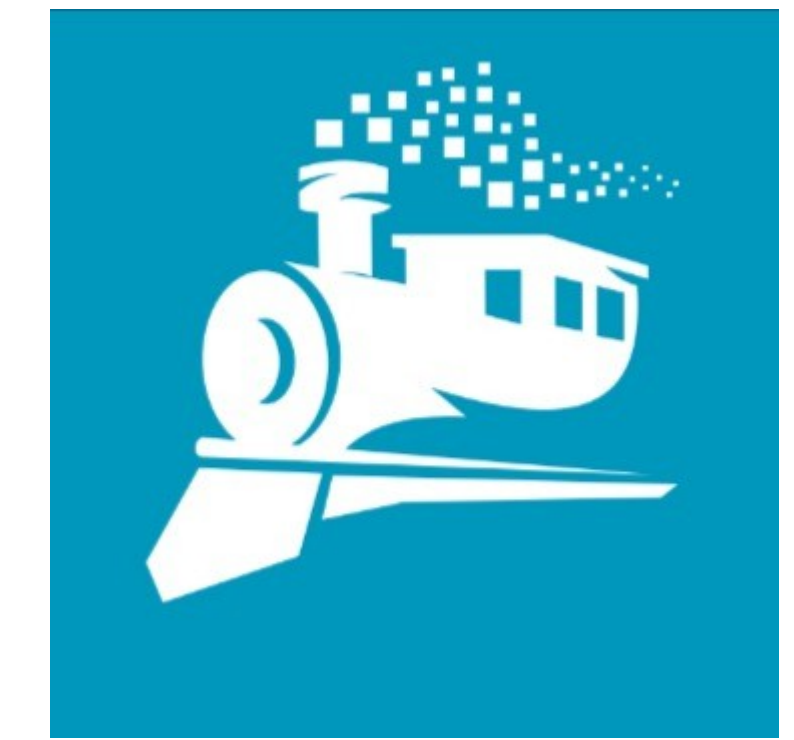


Processamento e Análise de Dados em Tempo Real com Python, Kafka e ElasticSearch

DataTrain meetup



Cícero
Moura

Desenvolvedor Full Stack na
Máxima Tech, Pós-Graduando em
IoT, Big Data e Machine Learning.
Entusiasta de Tecnologia
Assistiva.

Sobre o que vamos conversar?



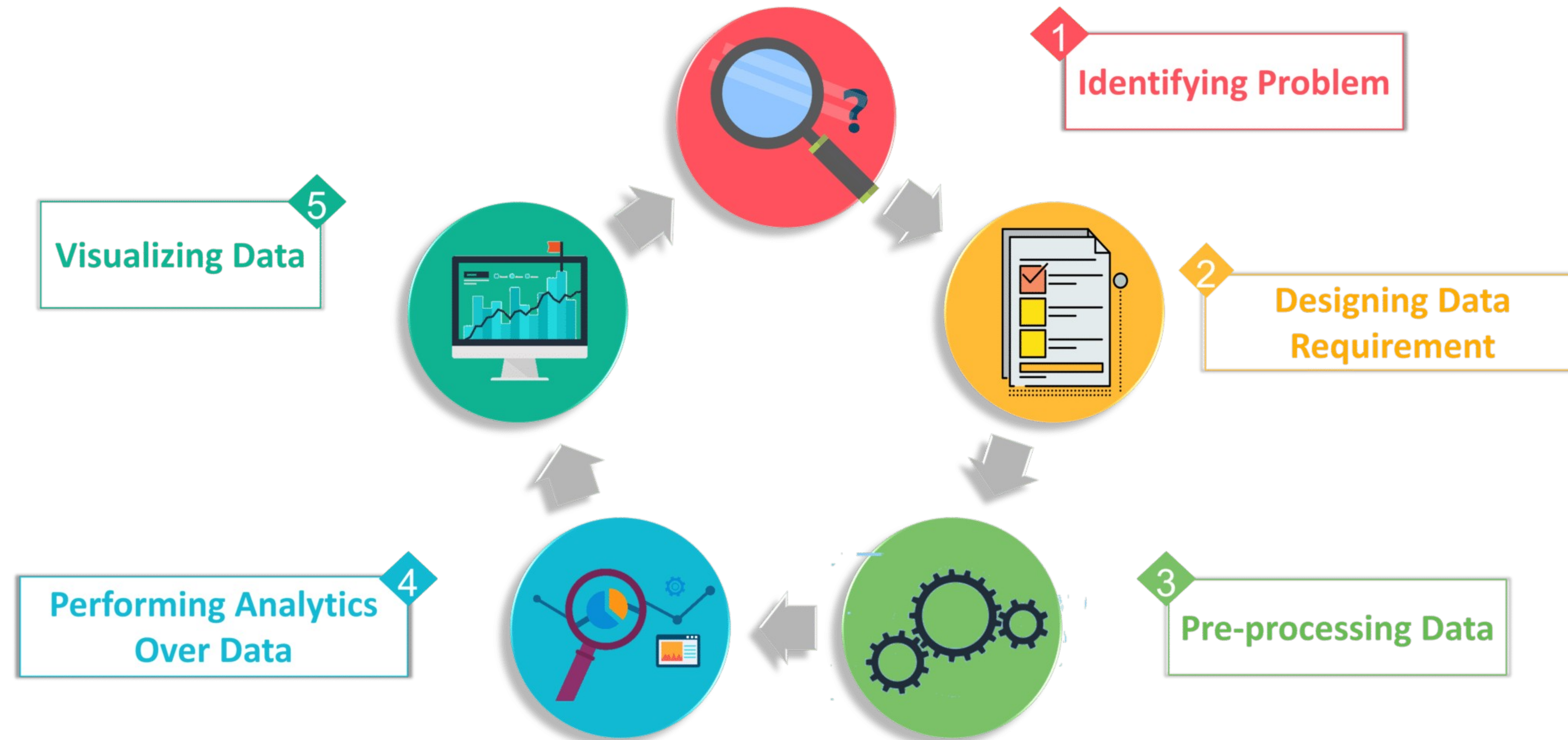
- Conceitos;
- Arquiteturas de Big Data;
- Python;
- Apache Kafka;
- ElasticSearch;
- Análise de Dados em “Tempo Real”.

Big Data?

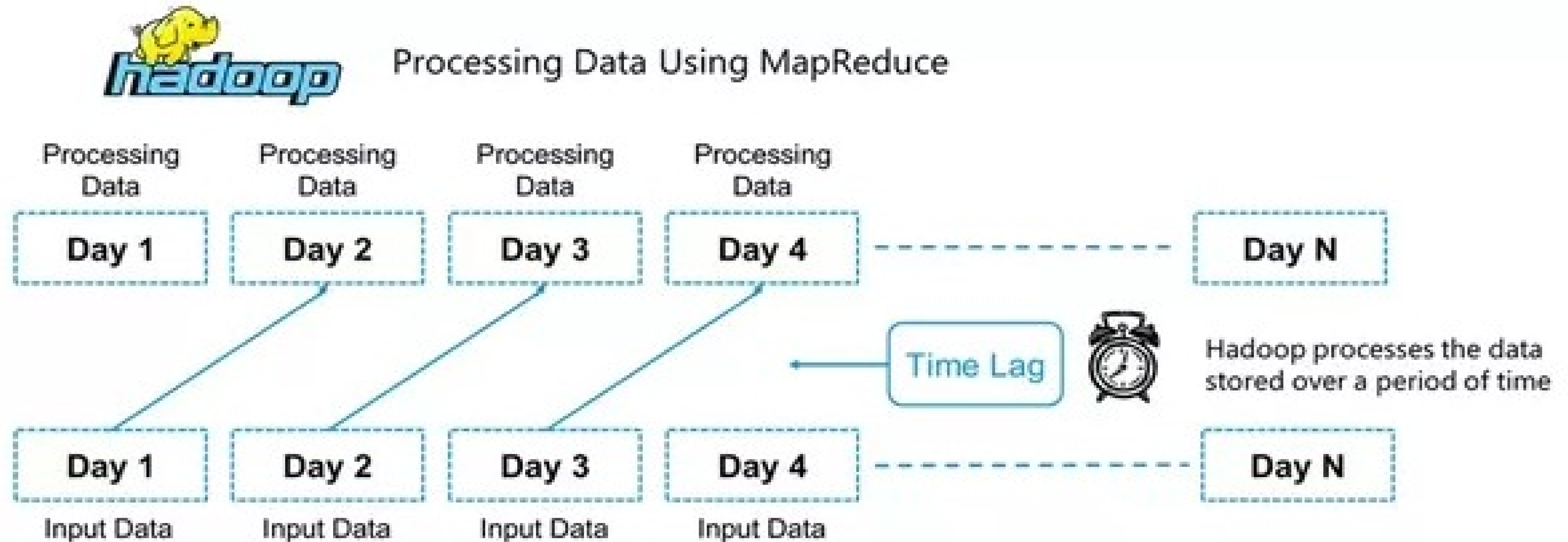


Conceitos

Processamento e Análise de Dados



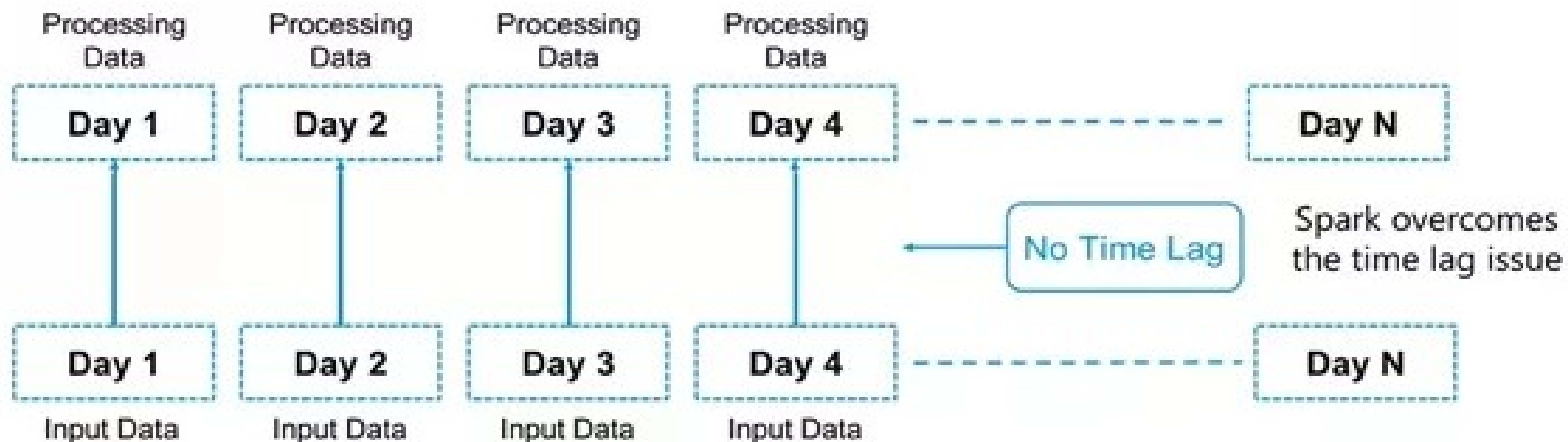
Processamento por Batch



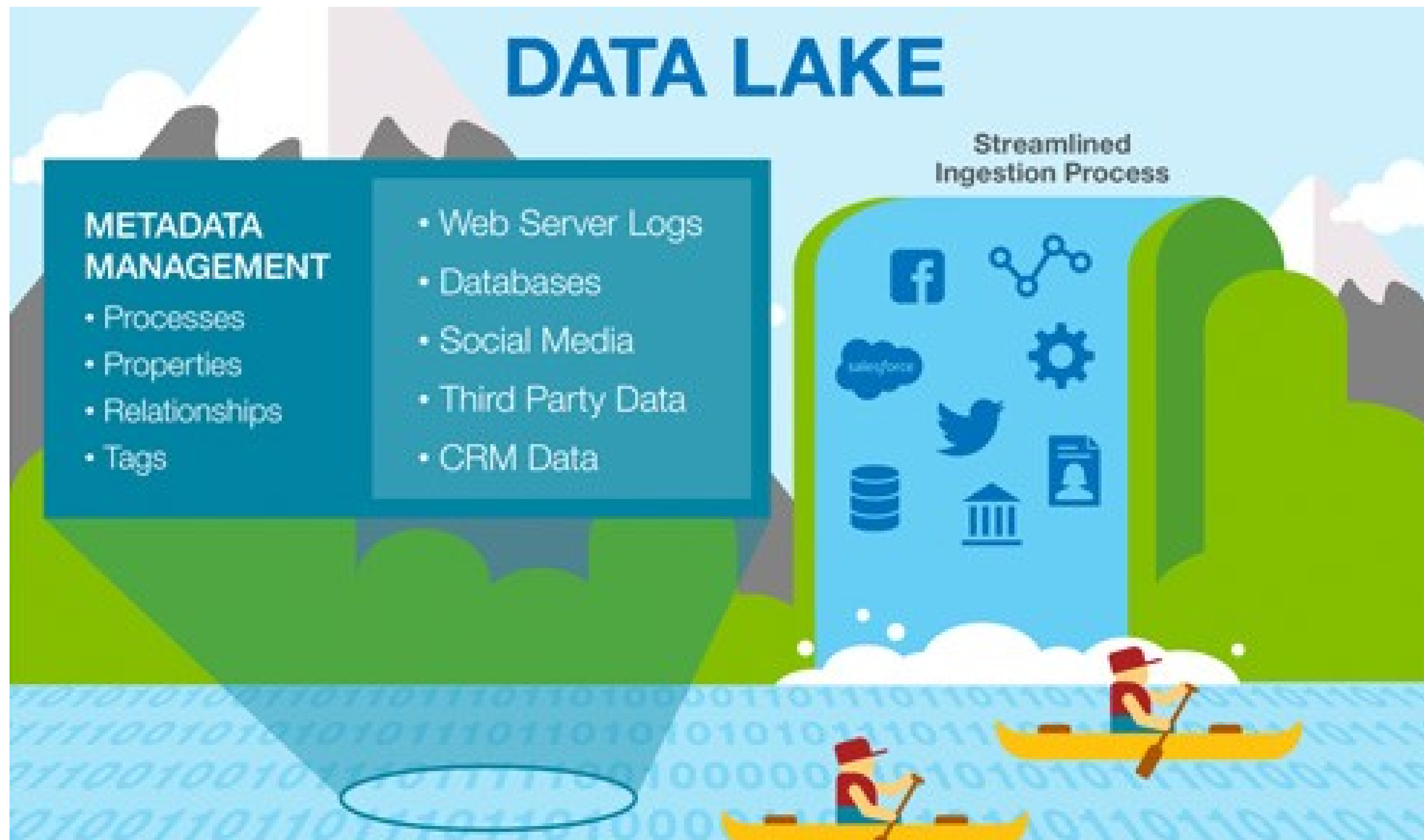
Processamento por Streaming



Real Time Processing in Spark



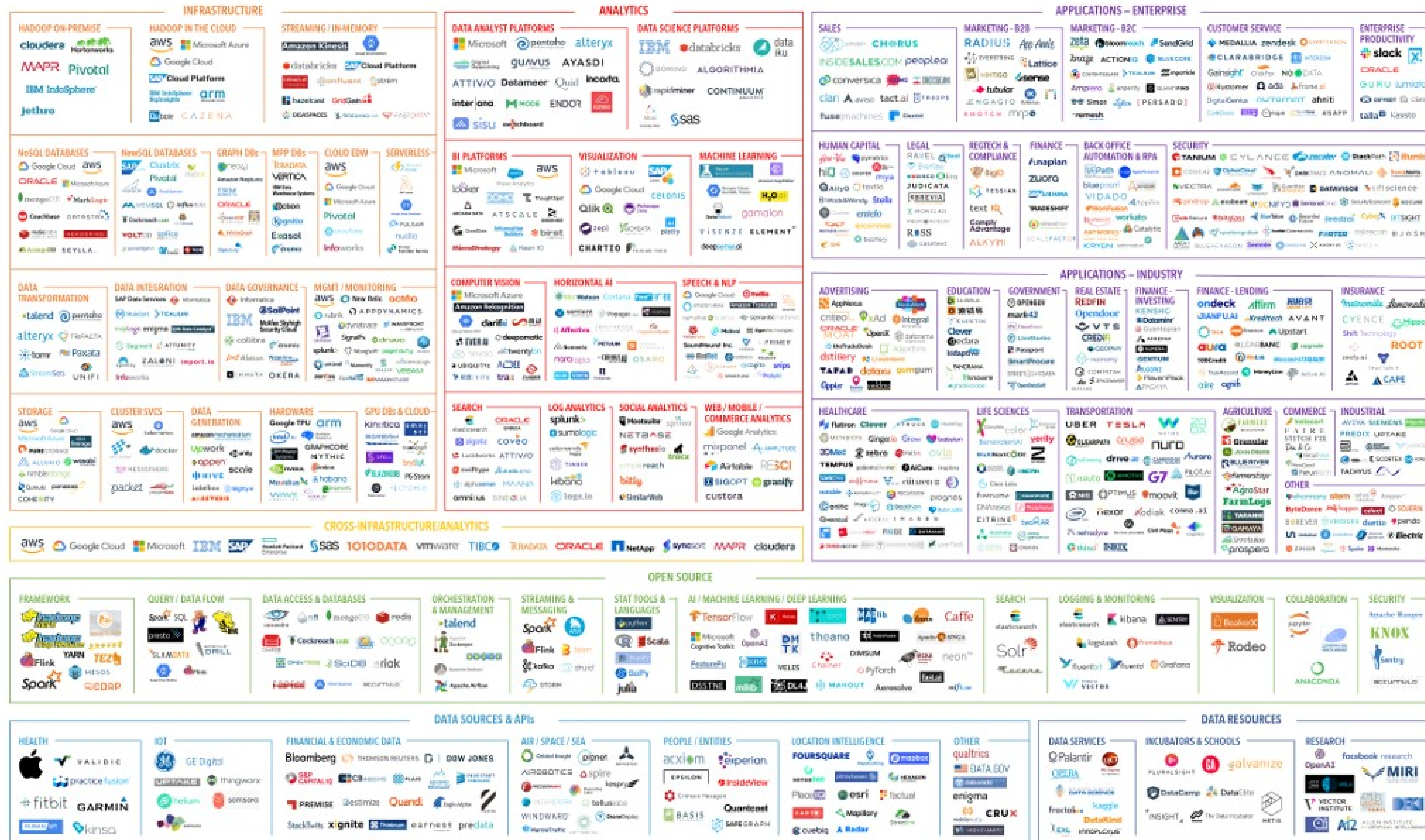
Data Lake



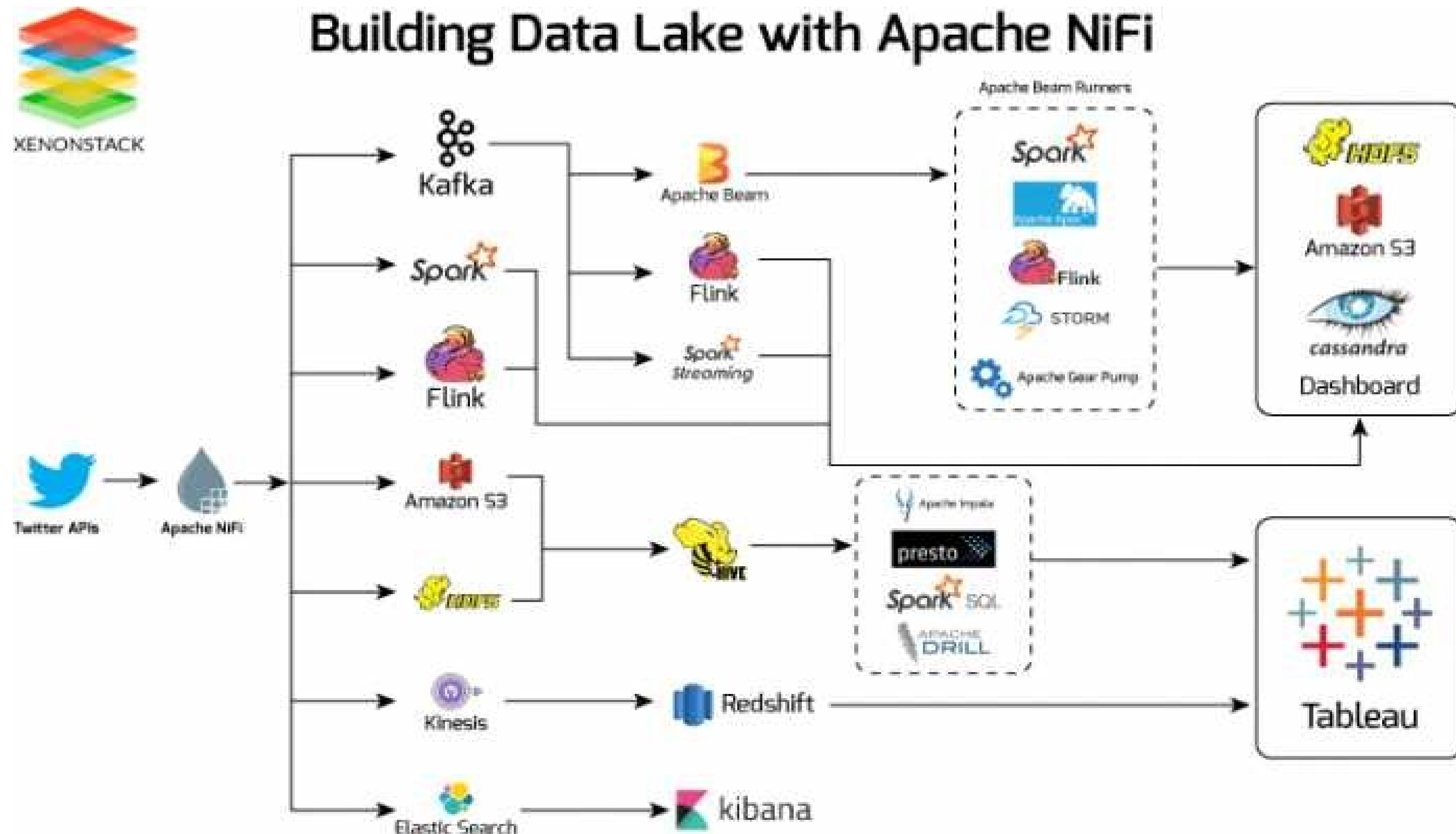
E como fazer tudo isso?



DATA & AI LANDSCAPE 2019



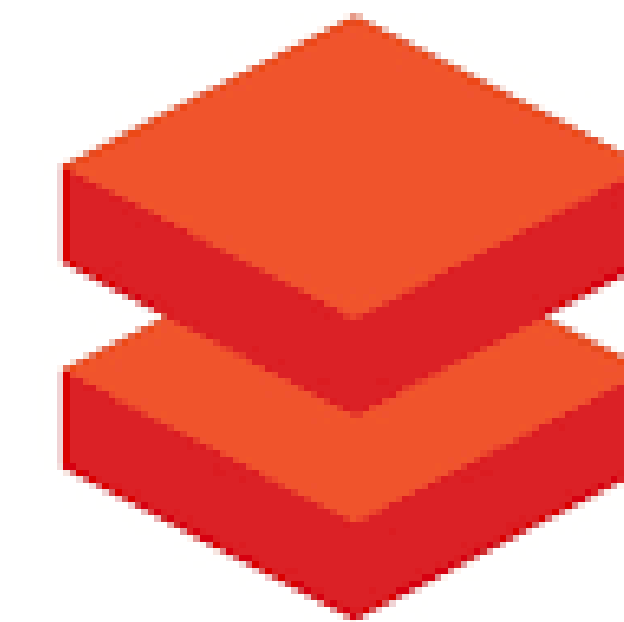
Algumas Ferramentas



Plataformas em nuvem



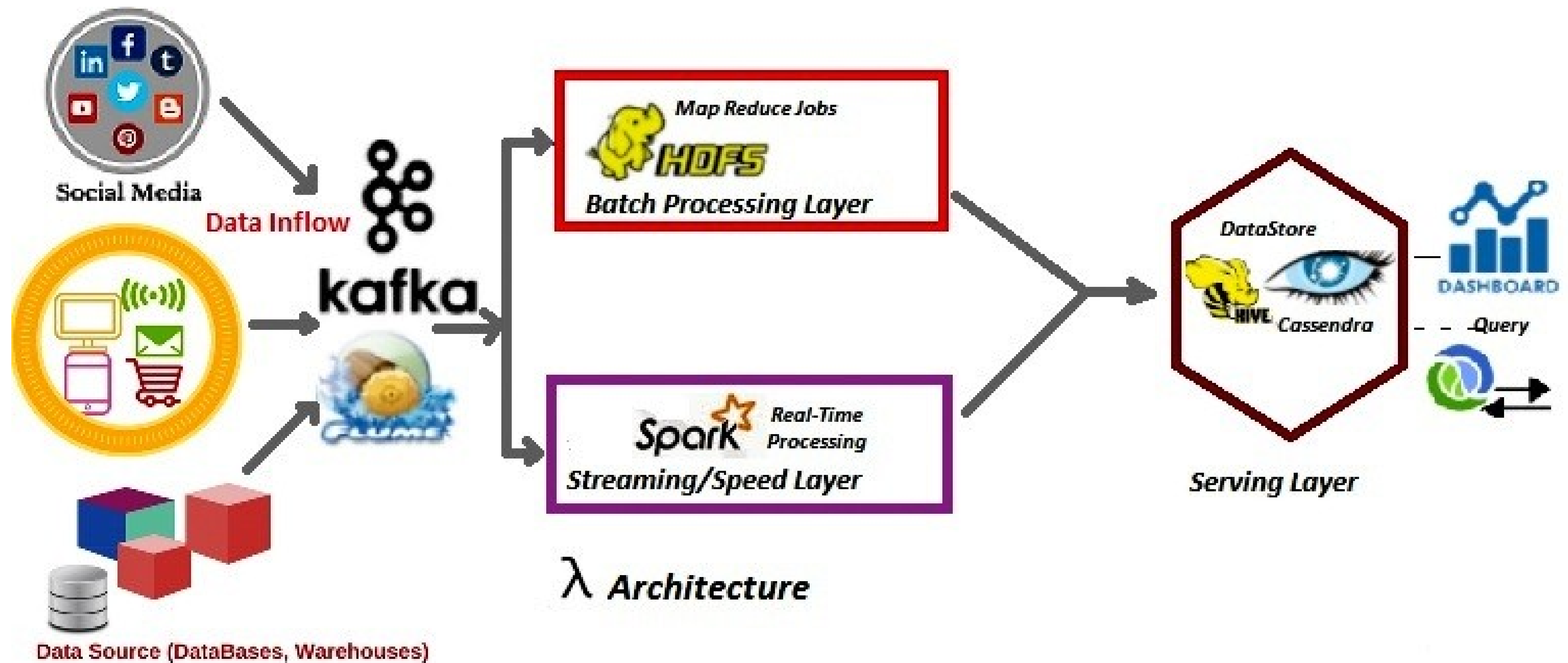
Google Cloud



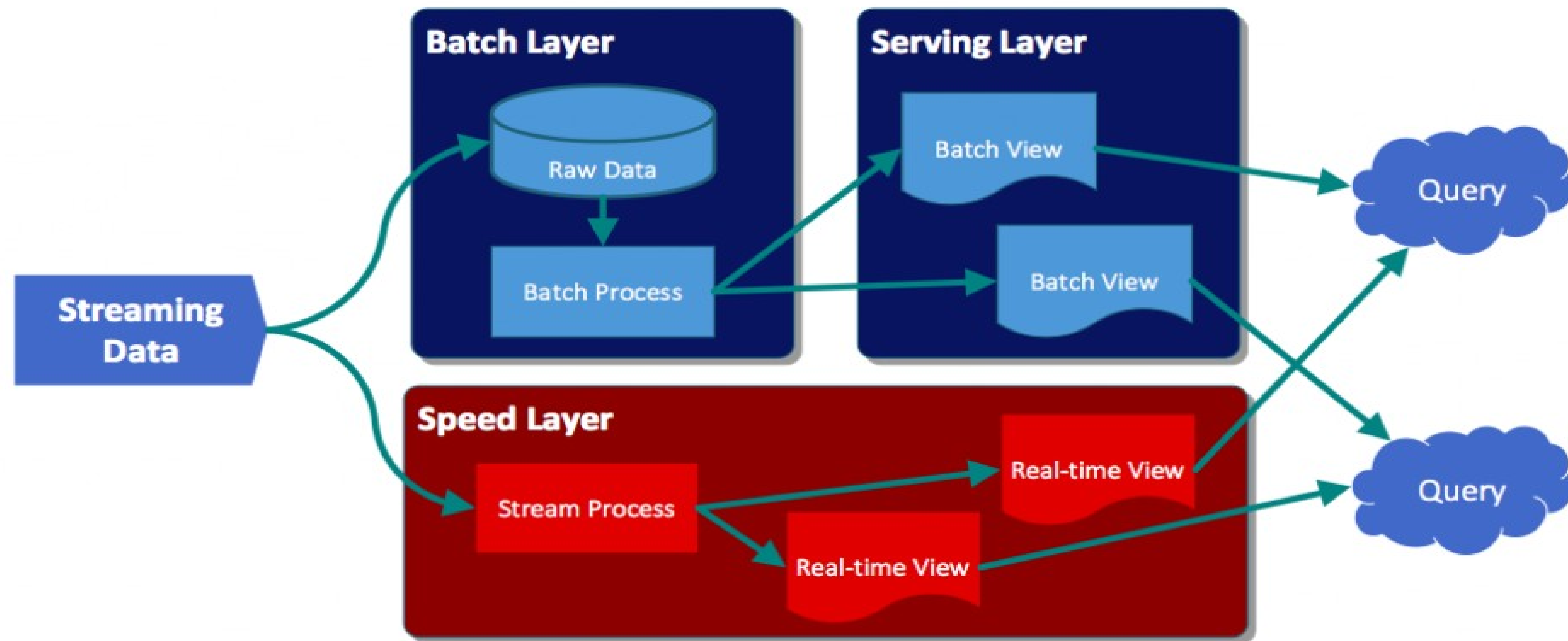
databricks®

Arquiteturas

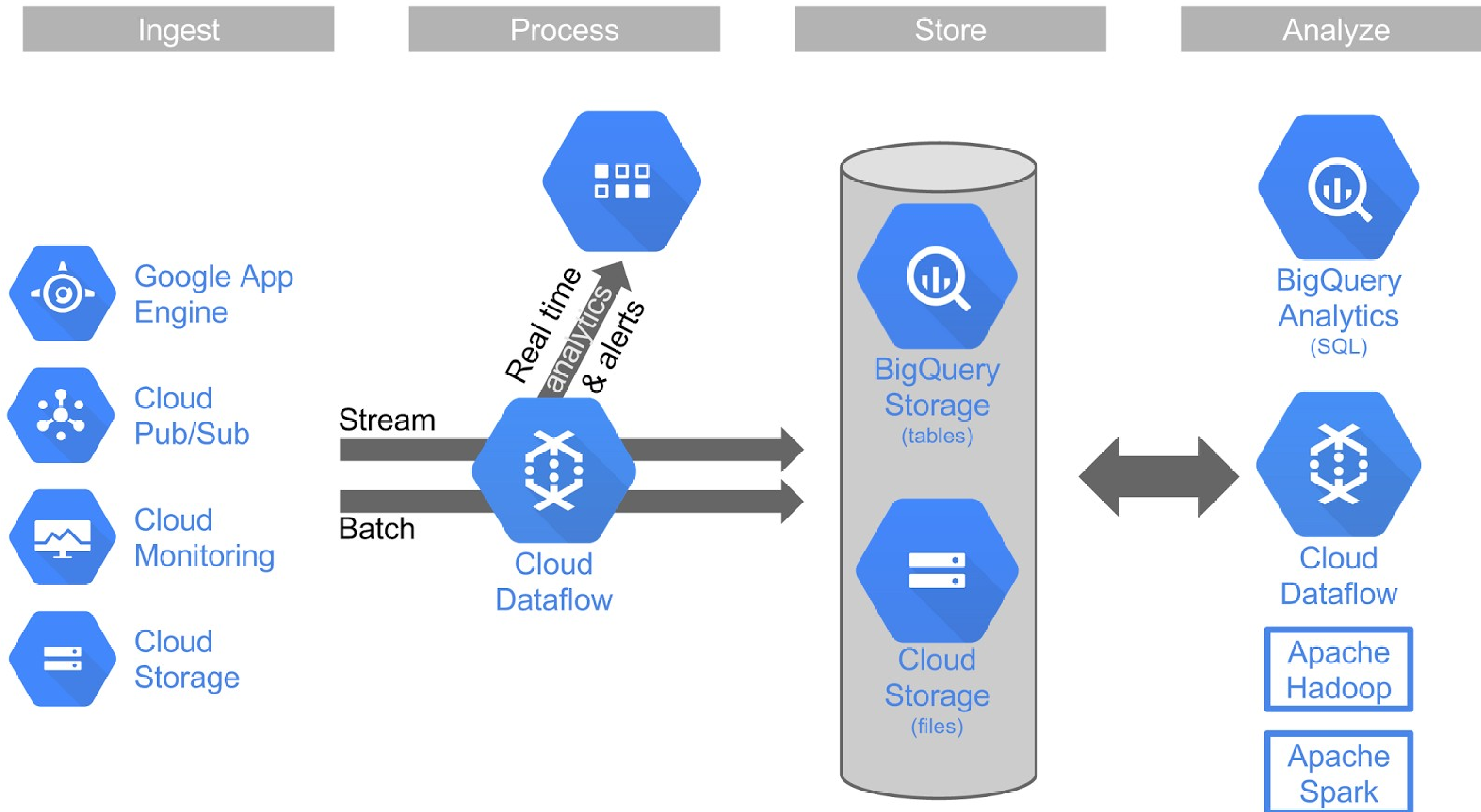
Lambda



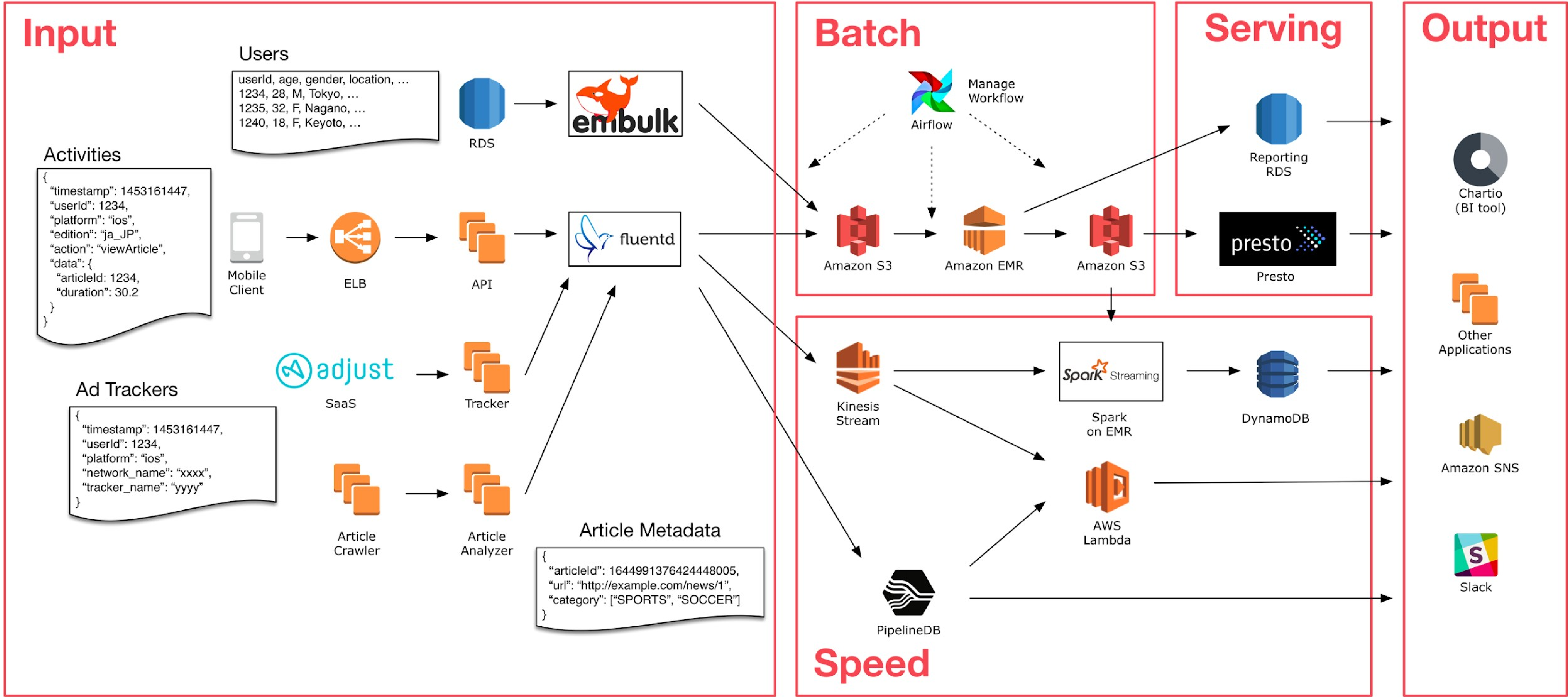
Kappa



Plataforma Google Cloud

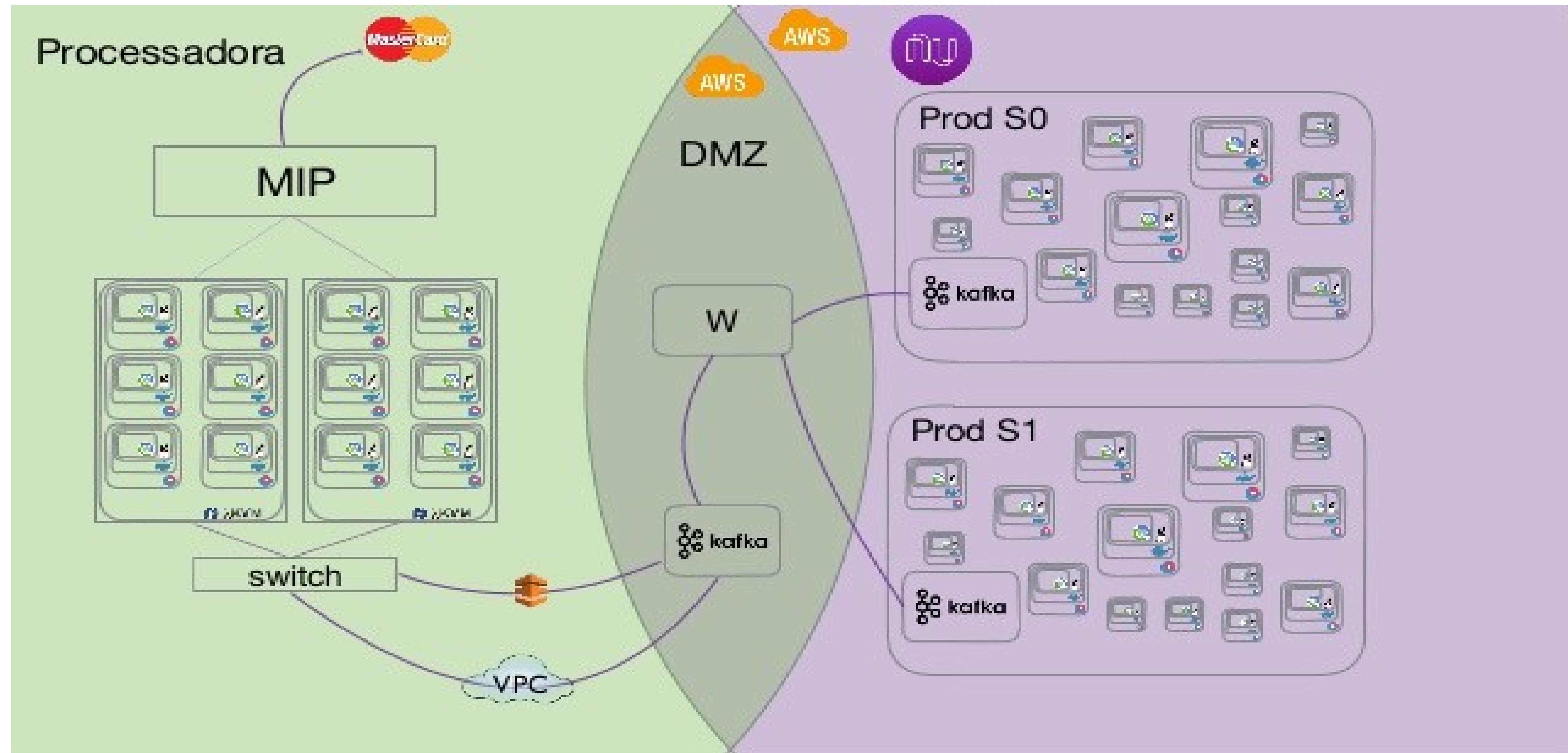


Plataforma AWS

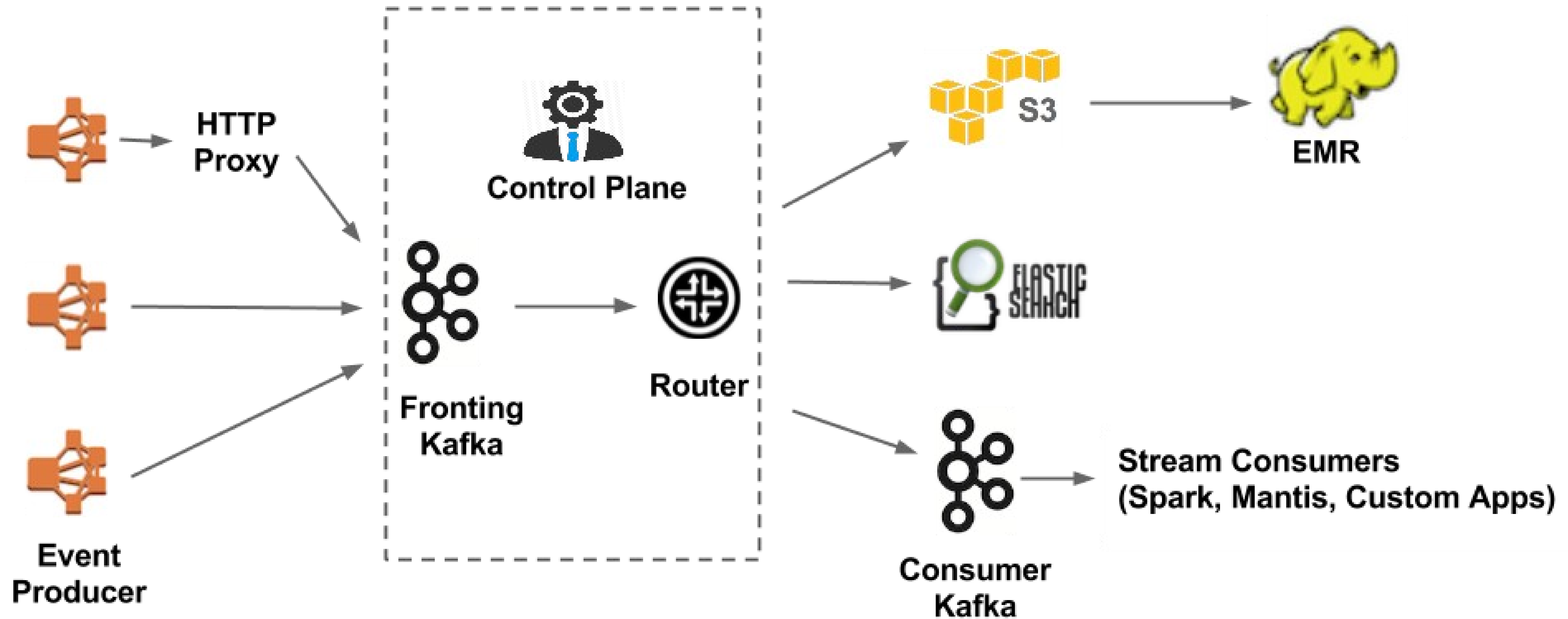


Cases

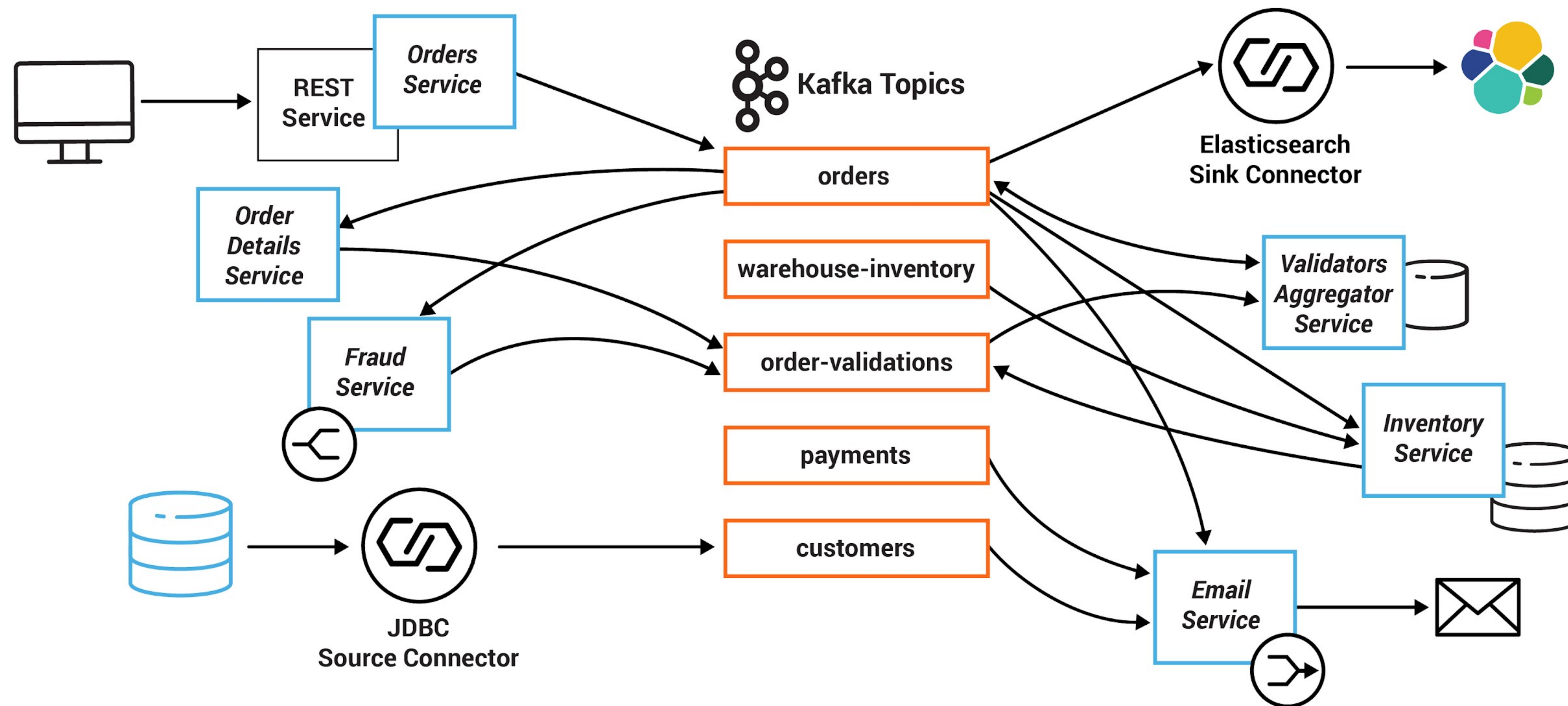
Pagamentos Nubank



Pipeline Netflix



Pipeline com Microserviços



Hands-on

Ferramentas

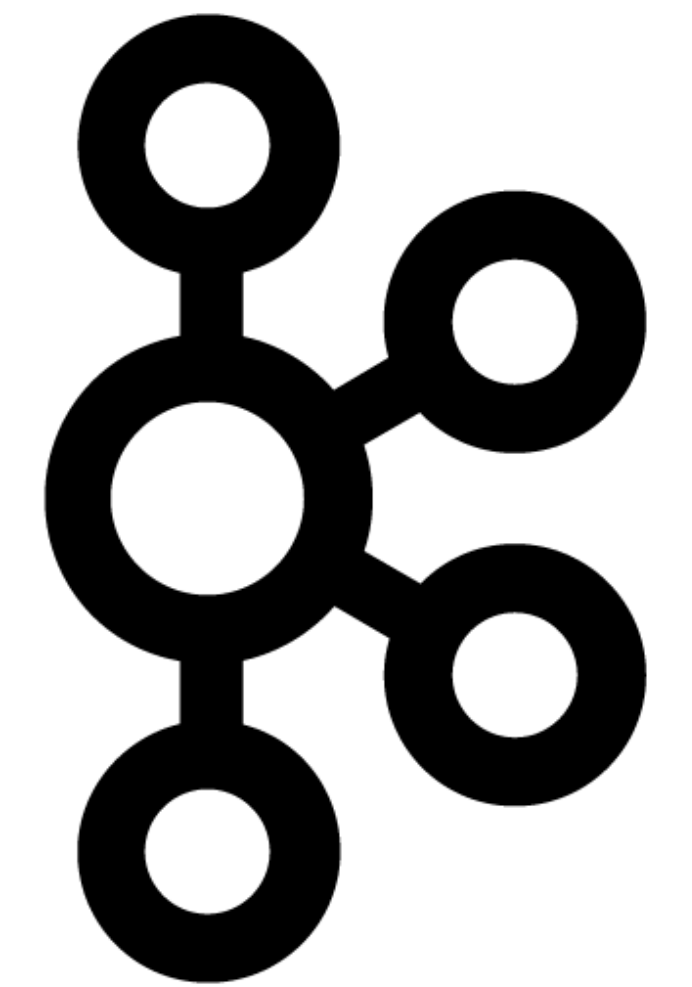


elasticsearch

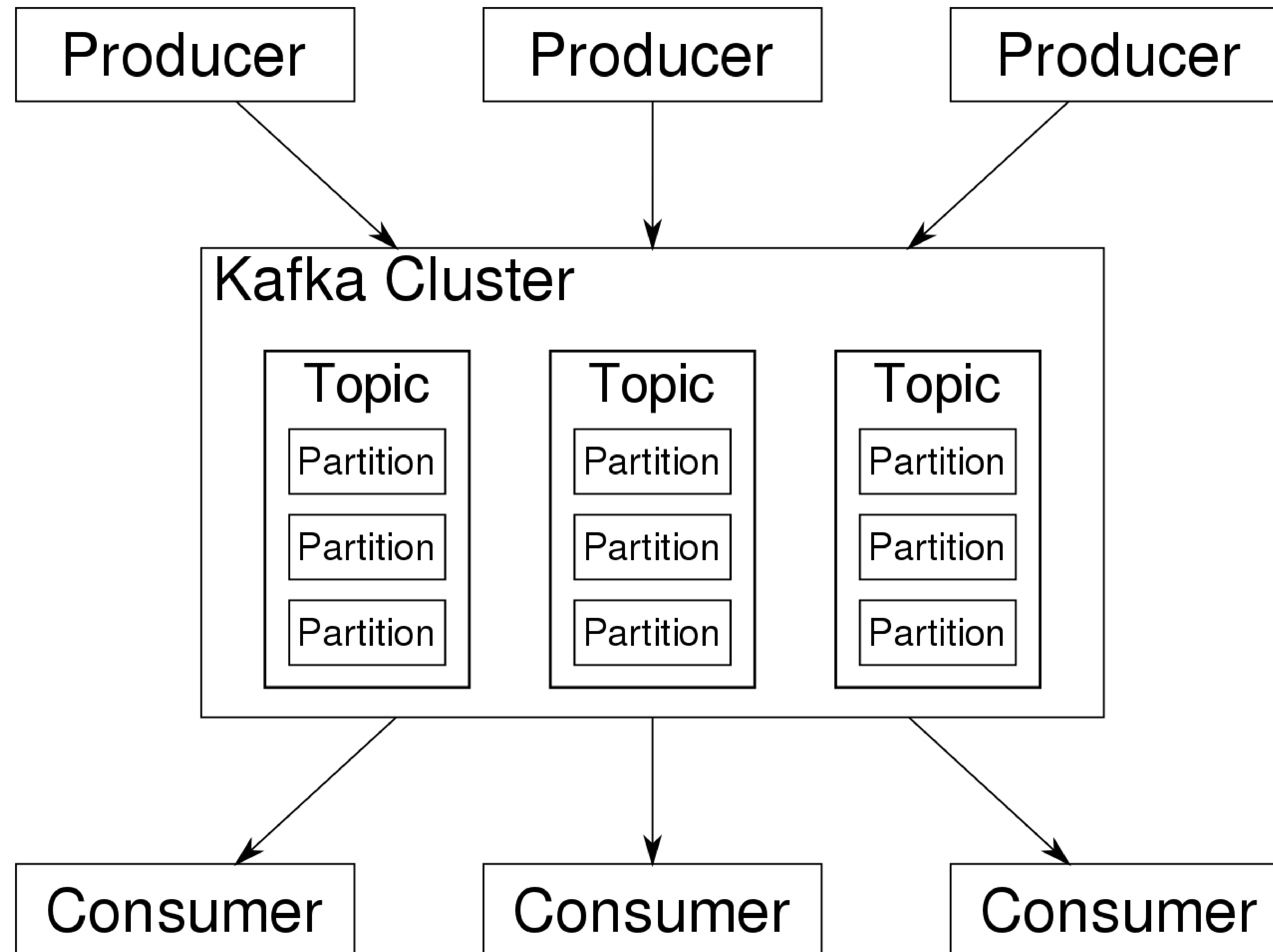


Apache Kafka: Conceitos

- Topics e partitions;
Brokers;
Producers;
Consumers;
Zookeeper;
Kafka Connectors.



Apache Kafka: Conceitos



ElasticSearch

- Full text search;
Ferramenta de buscas opensource;
Escável e tolerante a falhas;
Possui API's;
Banco no formato JSON.

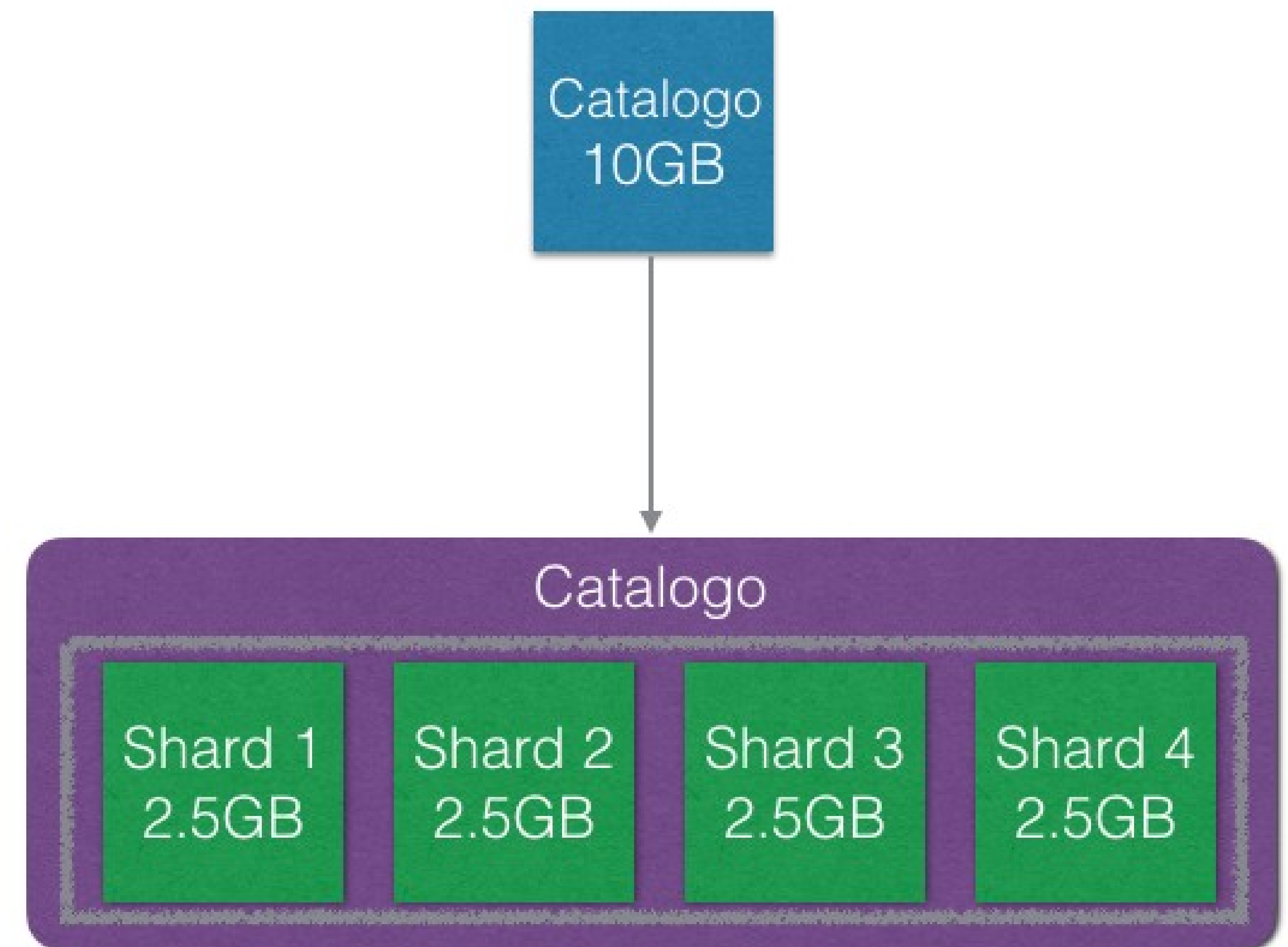


ElasticSearch: Conceitos

Banco Relacional	ElasticSearch
Instância	Index
Tabela	Type
Schema	Mapping
Tupla	Documento
Coluna	Atributo

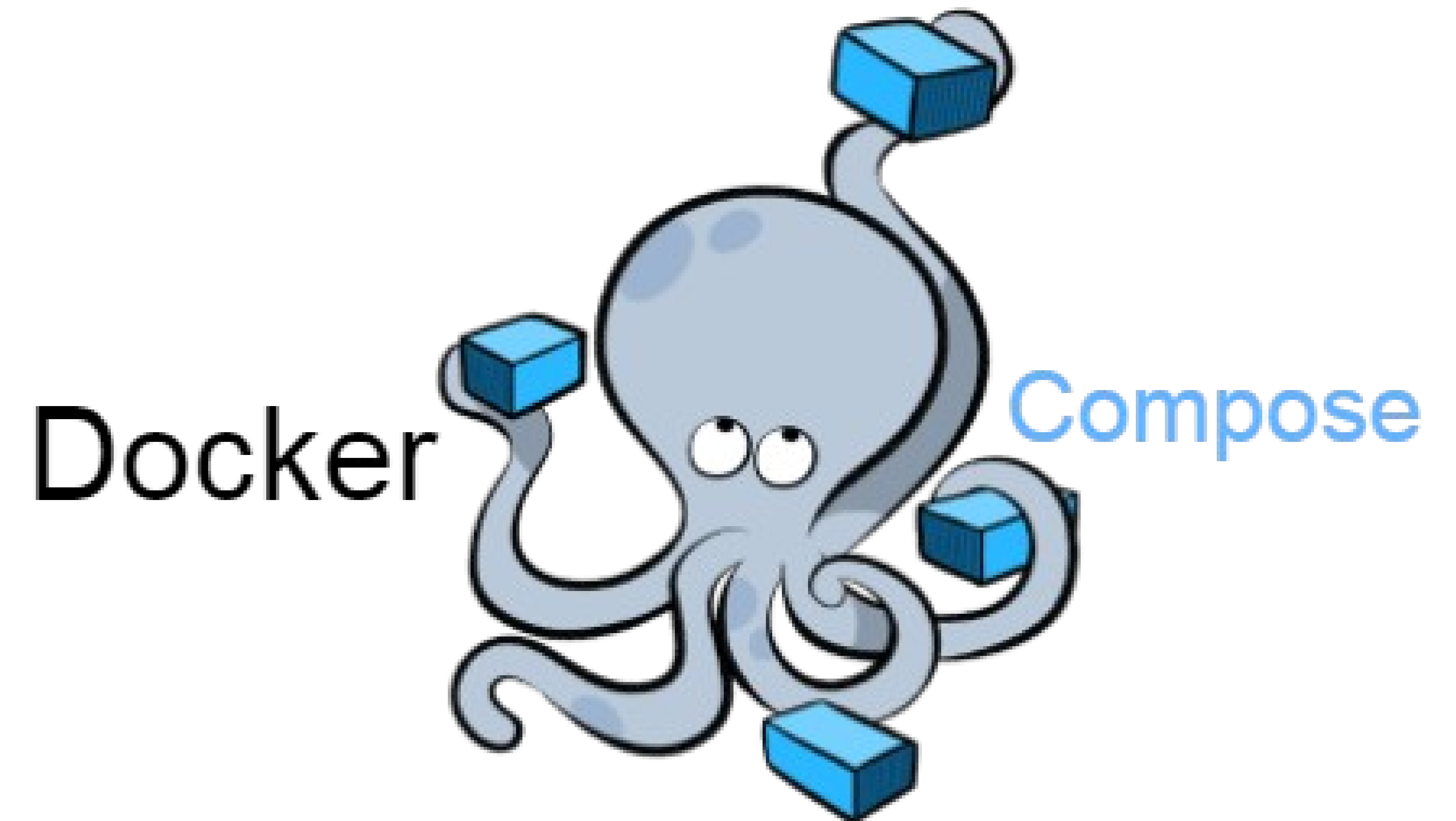
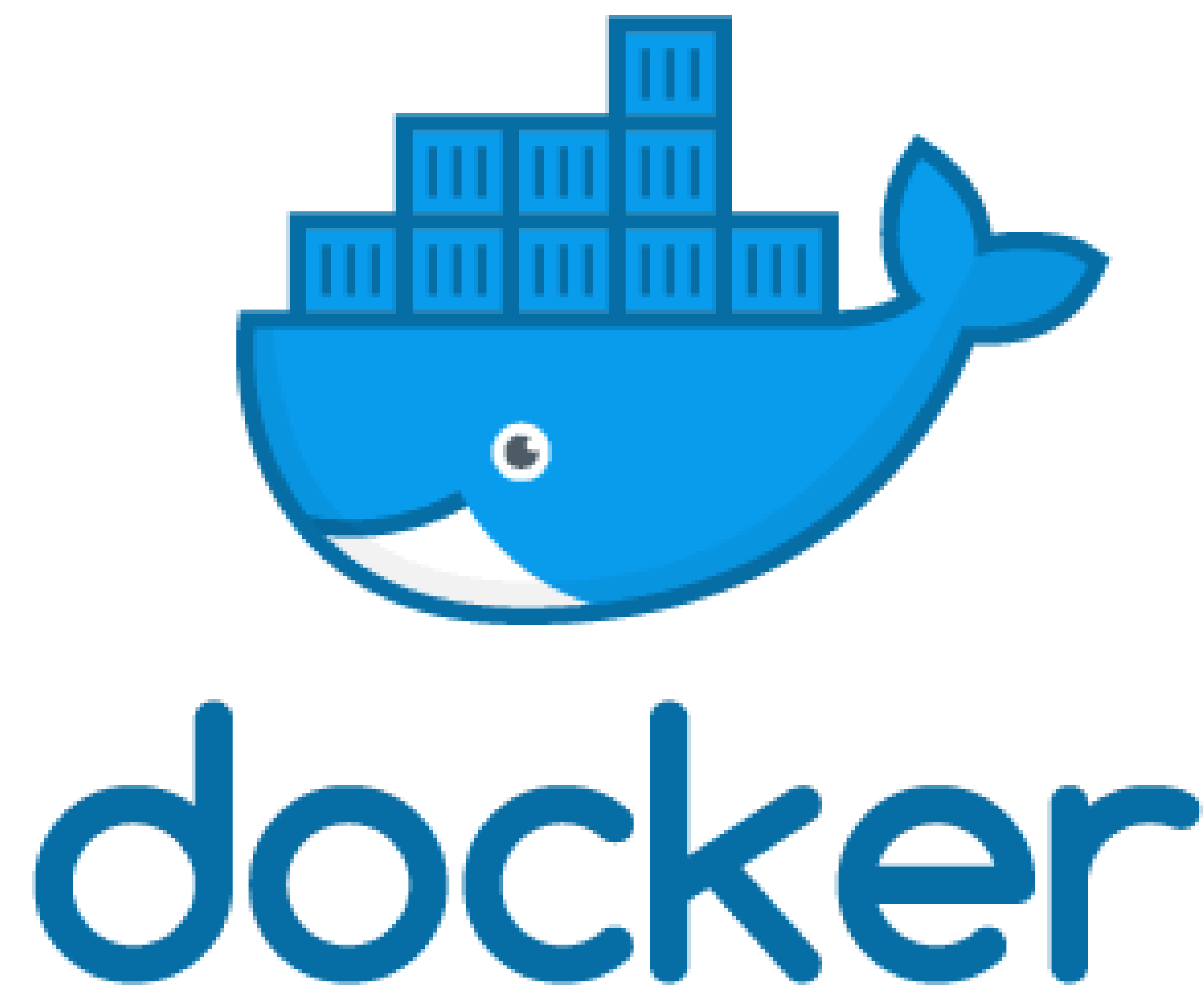
ElasticSearch: Conceitos

- Cluster;
Réplicas.



Show me the code!

Ambiente com Docker e Docker Compose



Cluster Elastic Search + Cerebro

The screenshot displays the Cerebro web interface for managing an Elasticsearch cluster. The top navigation bar includes icons for overview, nodes, rest, and more. The main header shows the cluster name 'docker-cluster' and summary statistics: 2 nodes, 5 indices, and 12 shards. Below this, there are filter inputs for indices (by name or aliases) and nodes (by name), along with checkboxes for 'closed (0)' and '.special (2)' indices.

The main content area is a table with columns for indices and nodes. The indices listed are 'dados', 'dados-tweets', and 'test-topic4'. The nodes listed are 'elasticsearch' and 'elasticsearch01'. The table shows the number of shards, documents, and size for each index, and the status of each node (green for healthy, yellow for warning, red for error).

Index	Shards	Docs	Size	Node	Status
dados	2 * 2	0	460.00b	elasticsearch	Green
				elasticsearch01	Green
dados-tweets	1 * 2	110	97.25KB	elasticsearch	Green
				elasticsearch01	Green
test-topic4	1 * 2	0	283.00b	elasticsearch	Green
				elasticsearch01	Green

Kafka + Connector + Connector UI

KAFKA CONNECT

2 Connectors

NEW

Search connectors

elasticsearch-sink-kafka

1x

→ Elastic Search

elasticsearch-sink1

1x

→ Elastic Search

Kafka Connect : /api/kafka-connect-1

Kafka Connect Version : 5.3.0-ccs

Kafka Connect UI Version : 0.9.4

Dashboard

EXPORT CONFIG

SINK CONNECTORS

2

SOURCE CONNECTORS

0

TOPICS USED BY CONNECTORS

2

Connect topology

dados-tweets

elasticsearch-sink-kafka

test-topic4

elasticsearch-sink1

33

Kafka Connector

```
{
  "connector.class": "io.confluent.connect.elasticsearch.ElasticsearchSinkConnector",
  "type.name": "kafka-connect",
  "key.converter.schemas.enable": "false",
  "tasks.max": "1",
  "topics": "dados-tweets",
  "name": "elasticsearch-sink-kafka",
  "value.converter.schemas.enable": "false",
  "key.ignore": "true",
  "connection.url": " http://10.158.0.5:9200",
  "value.converter": "org.apache.kafka.connect.json.JsonConverter",
  "key.converter": "org.apache.kafka.connect.storage.StringConverter",
  "schema.ignore": "true"
}
```

Jupyter Notebook

```
In [59]: #geração da nuvem de palavras em tempo real
frases = ''
for mensagem in consumer:
    texto = json.loads(mensagem.value.decode('utf-8'))
    frases = frases + texto['tweet']
    clear_output()
    wordcloud = WordCloud(max_font_size=100, width = 1520, height = 535).generate(frases)
    plt.figure(figsize=(16,9))
    plt.imshow(wordcloud)
    plt.axis("off")
    plt.show()
```



Código Completo da Talk

<https://bit.ly/2VSW06W>



Obrigado!



 /cicerojmm

 /cicerojmm

 /in/cicero-moura

 /cicerojmm