Stack

O que é Engenharia de dados

Stack

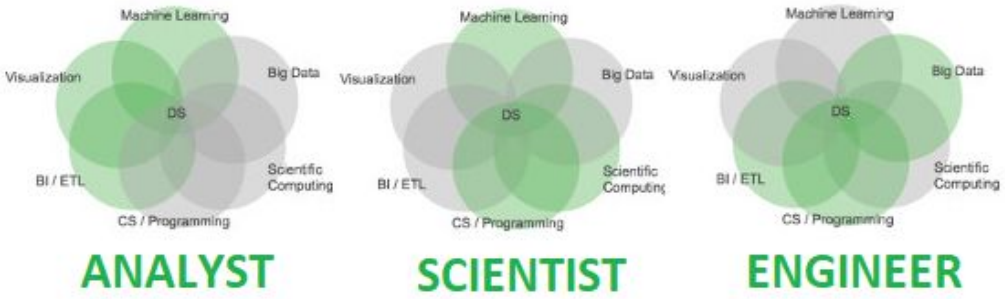
Para quem é esse curso?

1. Engenheiros de Dados.
2. Cientistas de Dados
3. Analista de Dados.
4. Gestores de Projeto.

Stack

Para quem é esse curso?

1. Engenheiros de Dados.
2. Cientistas de Dados
3. Analista de Dados.
4. Gestores de Projeto.



Stack

Responsabilidades do Engenheiro de Dados

Stack

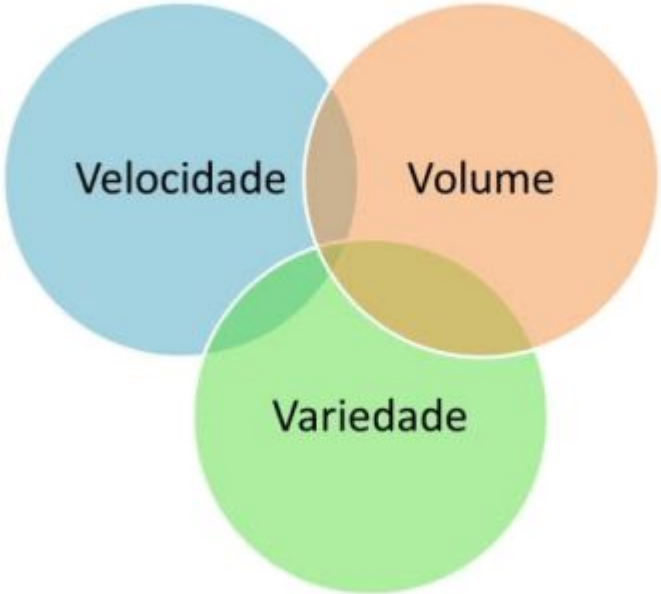
Responsabilidades do Cientista de Dados

Stack

O que é Big Data?

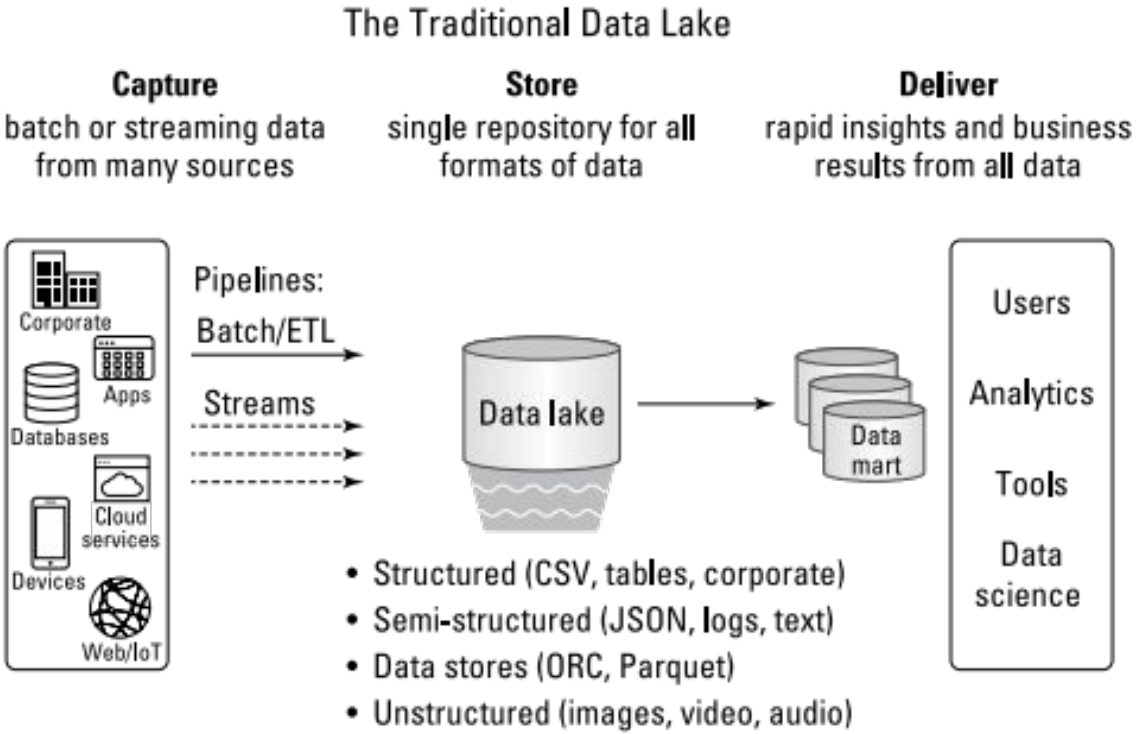
Stack

O que é Big Data?



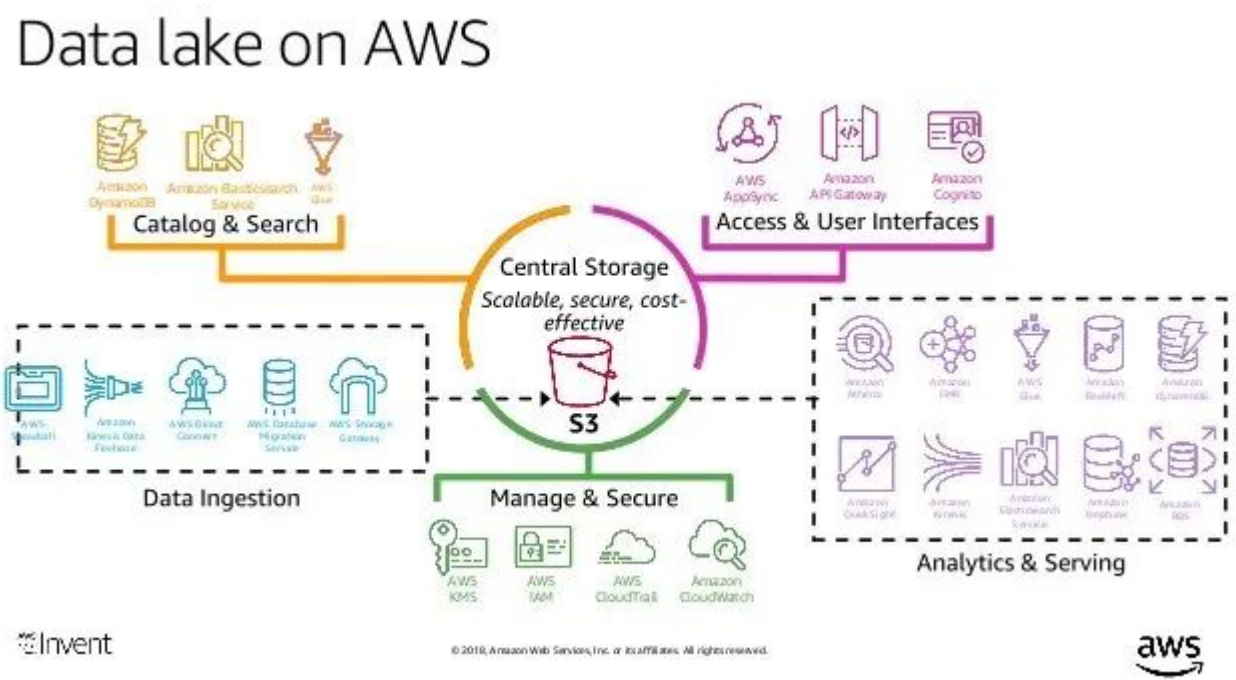
Stack

O que são Data Lakes?



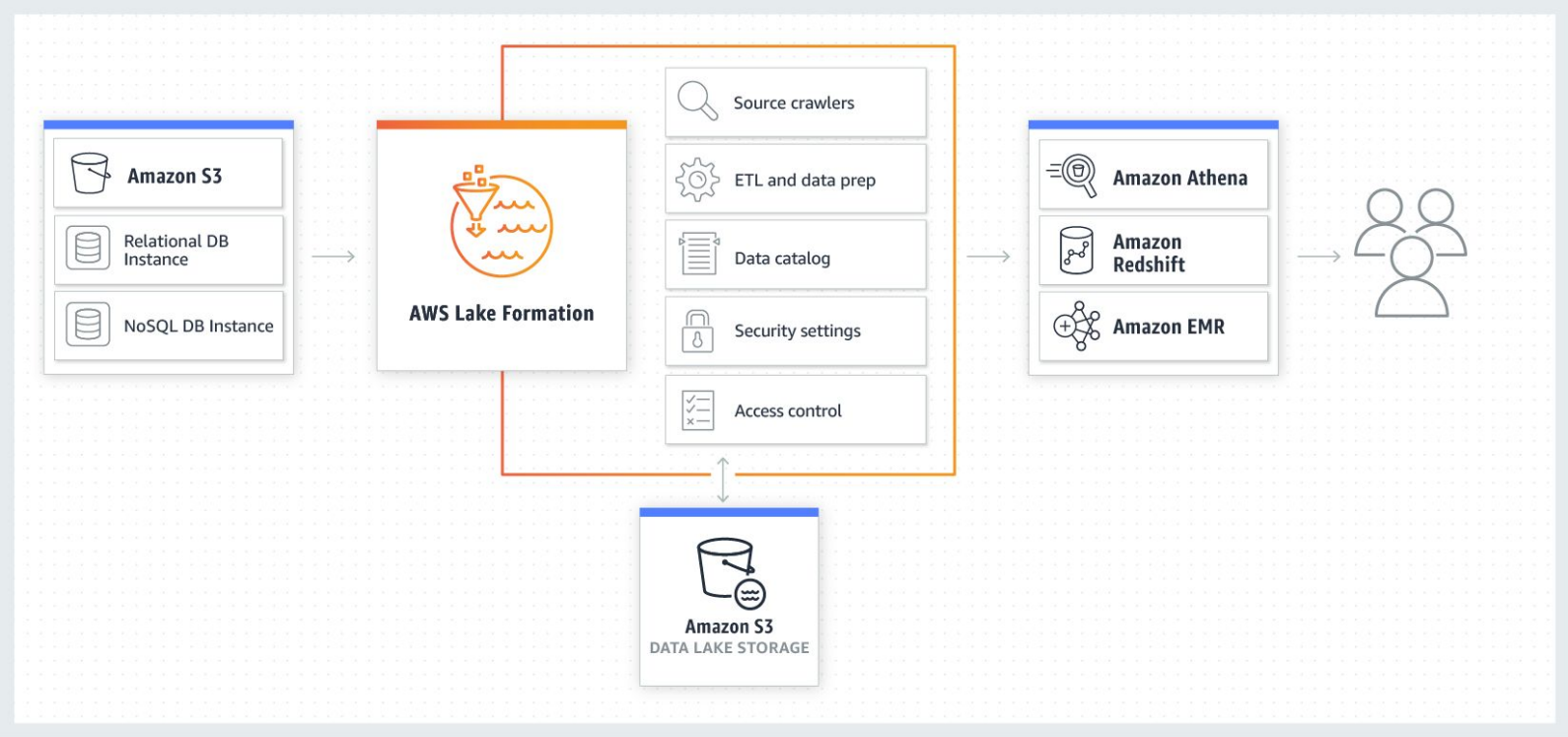
Stack

Soluções de Data Lakes



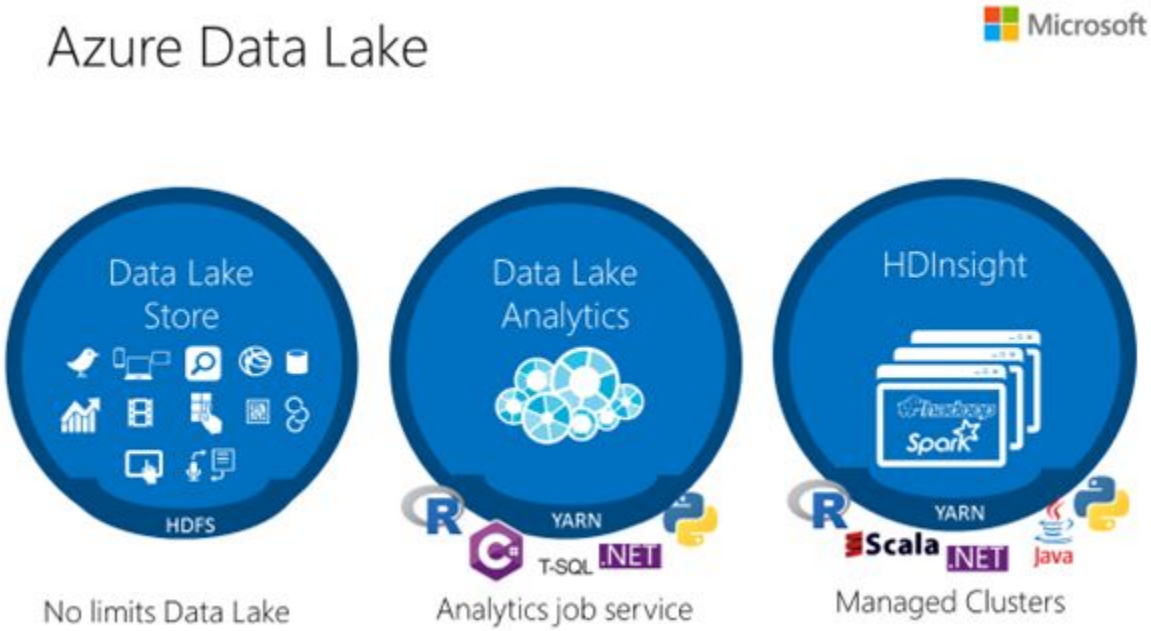
Stack

Soluções de Data Lakes



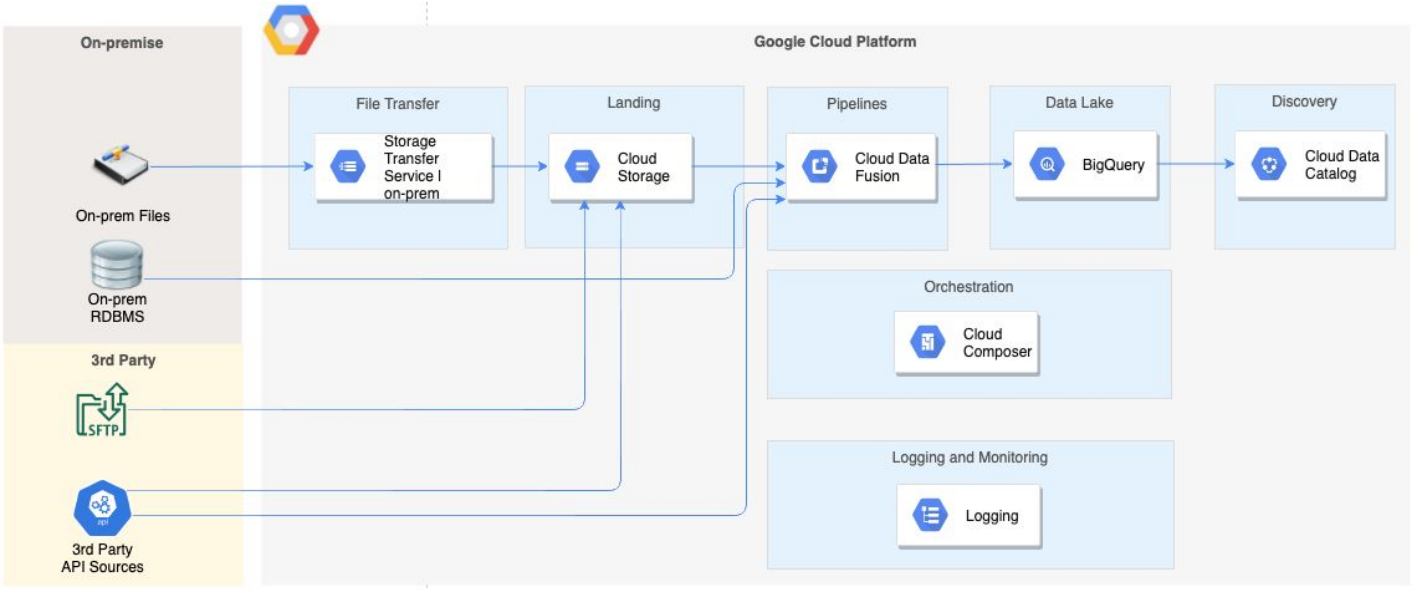
Stack

Soluções de Data Lakes



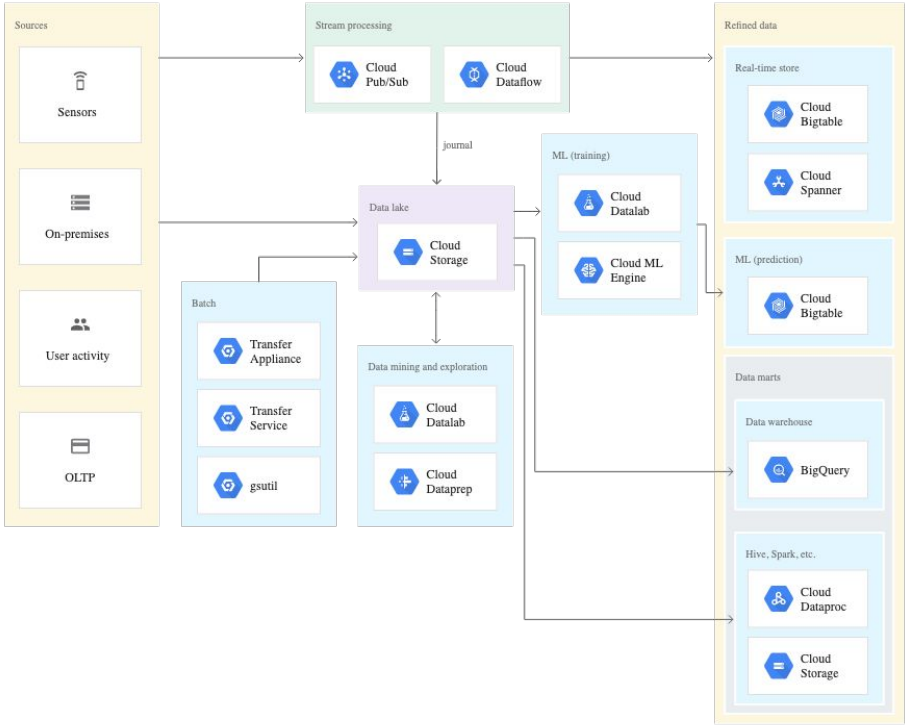
Stack

Soluções de Data Lakes



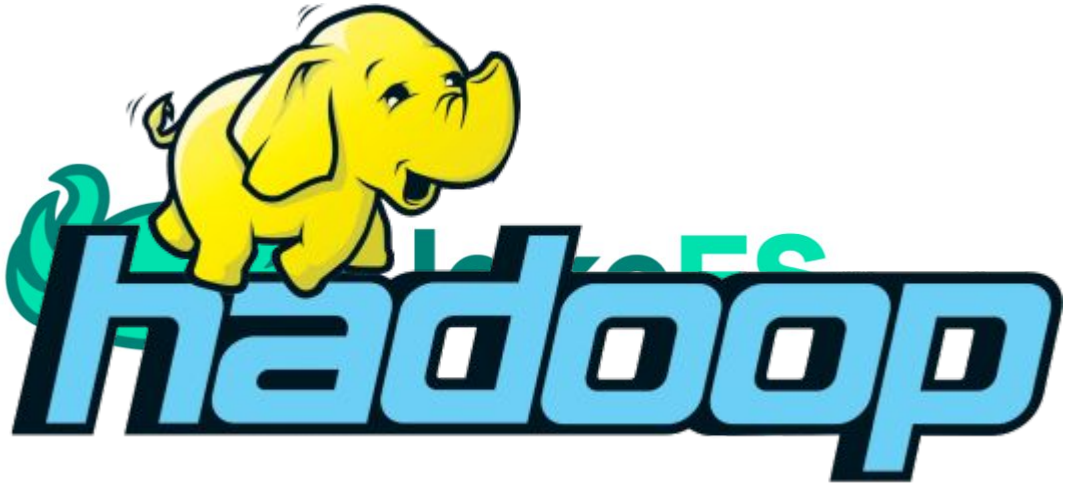
Stack

Soluções de Data Lakes



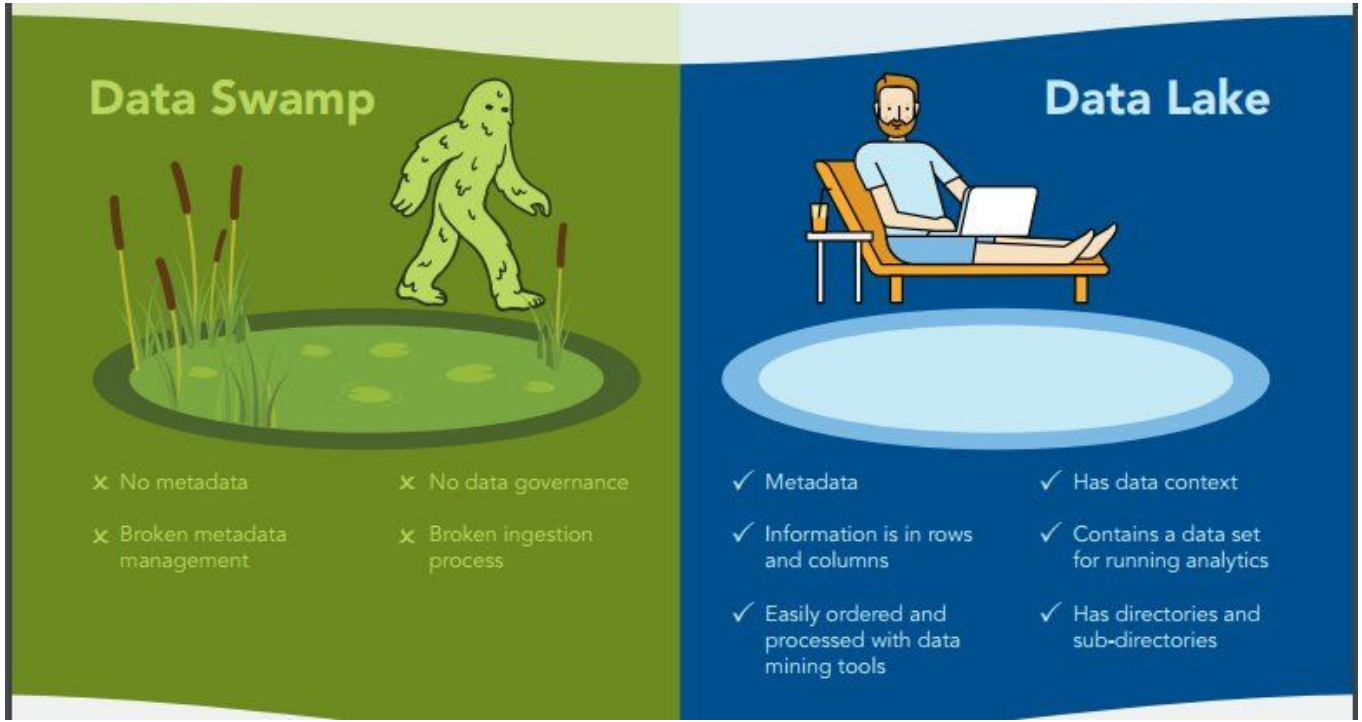
Stack

Soluções de Data Lakes Open Source



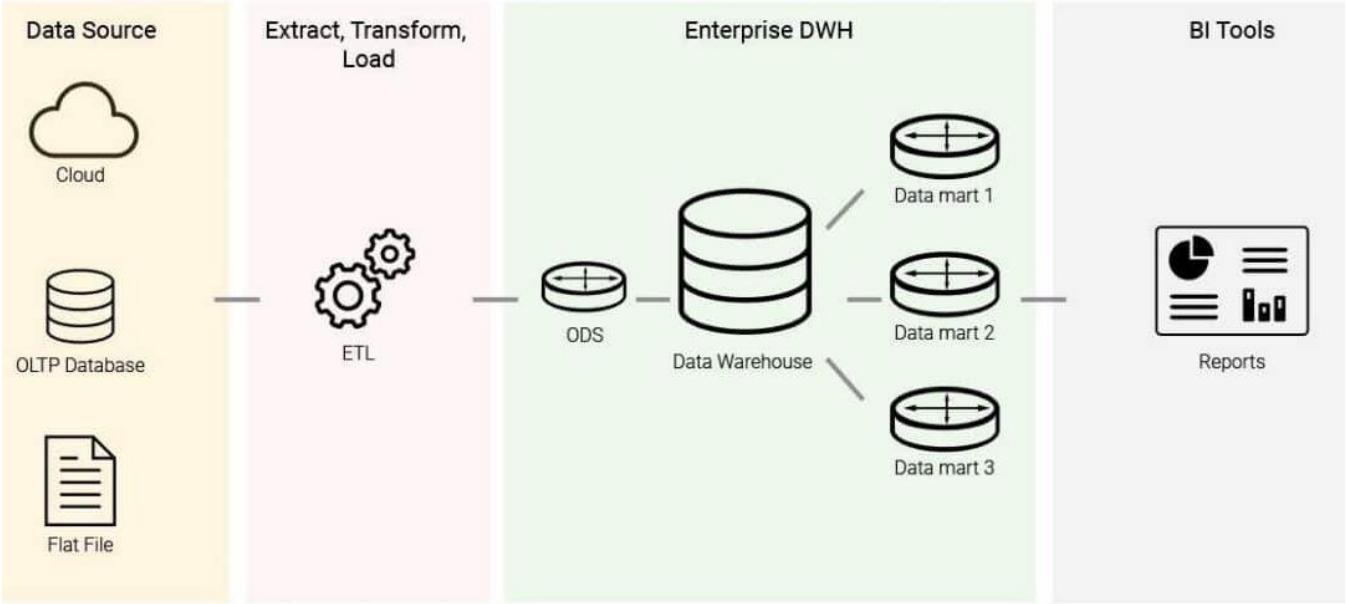
Stack

Data Swamp



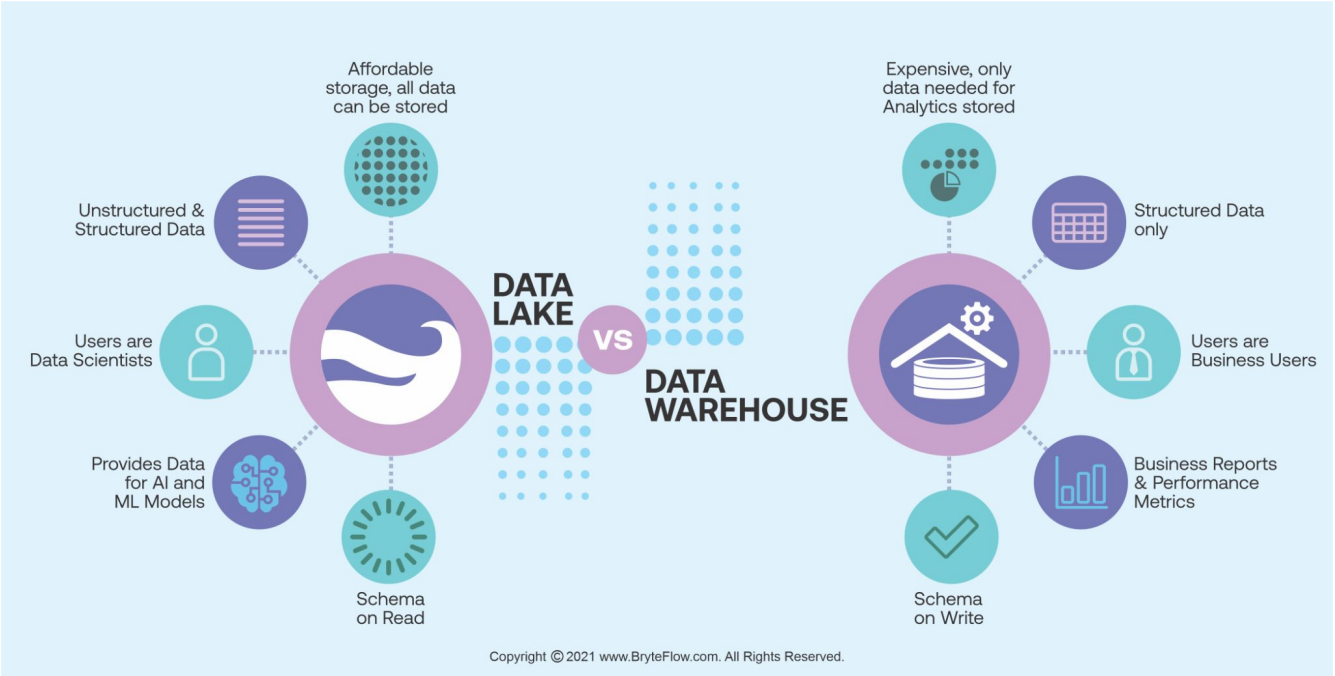
Stack

O que são Data Warehouses?



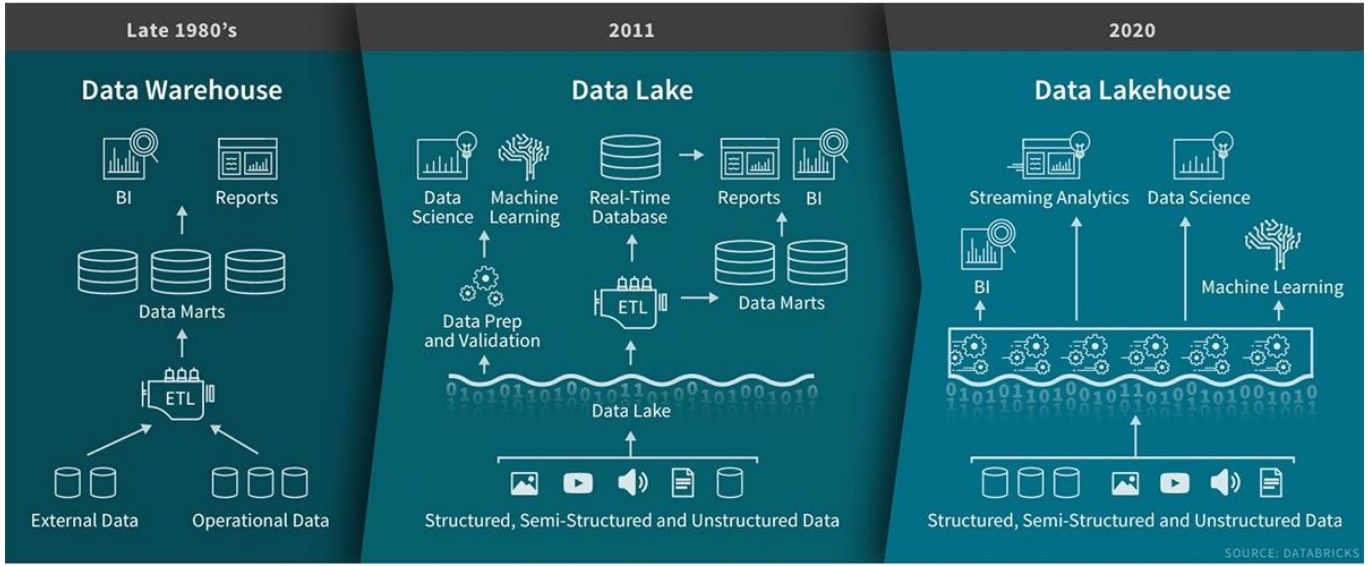
Stack

Data Warehouses Vs Data Lakes



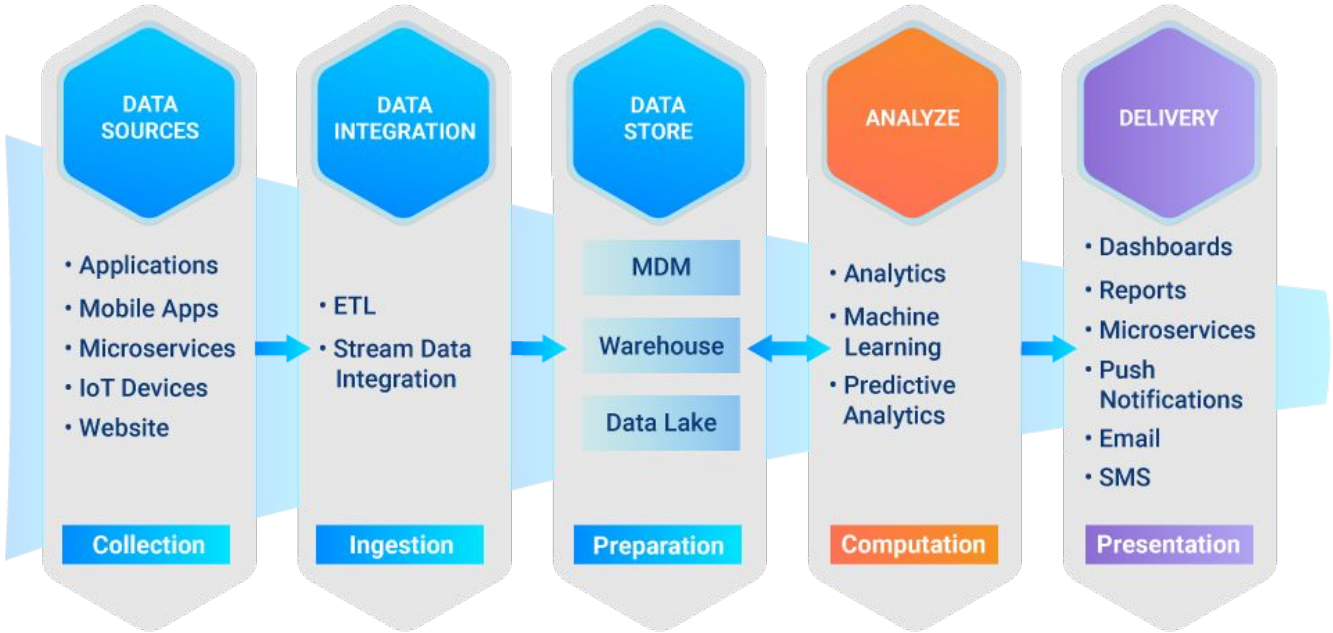
Stack

Data Warehouse vs Data Lake vs Data Lakehouse



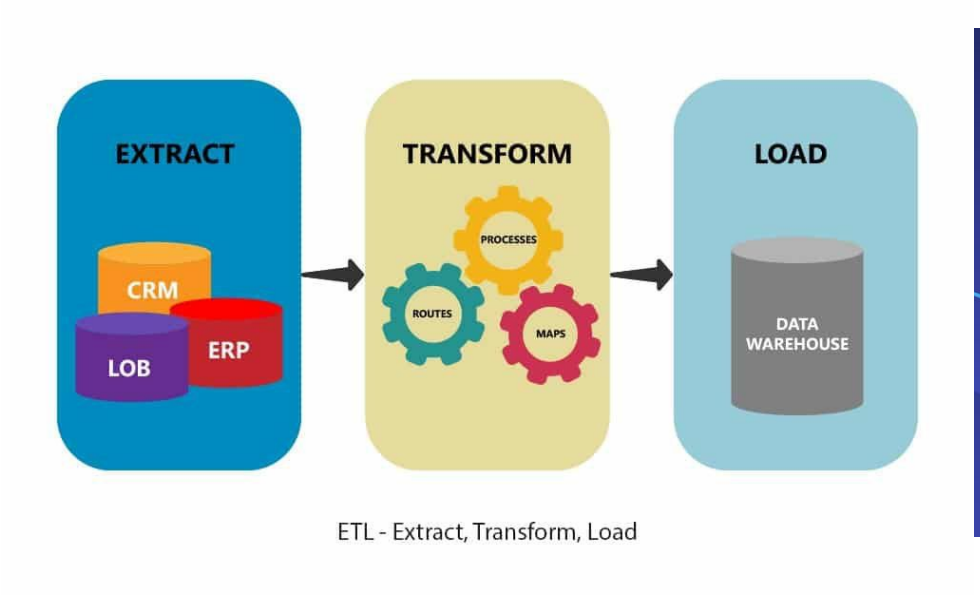
Stack

BIG DATA PIPELINE



Stack

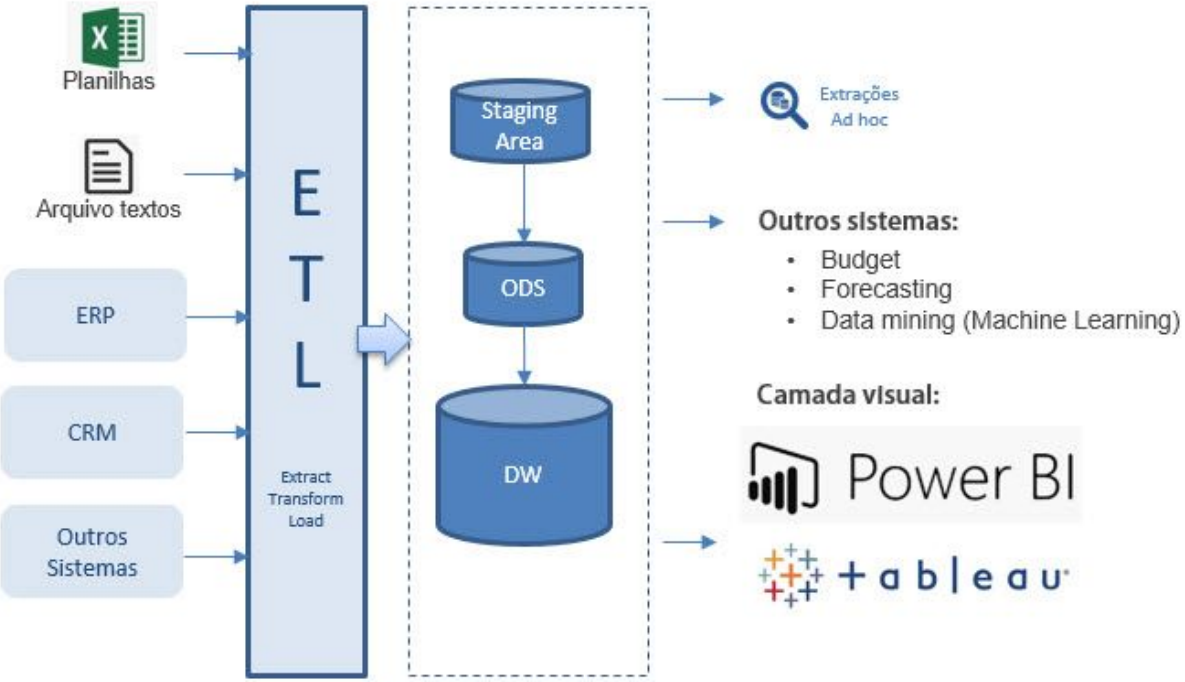
ETL vs ELT



ETL - Extract, Transform, Load

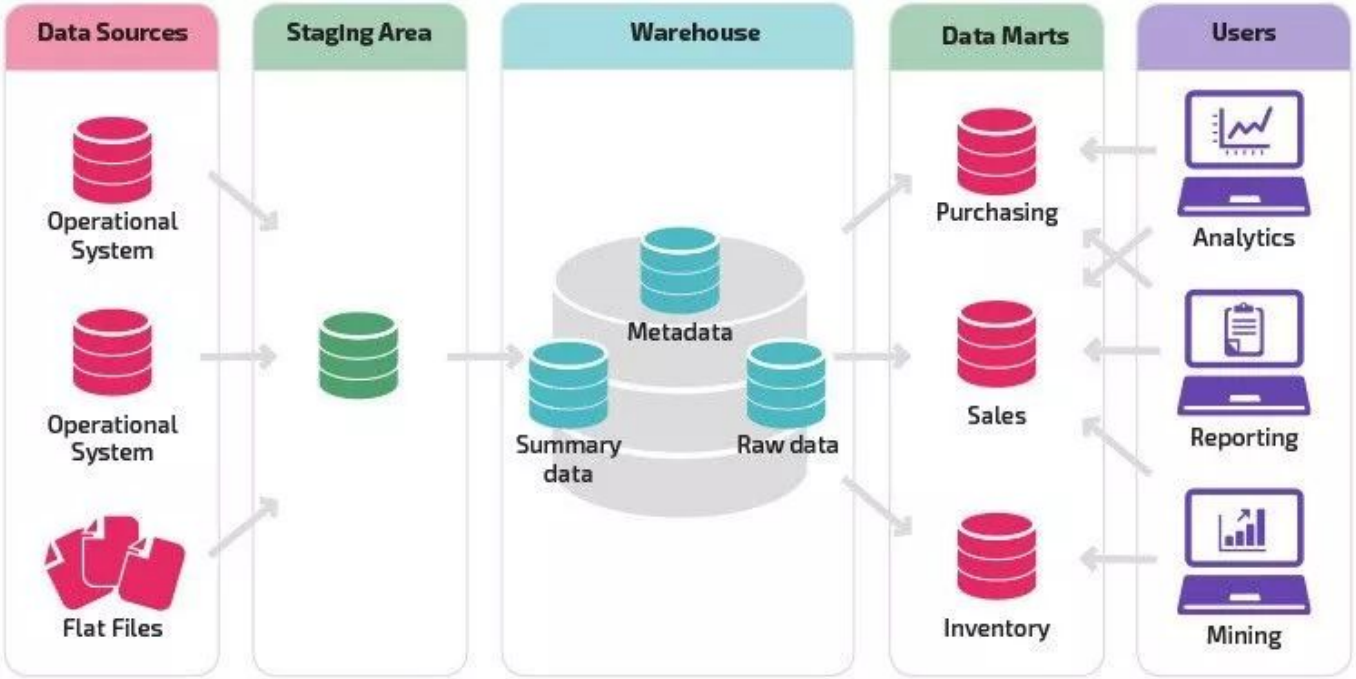
Stack

Extract Transform Loading



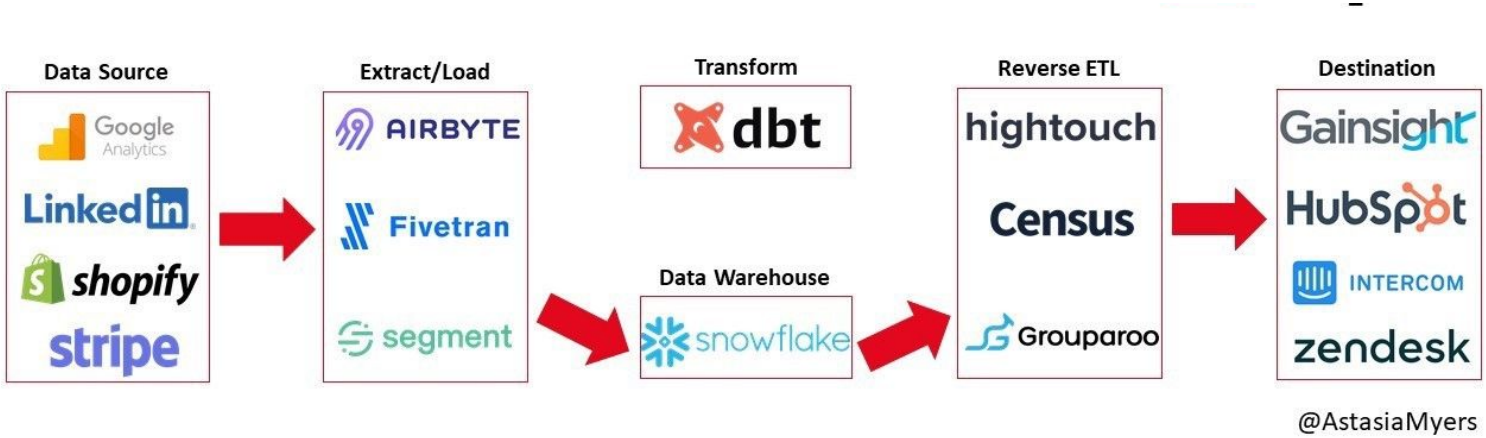
Stack

Extract Transform Loading - Data Marts



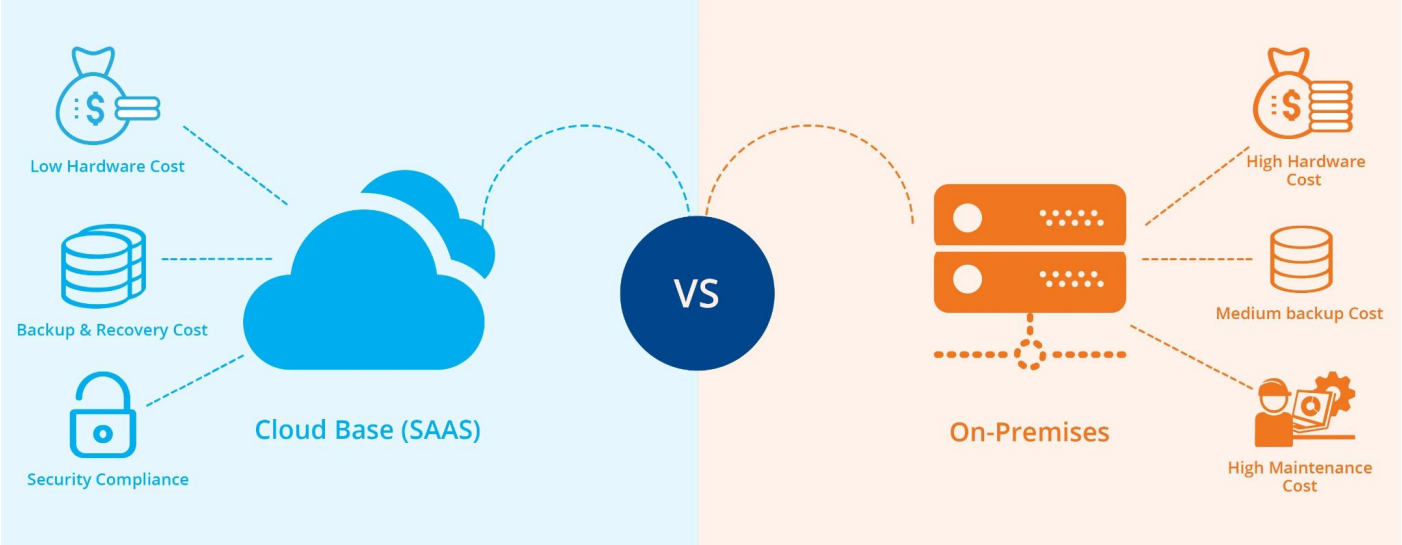
Stack

Soluções para ETL's



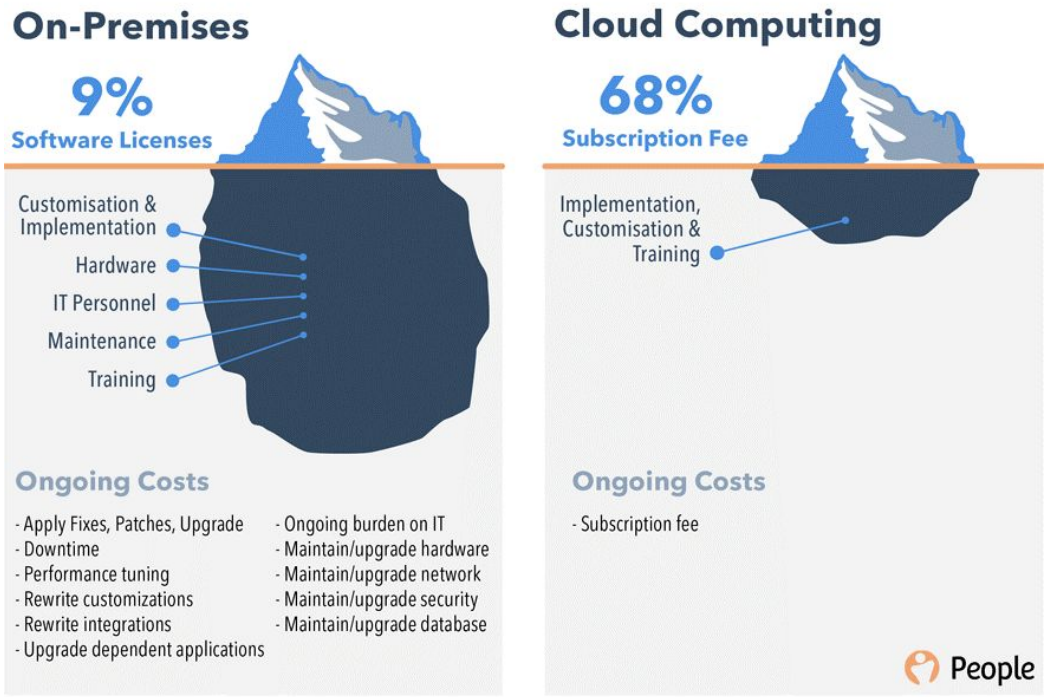
Stack

On-Premisses vs Cloud



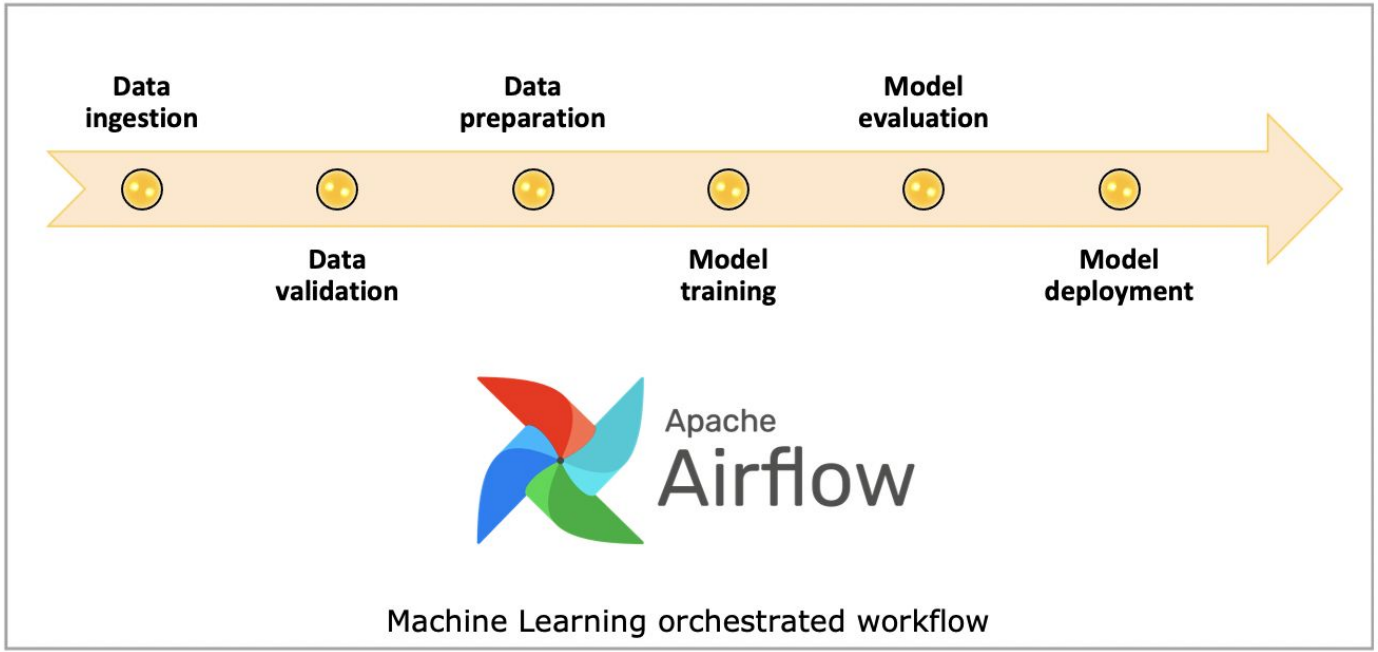
Stack

On-Premisses vs Cloud



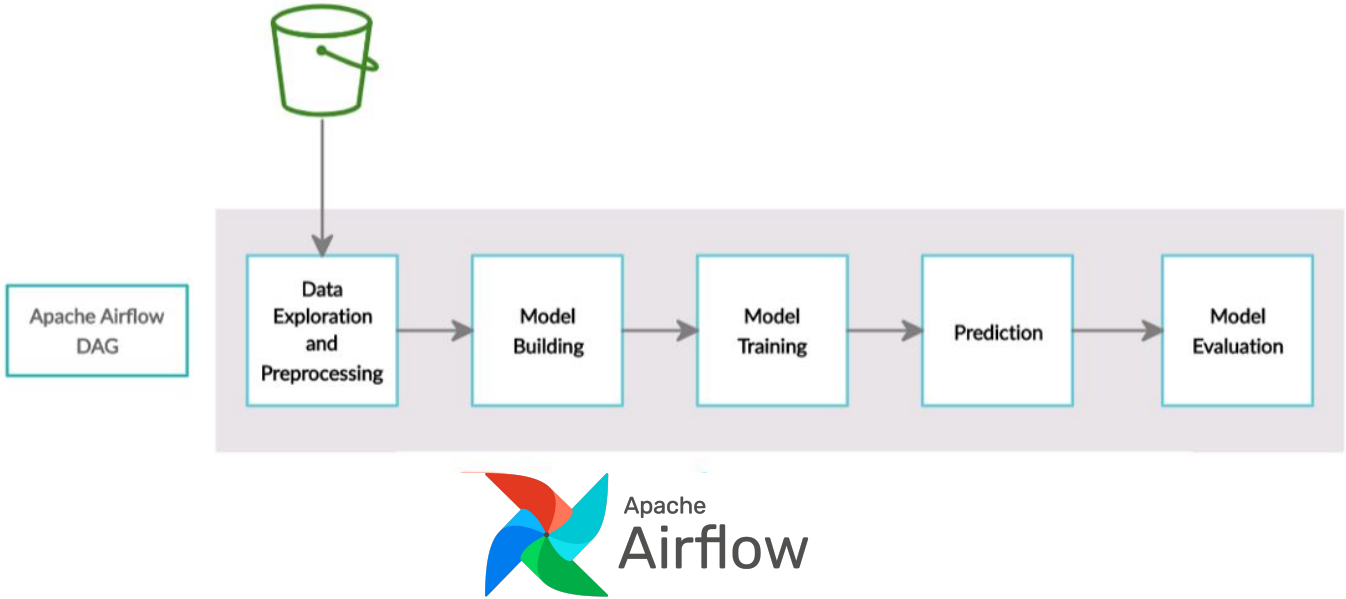
Stack

Soluções para Data Pipelines



Stack

Soluções para Data Pipelines

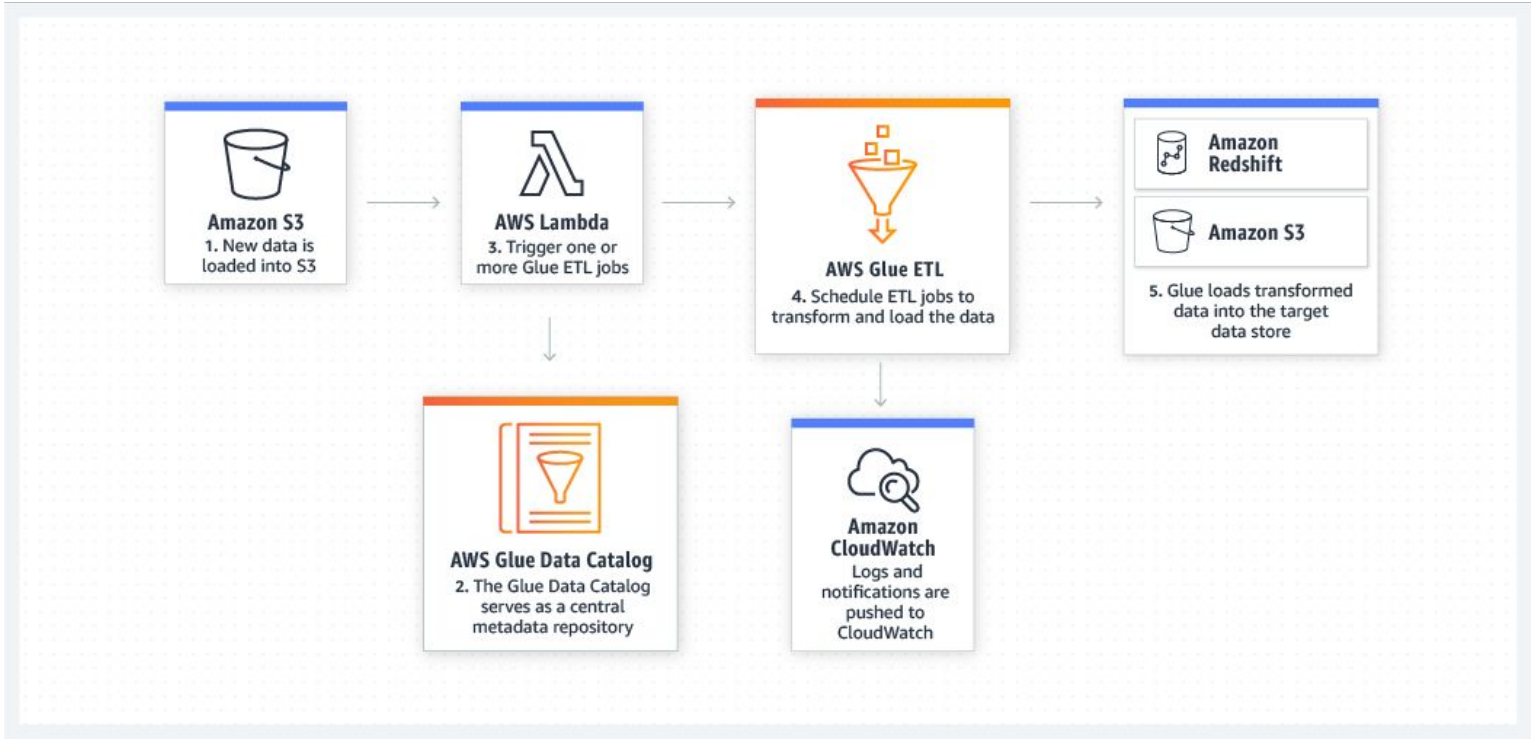


Stack

Soluções para Data Pipelines

Stack

Soluções para Data Pipelines



Stack

Soluções para Data Pipelines

Stack

Responsabilidades do Engenheiro de Dados para gerenciar um Data Pipeline

- Definir ambiente e arquitetura.
  - On-Premisse, On Cloud
- Especificar tecnologias para armazenamento.
  - Data Lakes, Data Warehouse.
- Definir formas de processamento.
  - Streaming, Clusters.
- Definir e implementar padrões e segurança e integração com outros sistemas.
- Monitoramento.

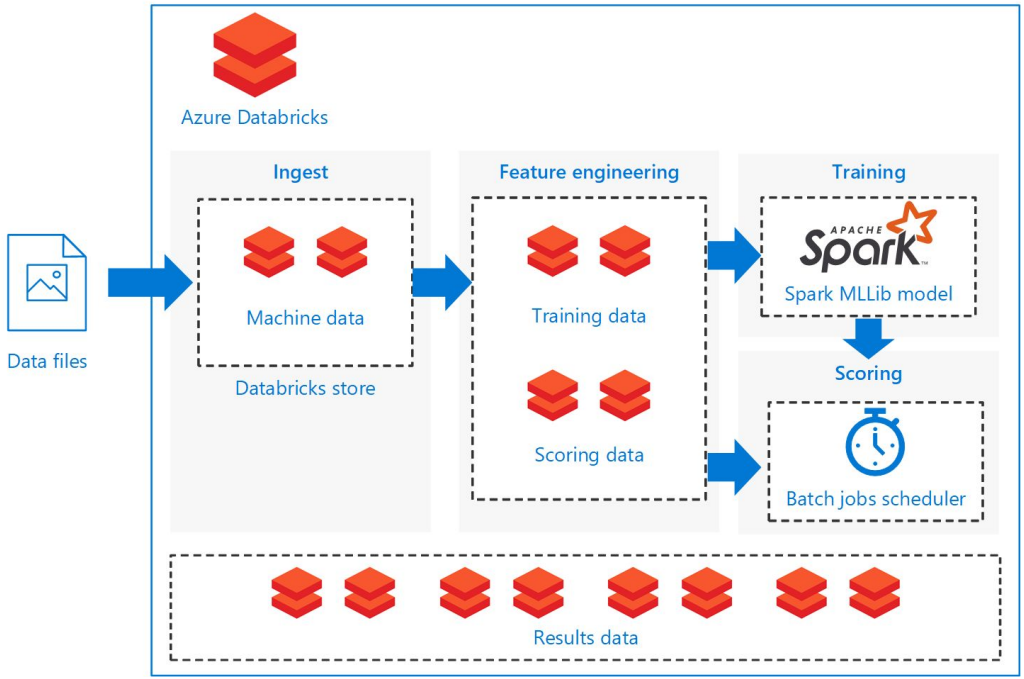
Stack

Data Engineering at World

databricks

Stack

Data Engineering at World

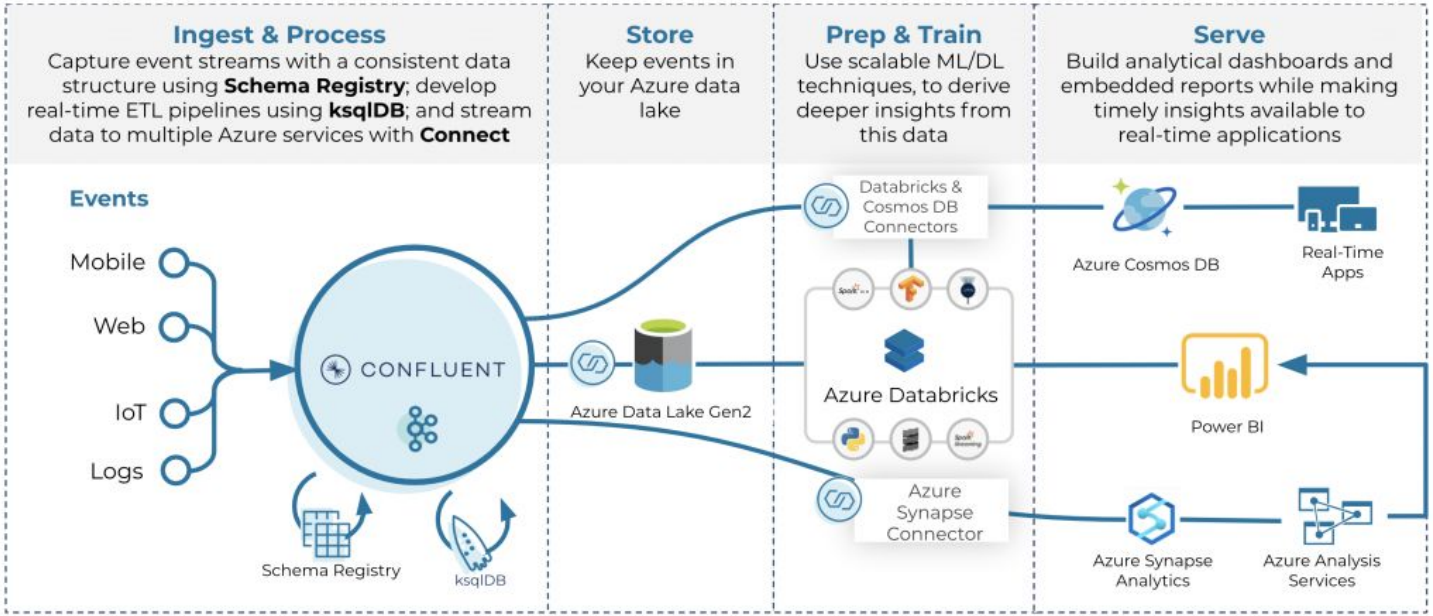


Stack

Data Engineering at World

Stack

Data Engineering at World

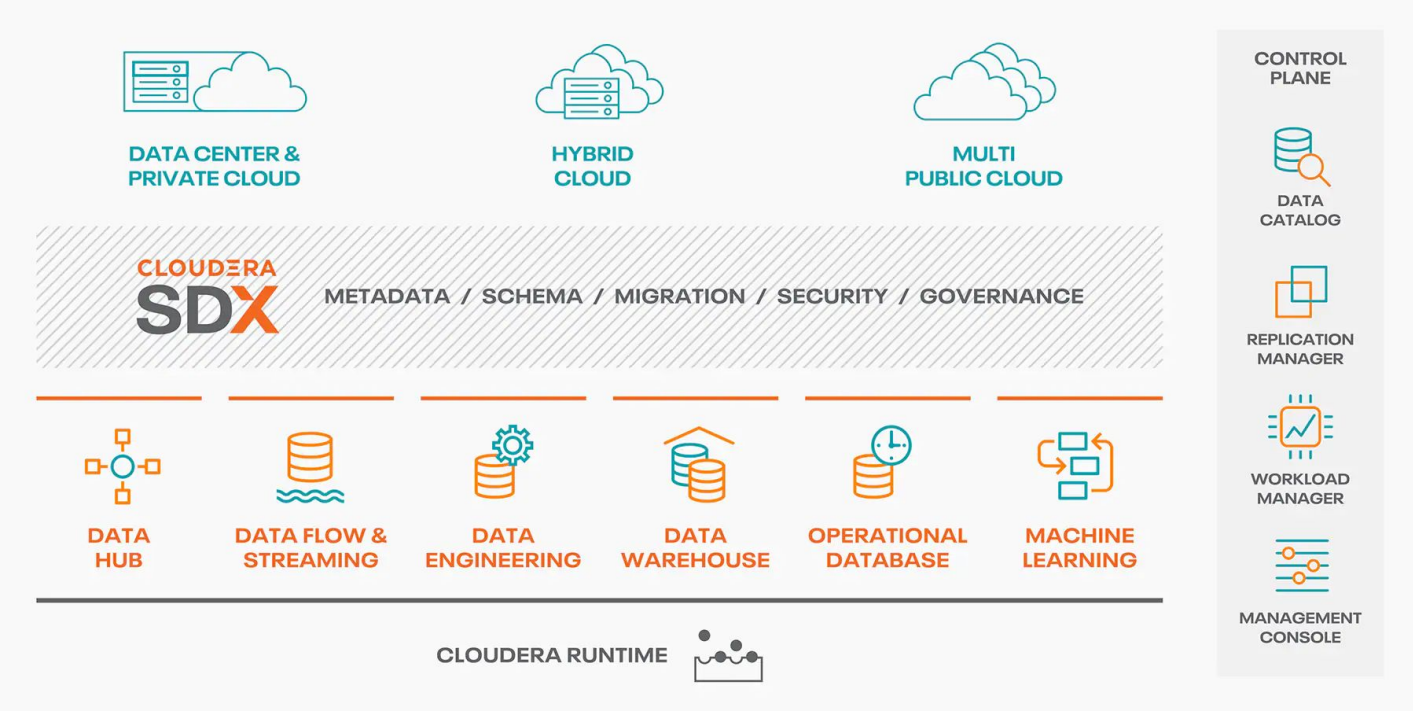


Stack

Data Engineering at World

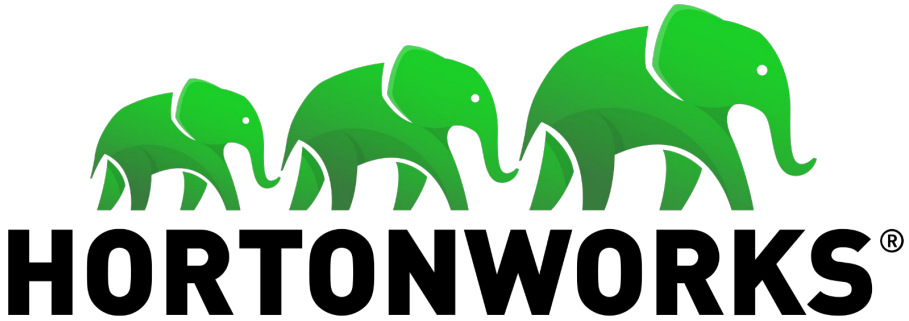
Stack

Data Engineering at World



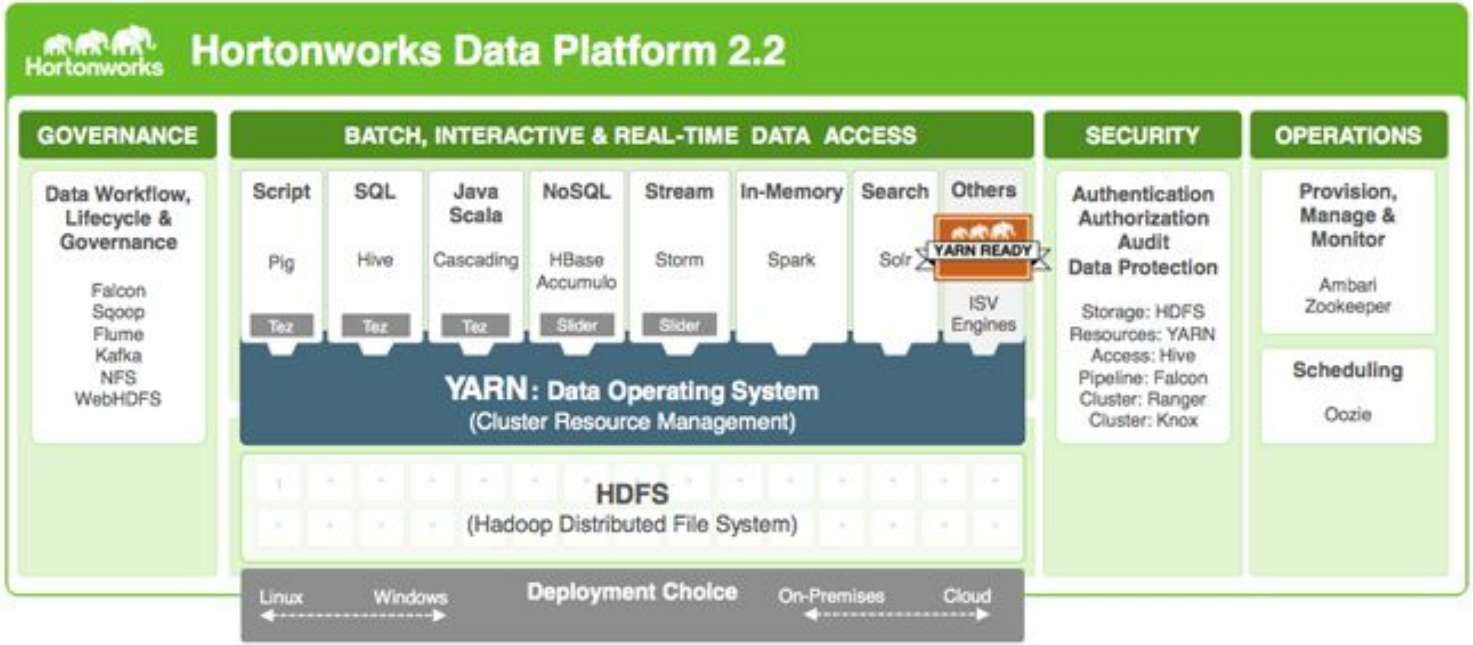
Stack

Data Engineering at World



Stack

Data Engineering at World

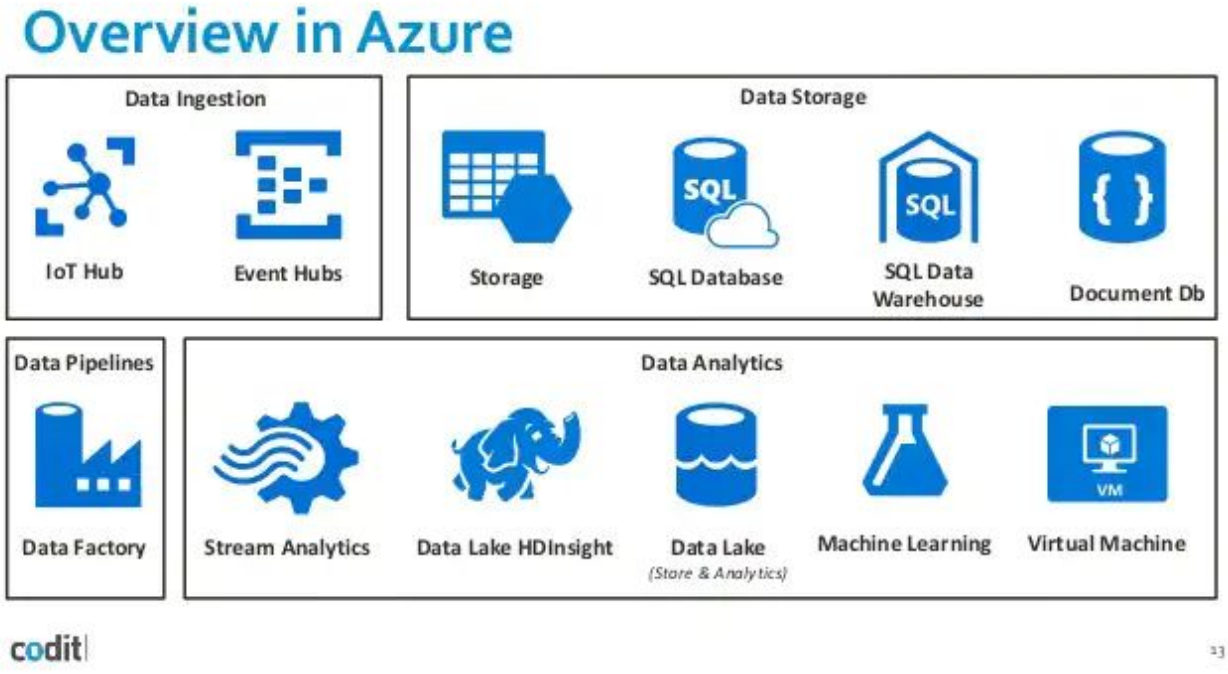


Stack

Data Engineering at World

Stack

Data Engineering at World

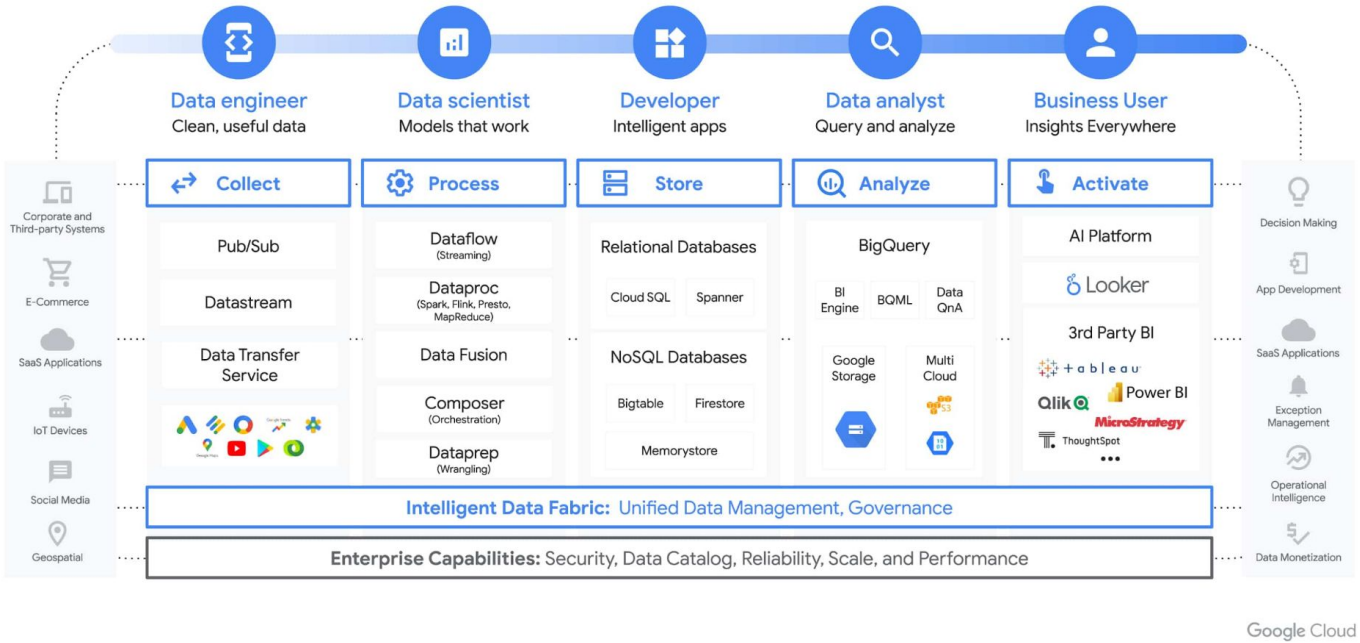


Stack

Data Engineering at World

Stack

Data Engineering at World



Stack

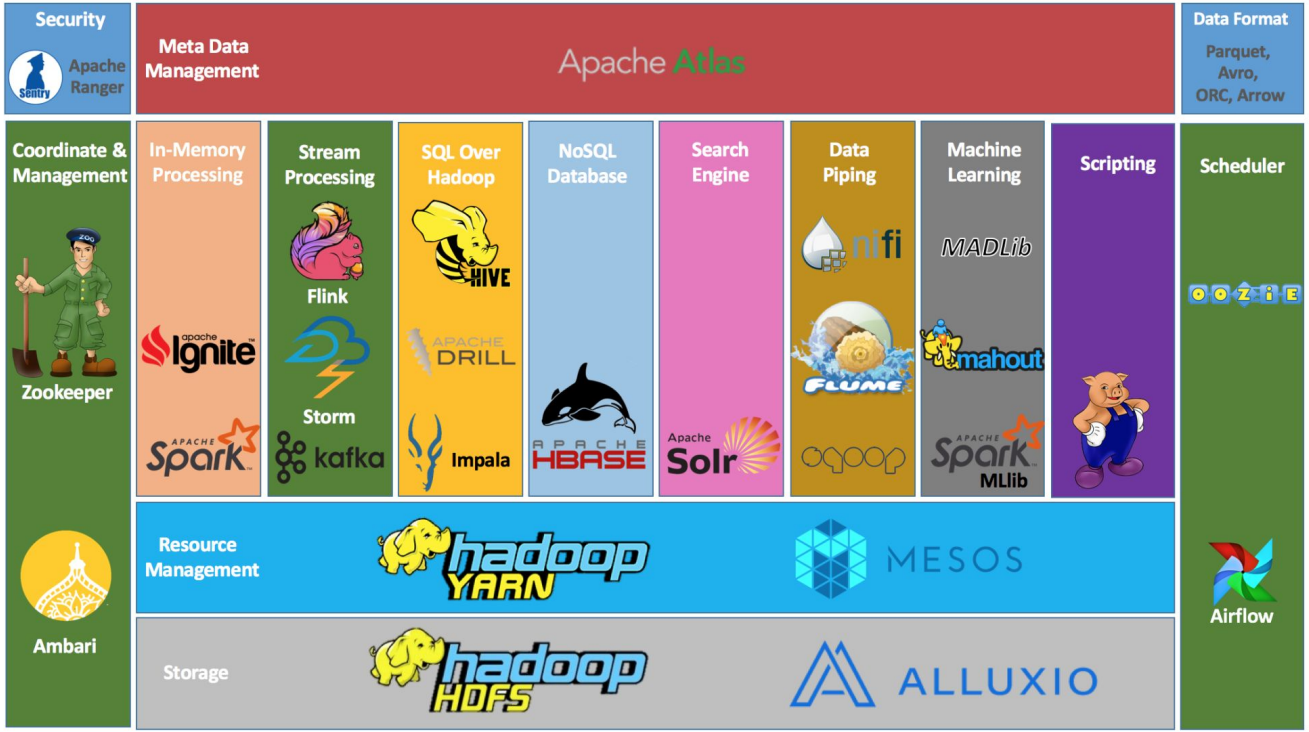
Data Engineering at World

Stack

Data Engineering at World

Stack

Ecossistema Hadoop

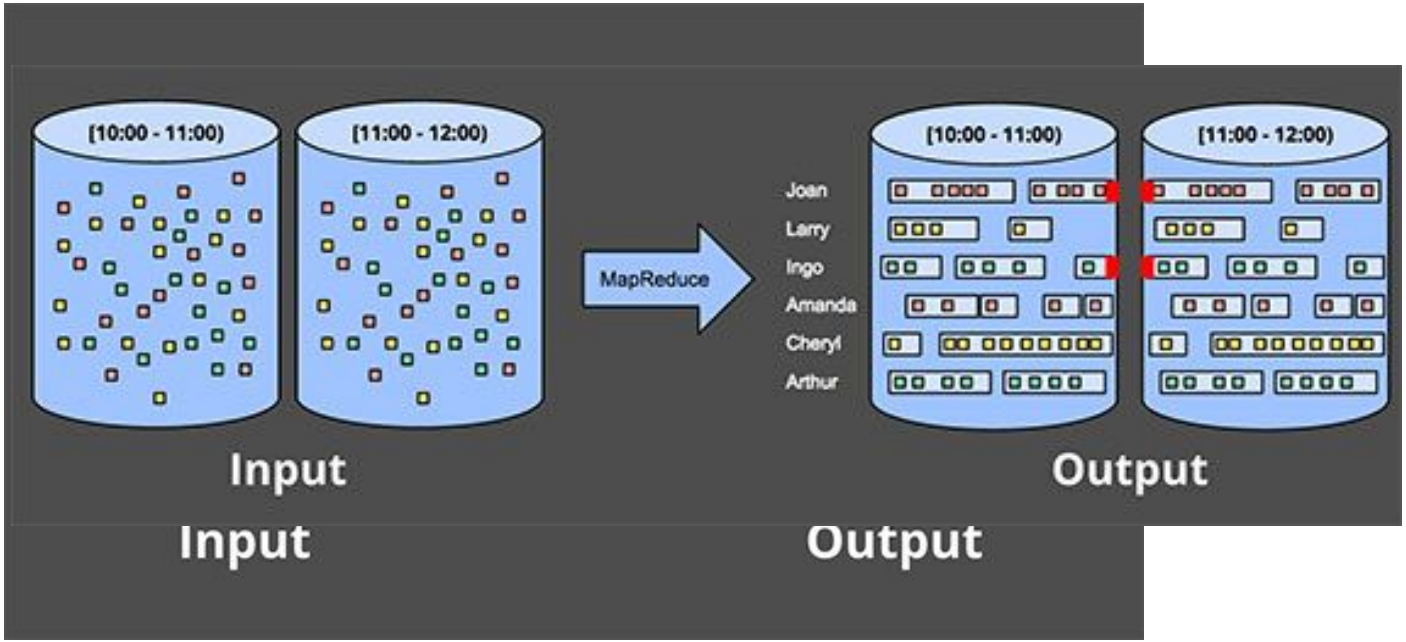


Stack

Processamento Batch

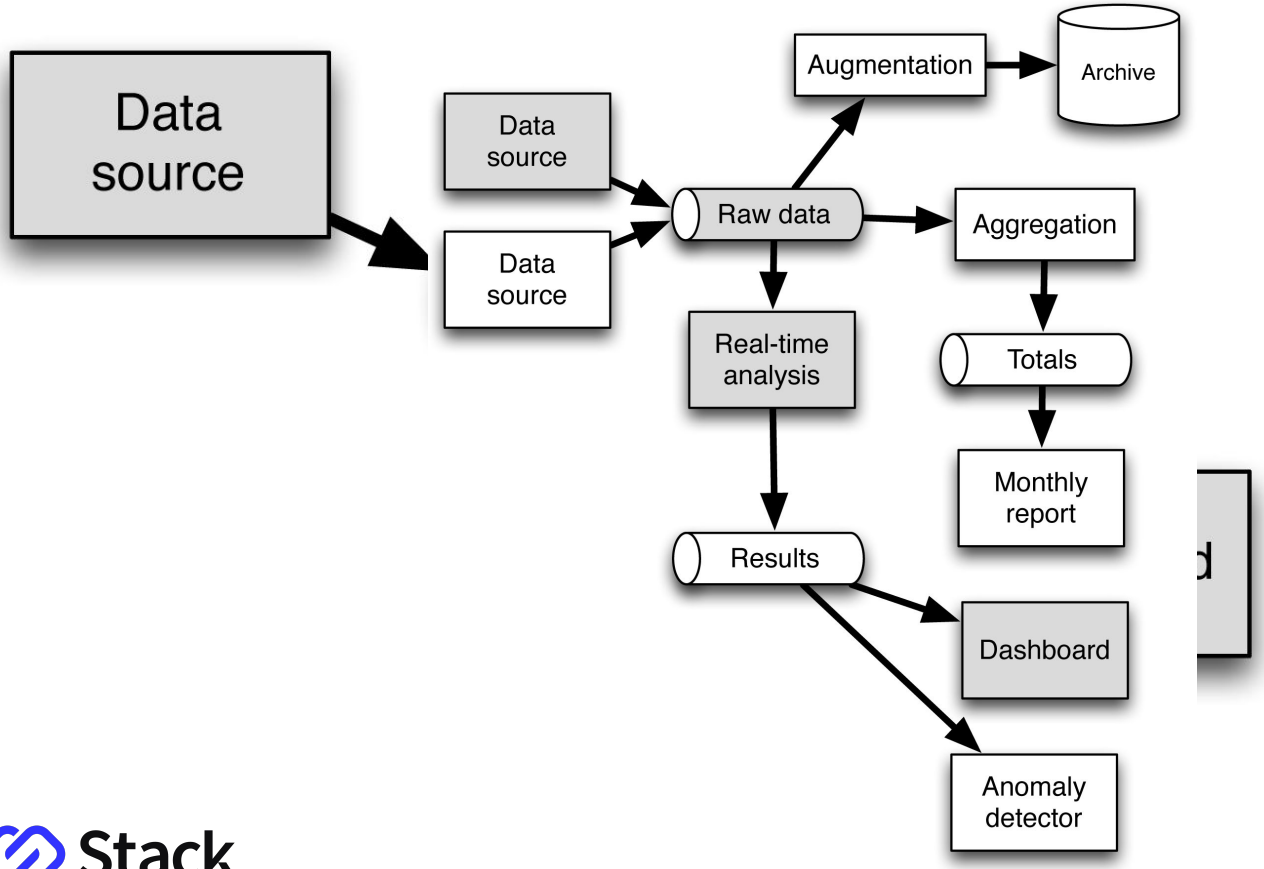
Stack

Processamento Streaming



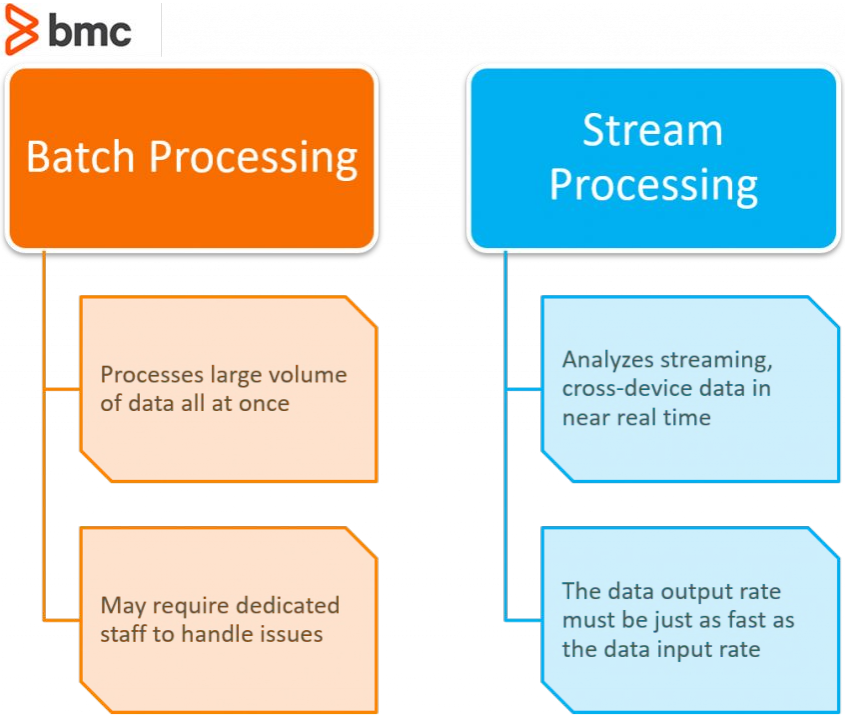
Stack

Processamento Streaming



Stack

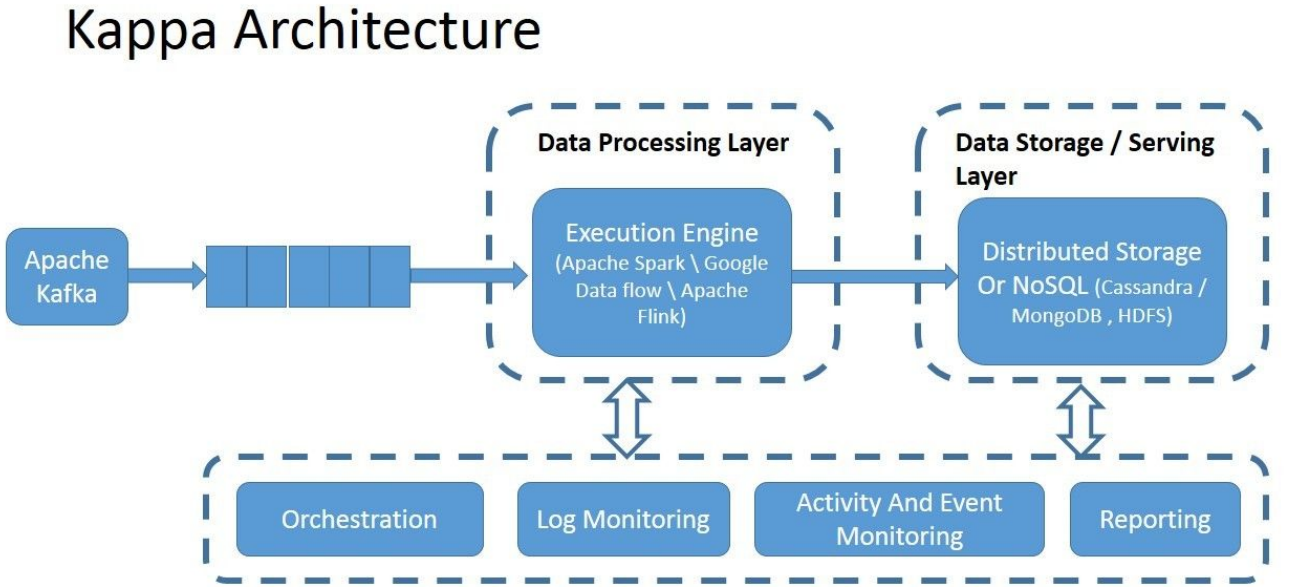
Batch Processing vs Streaming Processing



Benefits and drawbacks of common data processing types

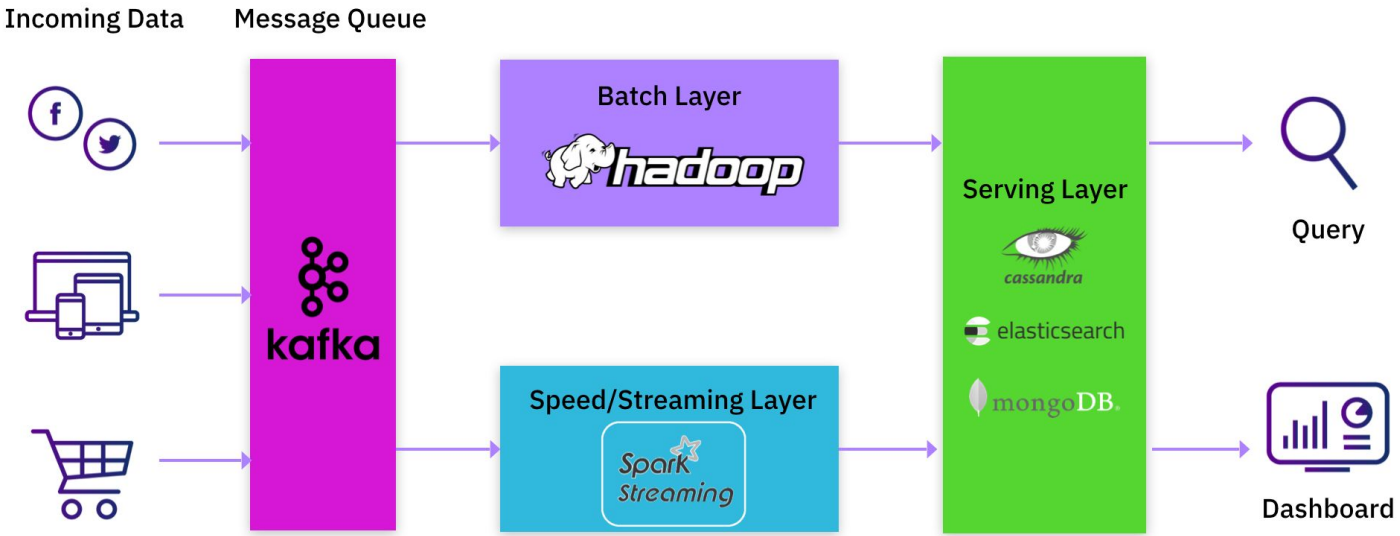
Stack

Arquitetura Kappa



Stack

Arquitetura lambda



Stack

Lambda vs Kappa

		
Processing paradigm	Batch + Streaming	Streaming
Re-processing paradigm	Every Batch cycle	Only when code changes
Resource consumption	Function = Query (All data)	Incremental algorithms, running on deltas
Reliability	Batch is reliable, Streaming is approximate	Streaming with consistency (exactly once)

Stack

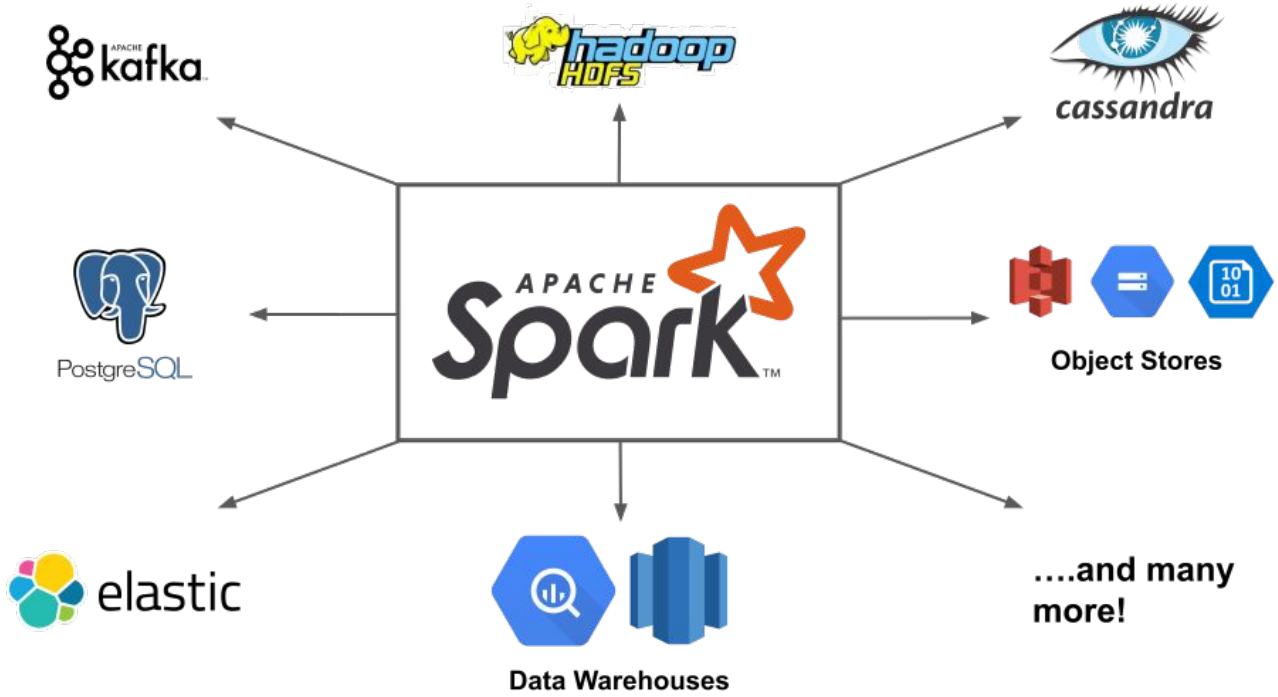
O que é o Apache Spark?

Stack

O que é o Cluster?

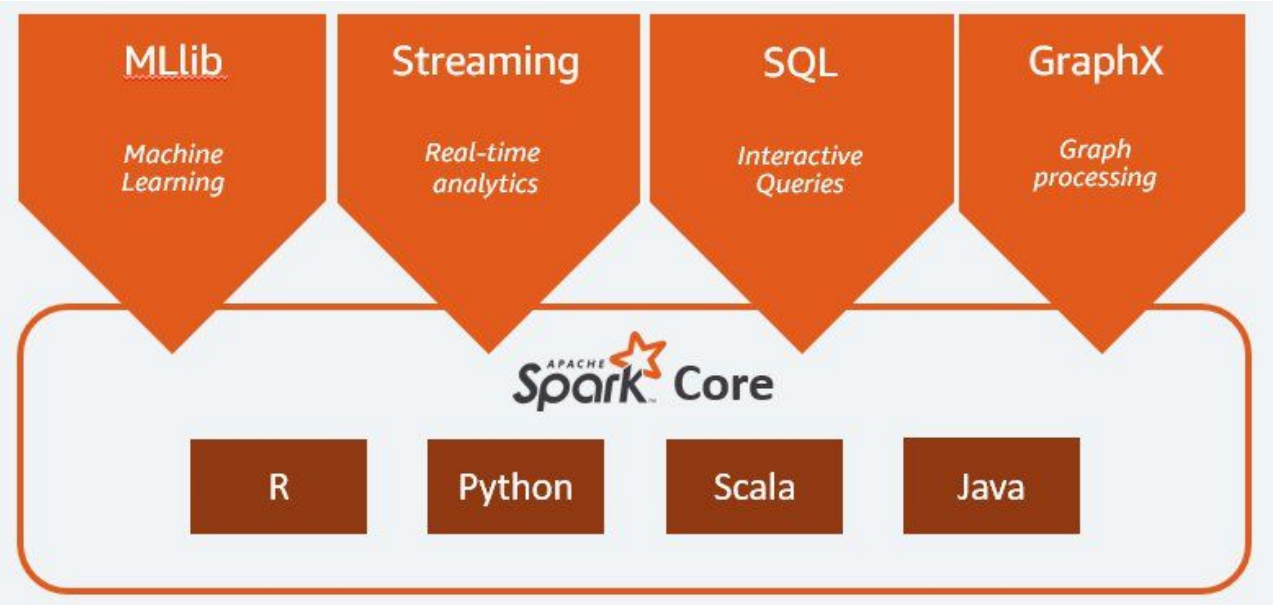
Stack

O que é o Apache Spark?



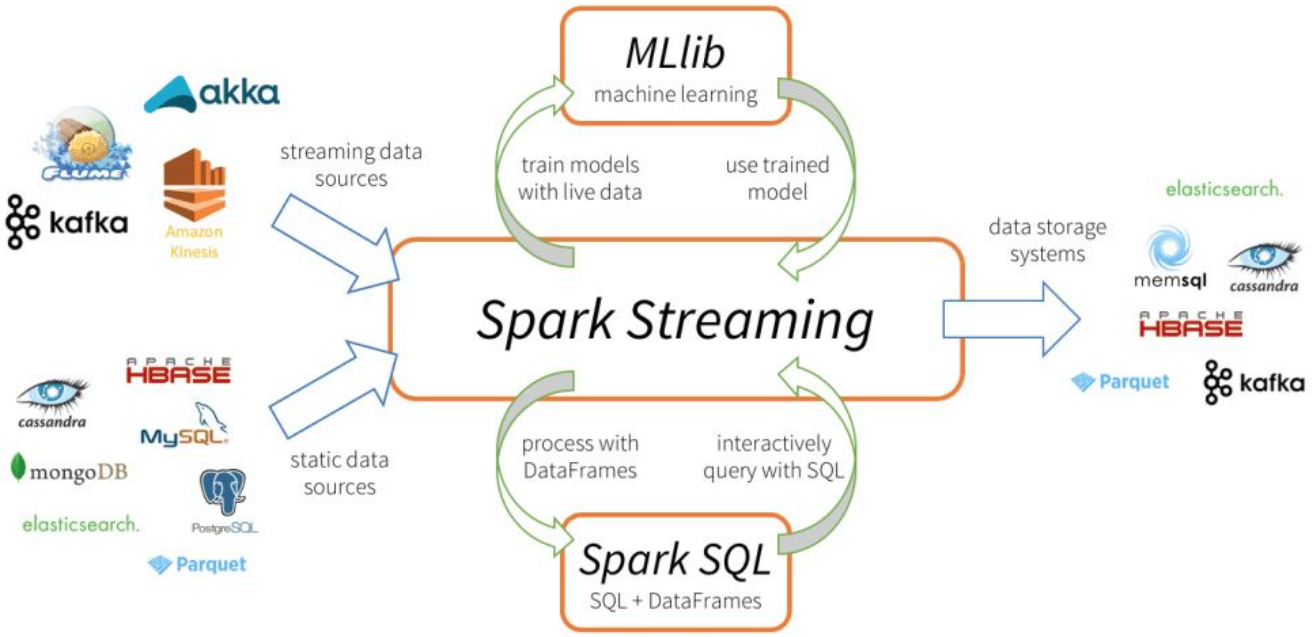
Stack

O que é o Apache Spark?



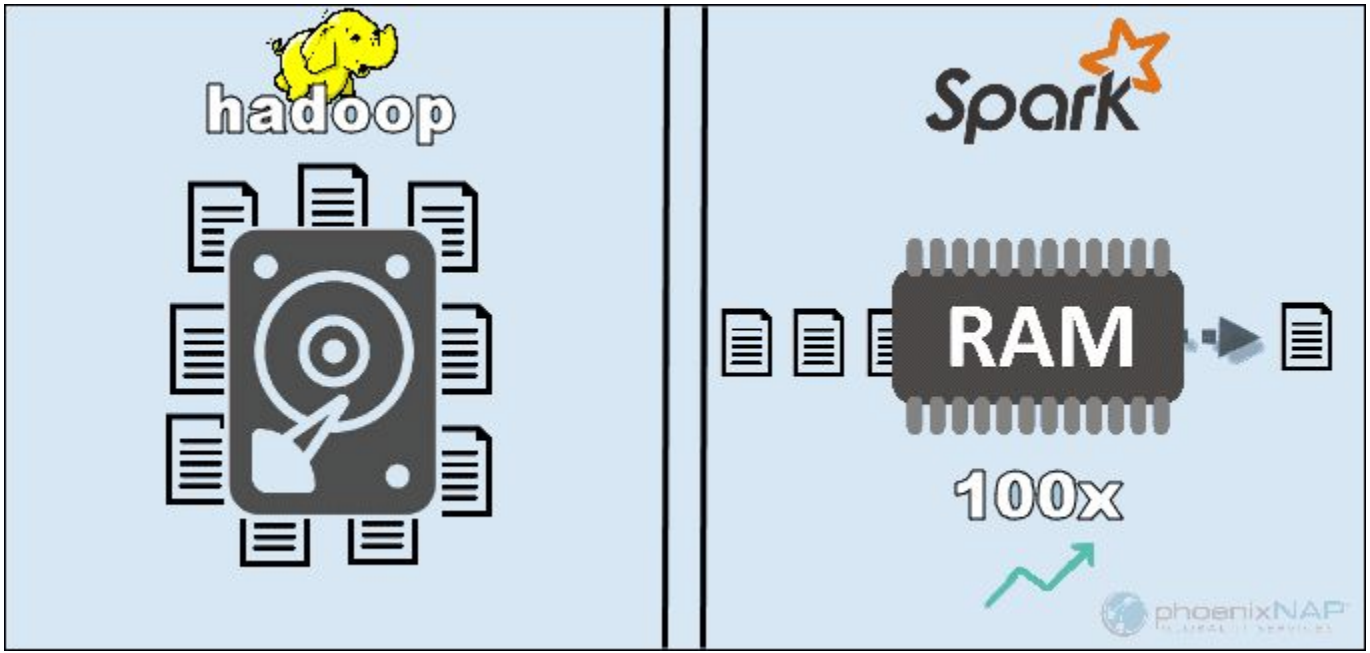
Stack

O que é o Apache Spark?



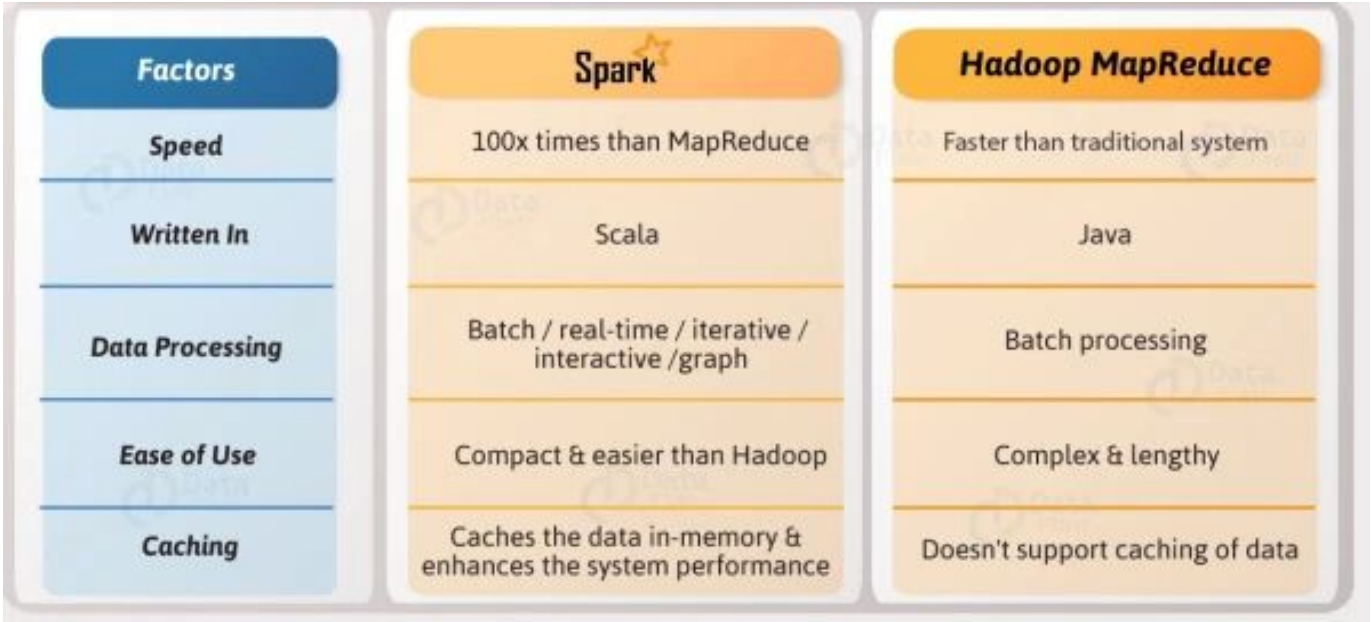
Stack

Spark Vs Hadoop



Stack

Spark Vs Hadoop



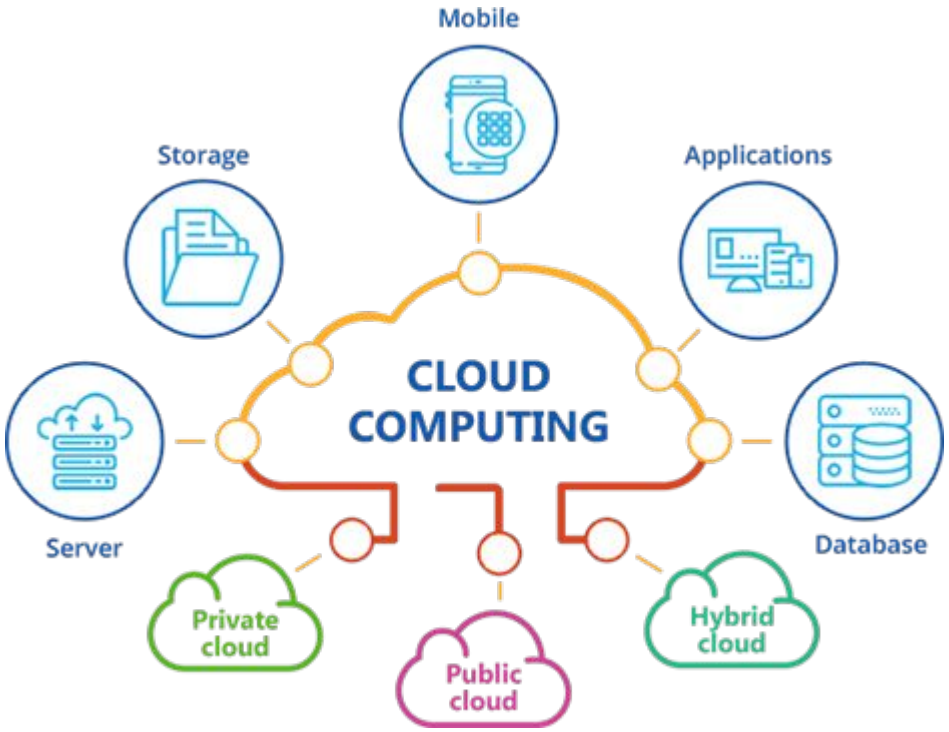
Factors	Spark	Hadoop MapReduce
Speed	100x times than MapReduce	Faster than traditional system
Written In	Scala	Java
Data Processing	Batch / real-time / iterative / interactive /graph	Batch processing
Ease of Use	Compact & easier than Hadoop	Complex & lengthy
Caching	Caches the data in-memory & enhances the system performance	Doesn't support caching of data

Stack

O que é Databricks?

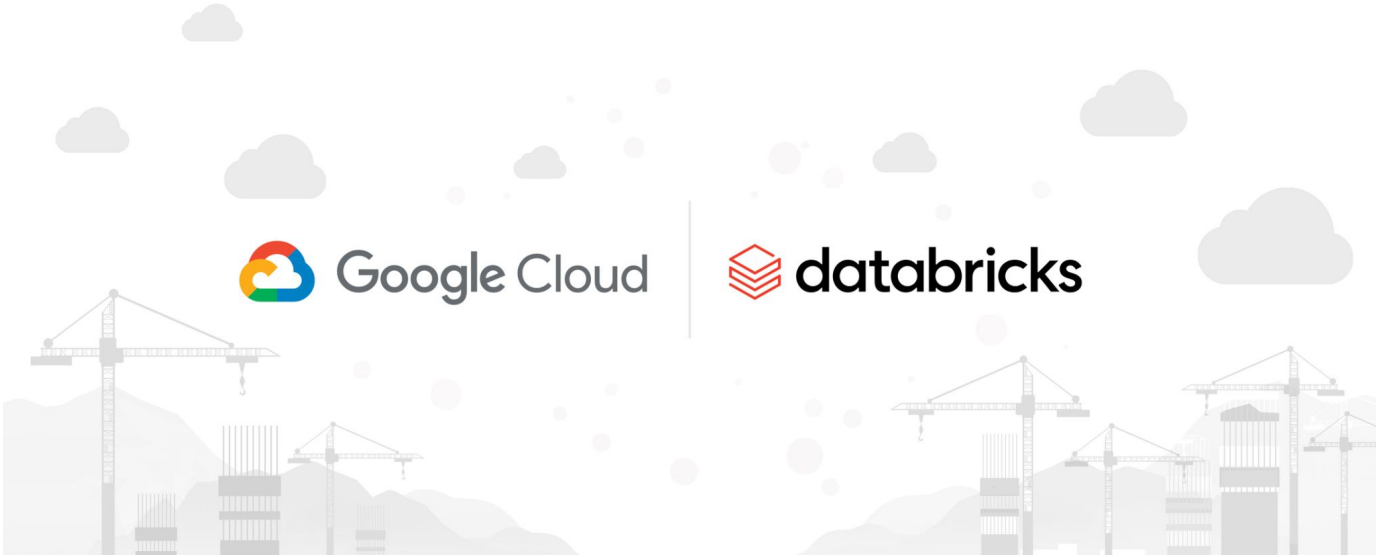
Stack

Por que Databricks?



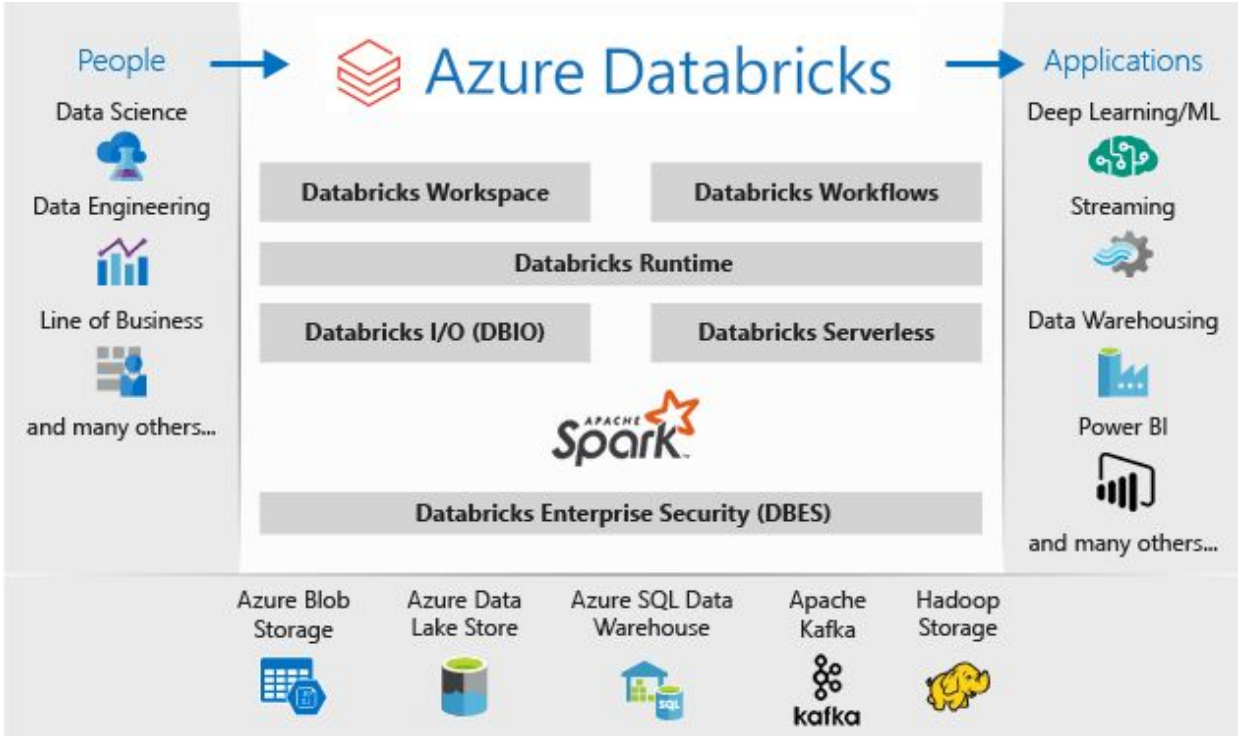
Stack

Por que Databricks?



Stack

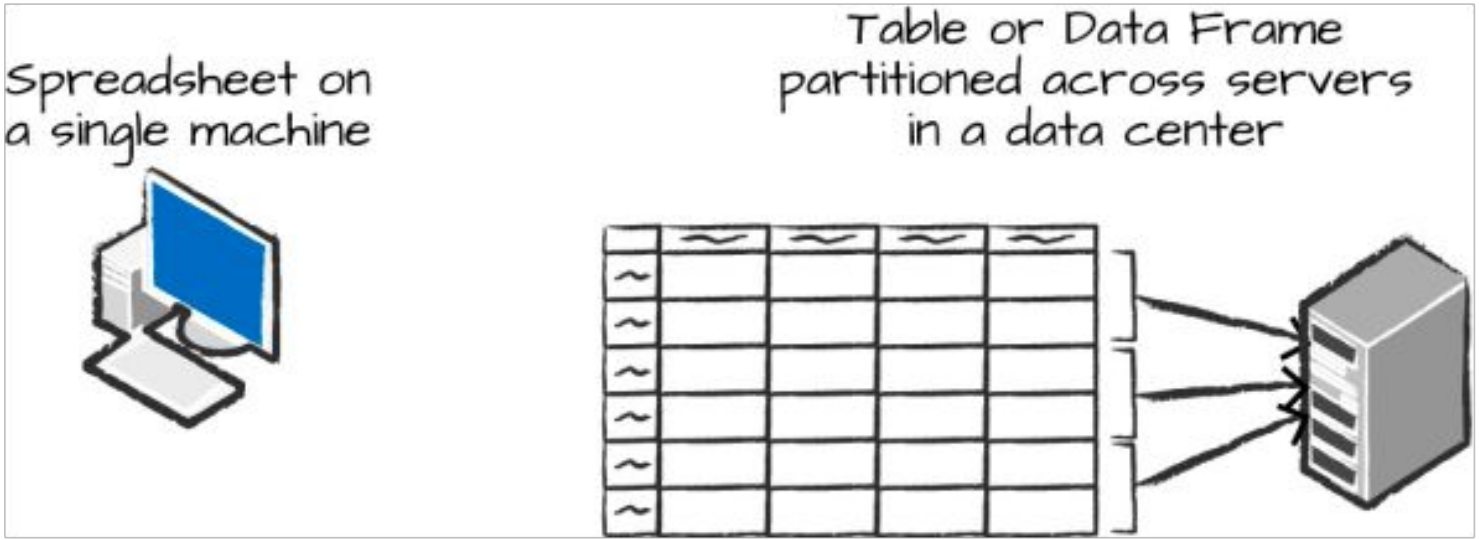
Por que Databricks?



Stack

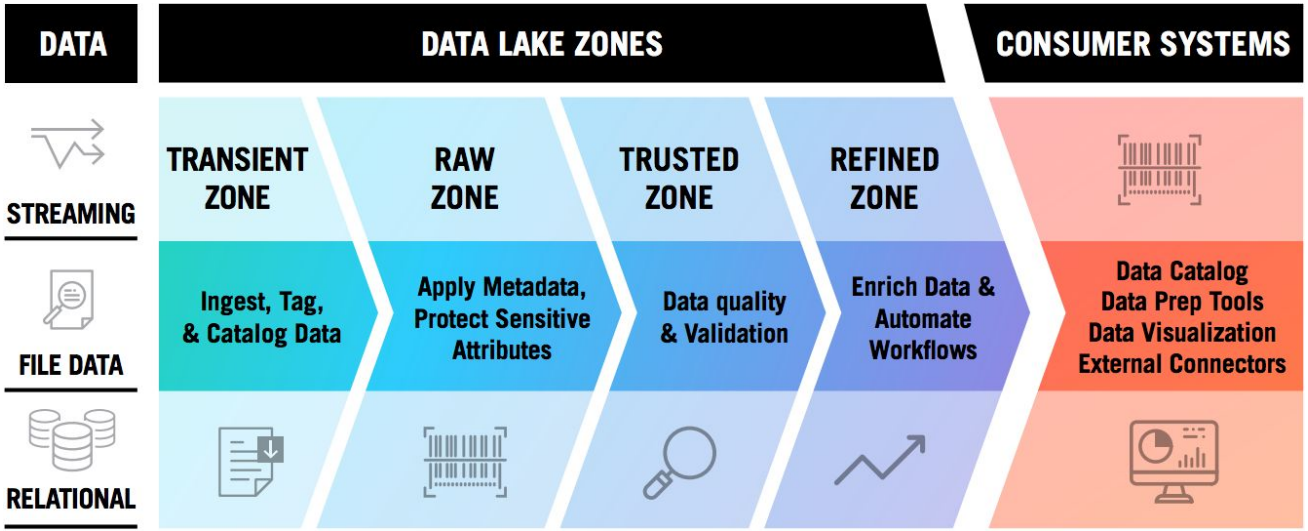
Hands on!

Dataframes e Partições no Spark



Stack

Dataframes e Partições no Spark

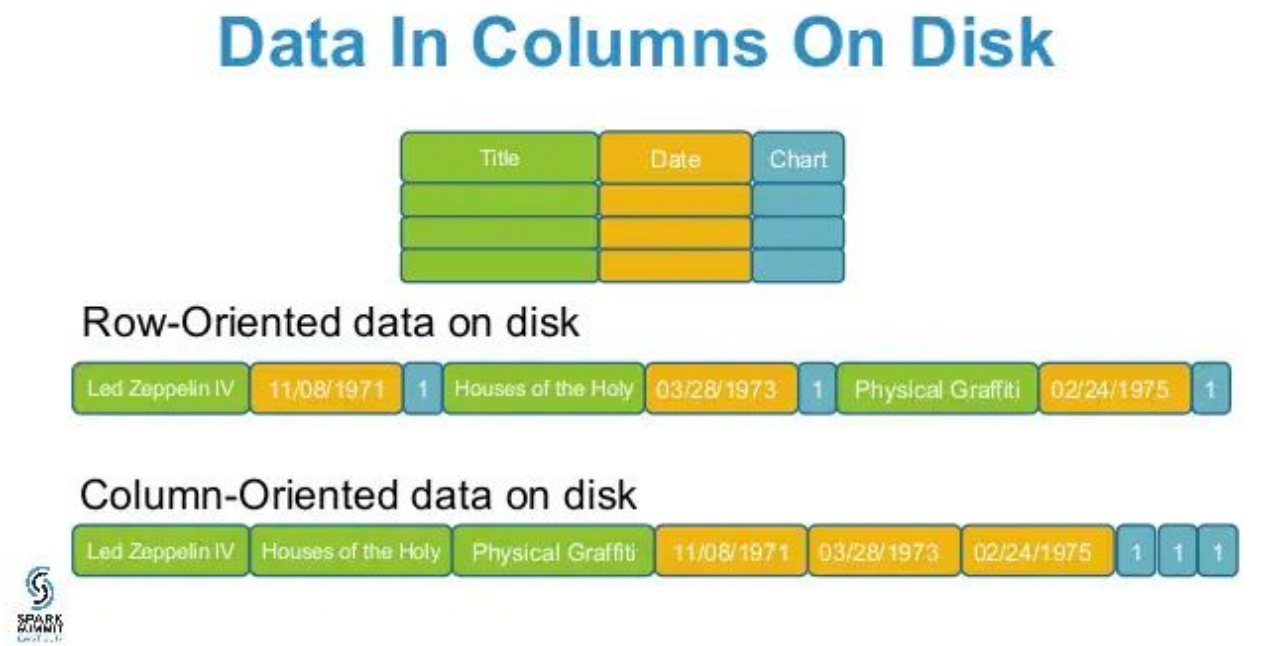


Stack

Apache Parquet

Stack

Apache Parquet



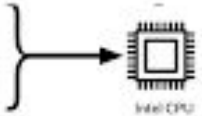
Stack

Apache Parquet

	day	location	product	sale
row 1	2017-01-01	l1	p1	300
row 2	2017-01-01	l1	p2	40
row 3	2017-01-01	l2	p1	44
row 4	2017-02-01	l1	p1	200

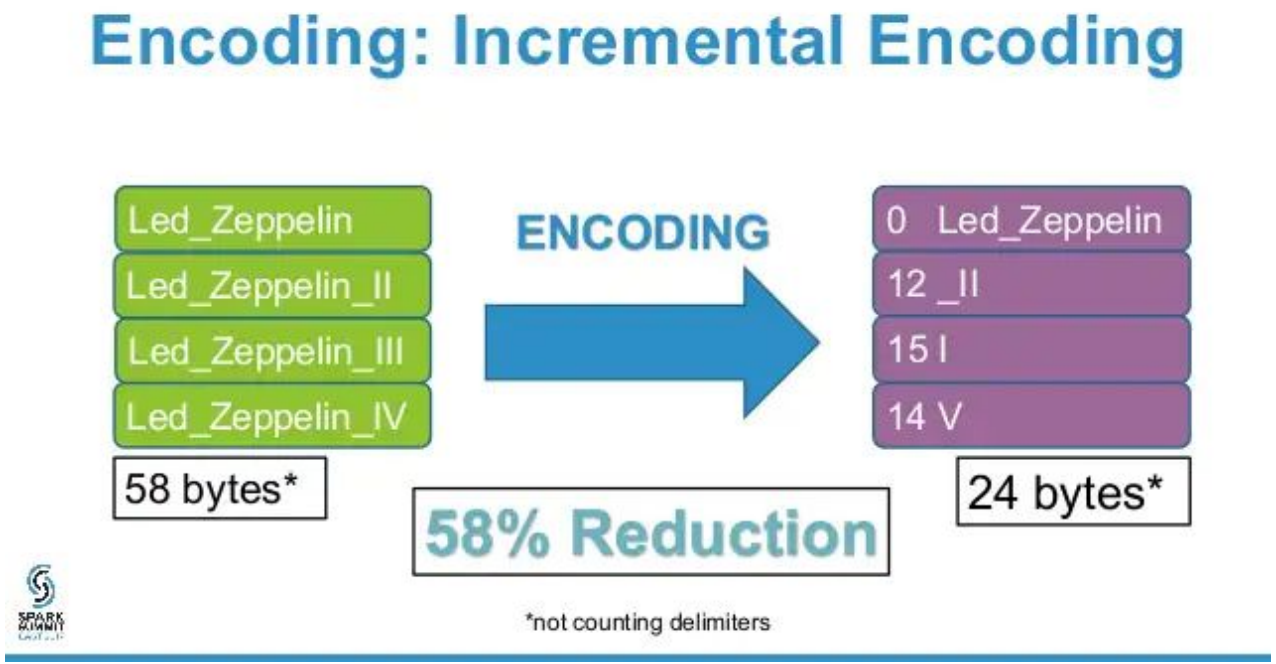
Traditional Memory Buffer	
row 1	2017-01-01
	l1
	p1
	300
row 2	2017-01-01
	l1
	p2
	40
row 3	2017-01-01
	l2
	p1
	44

Columnar Storage	
day	2017-01-01
	2017-01-01
	2017-01-01
	2017-01-02
location	l1
	l1
	l2
	l1
product	p1
	p2
	p1
	p1



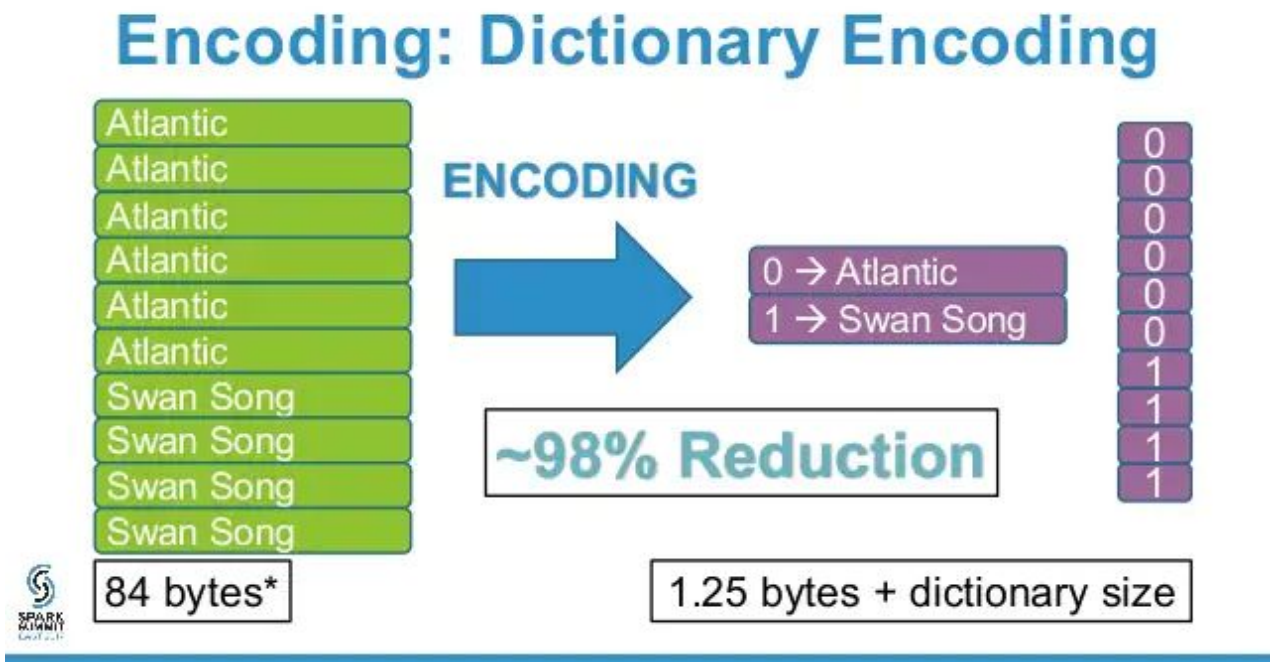
Stack

Apache Parquet



Stack

Apache Parquet



Stack

Apache Parquet

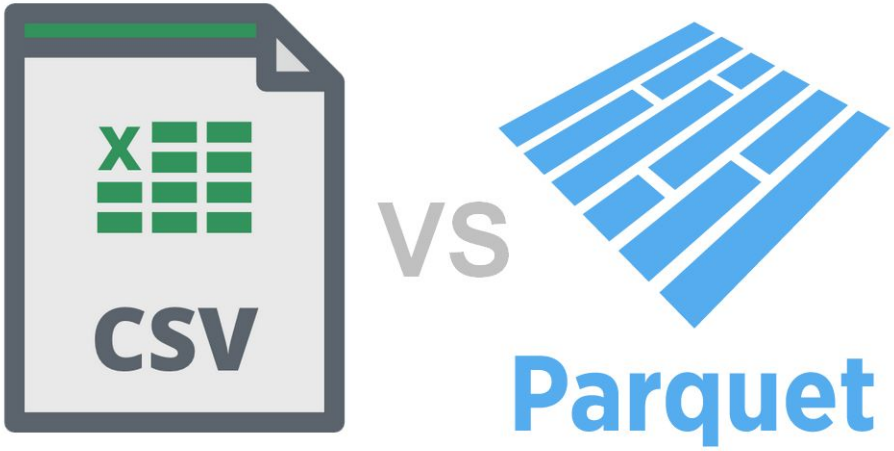
Partitioning

```
dataFrame
    .write
    .partitionBy("Whatever", "Columns", "You", "Want")
    .parquet(outputFile)

// For a common example
dataFrame
    .write
    .partitionBy("Year", "Month", "Day", "Hour")
    .parquet(outputFile)
```

Stack

Apache Parquet



- Não existe estatísticas dos dados no arquivo.
- É preciso ler todo o arquivo para definição dos data types.
- Não permite compressão por coluna.
- Ocupa mais espaço e mais processamento.

Stack

Apache Parquet

The following table compares the savings as well as the speedup obtained by converting data into Parquet from CSV.

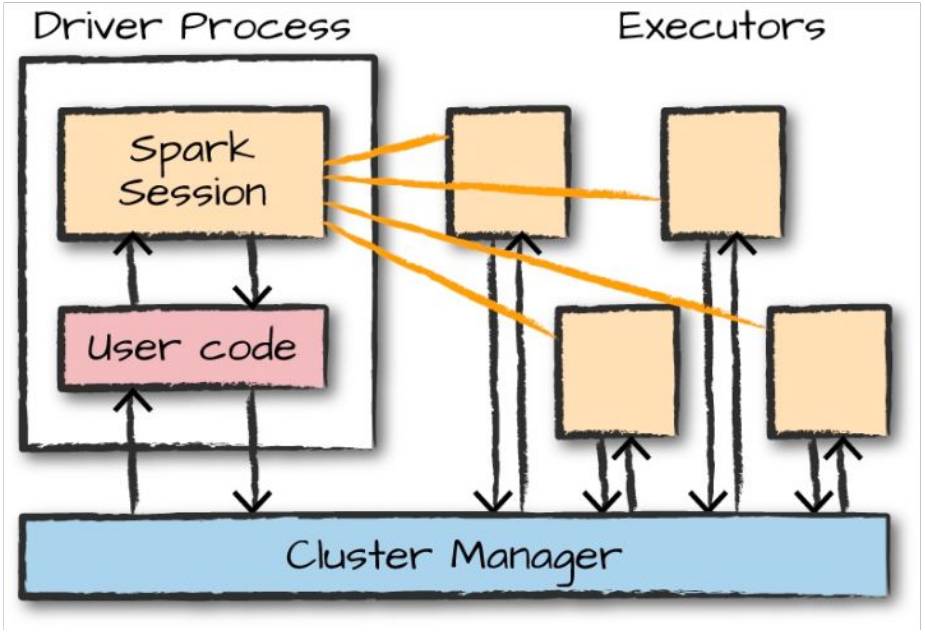
Dataset	Size on Amazon S3	Query Run Time	Data Scanned	Cost
Data stored as CSV files	1 TB	236 seconds	1.15 TB	$5.75
Data stored in Apache Parquet Format	130 GB	6.78 seconds	2.51 GB	$0.01
Savings	87% less when using Parquet	34x faster	99% less data scanned	99.7% savings

Stack

Hands on!

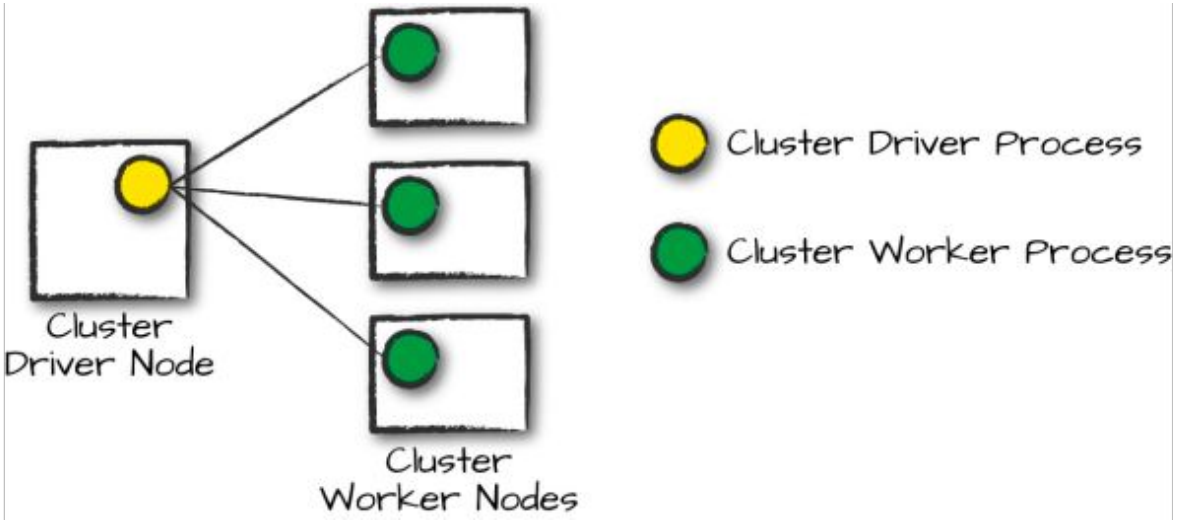
Arquitetura Spark

- Master: Sessões, Planos de Execução
- Slaves: Executores, máquinas escravas para processamento de jobs.



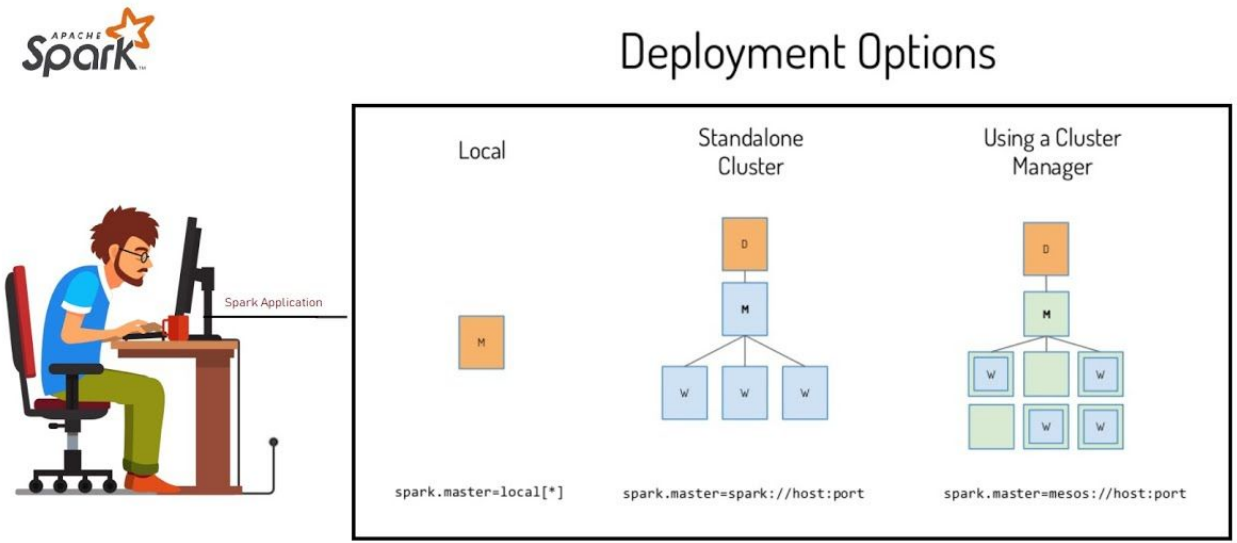
Stack

Cluster Communication



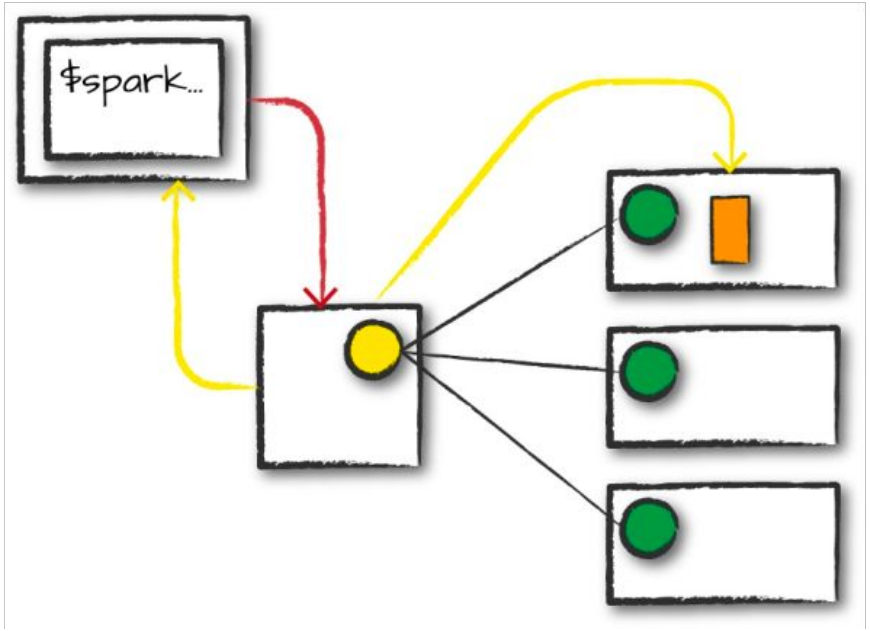
Stack

Deployment modes



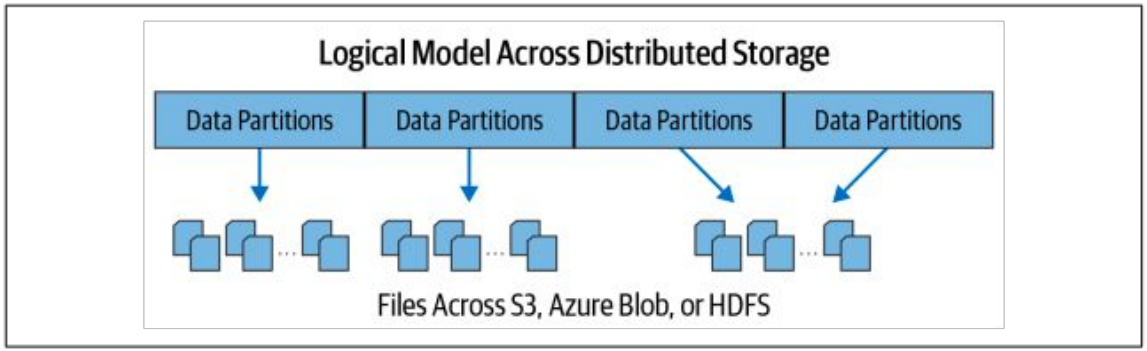
Stack

Executando uma aplicação



Stack

Leitura de dados

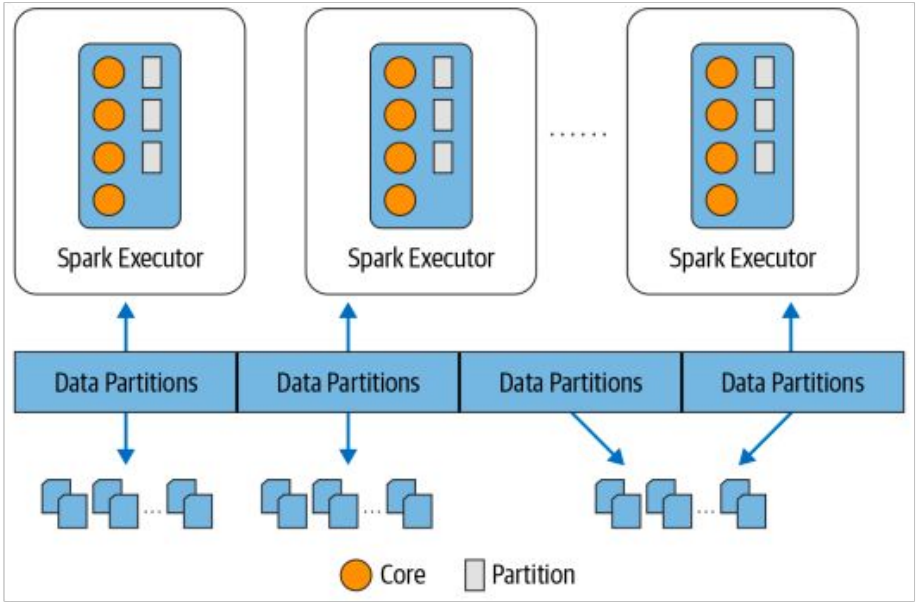


Storage

Database (jdbc)

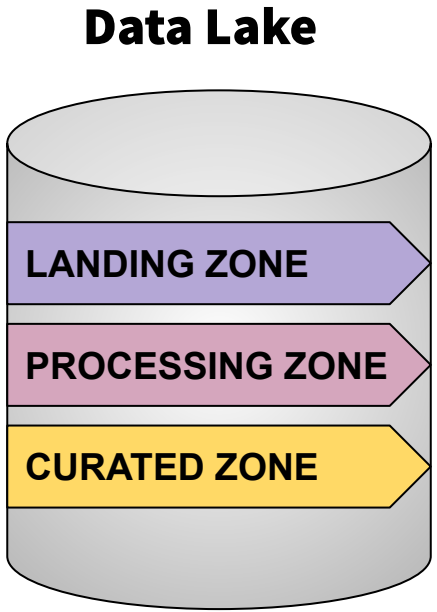
Stack

Leitura e processamento de dados



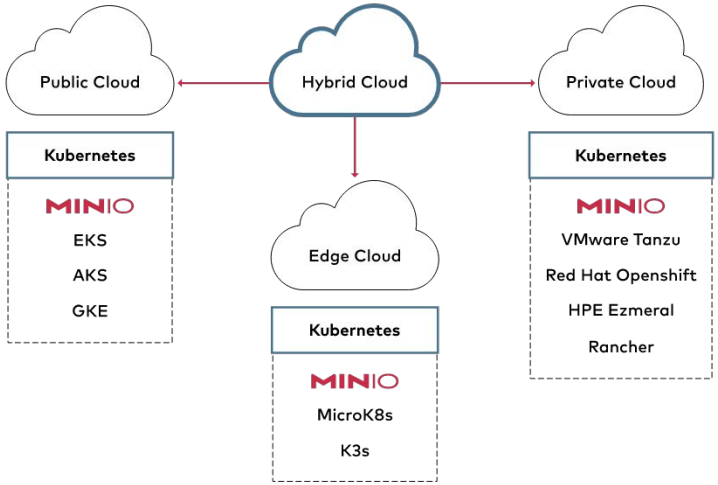
Stack

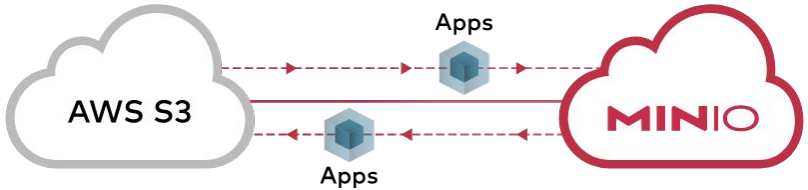
Criando um ambiente local



Stack

Criando um ambiente local





Stack

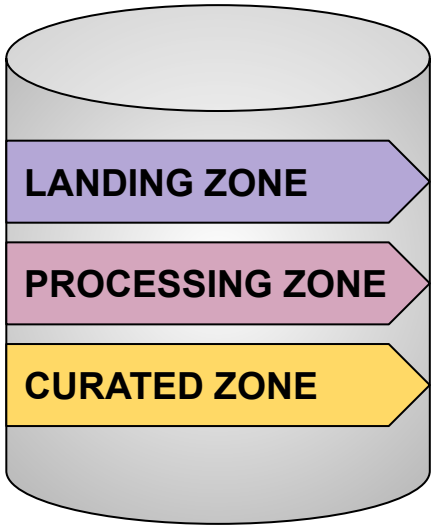
Hands on!

Criando um ambiente em Cloud

amazon S3

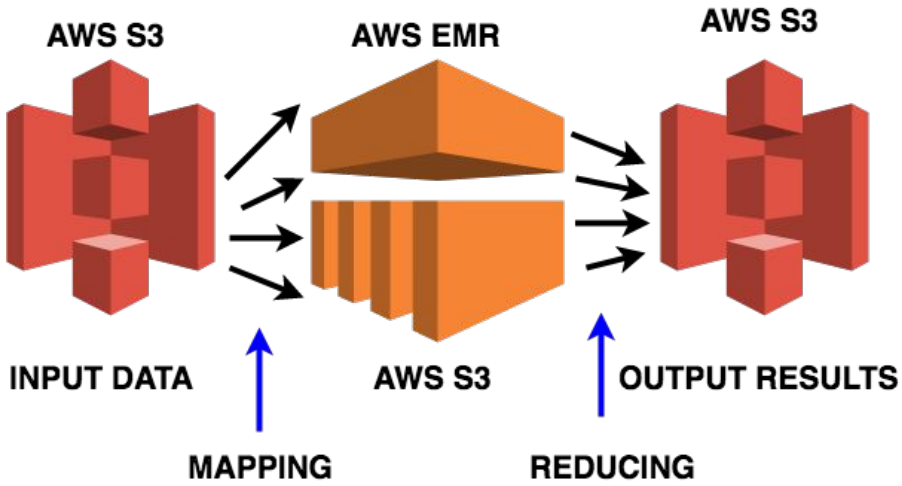
APACHE Spark™

Data Lake



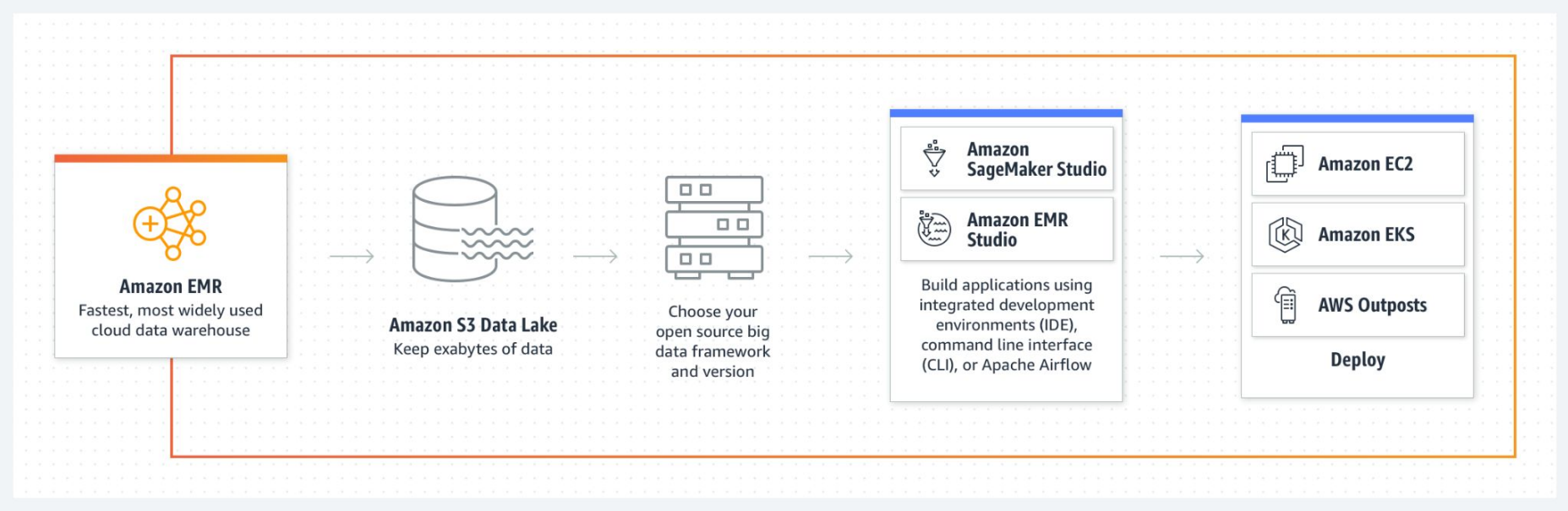
Stack

Criando um ambiente em Cloud



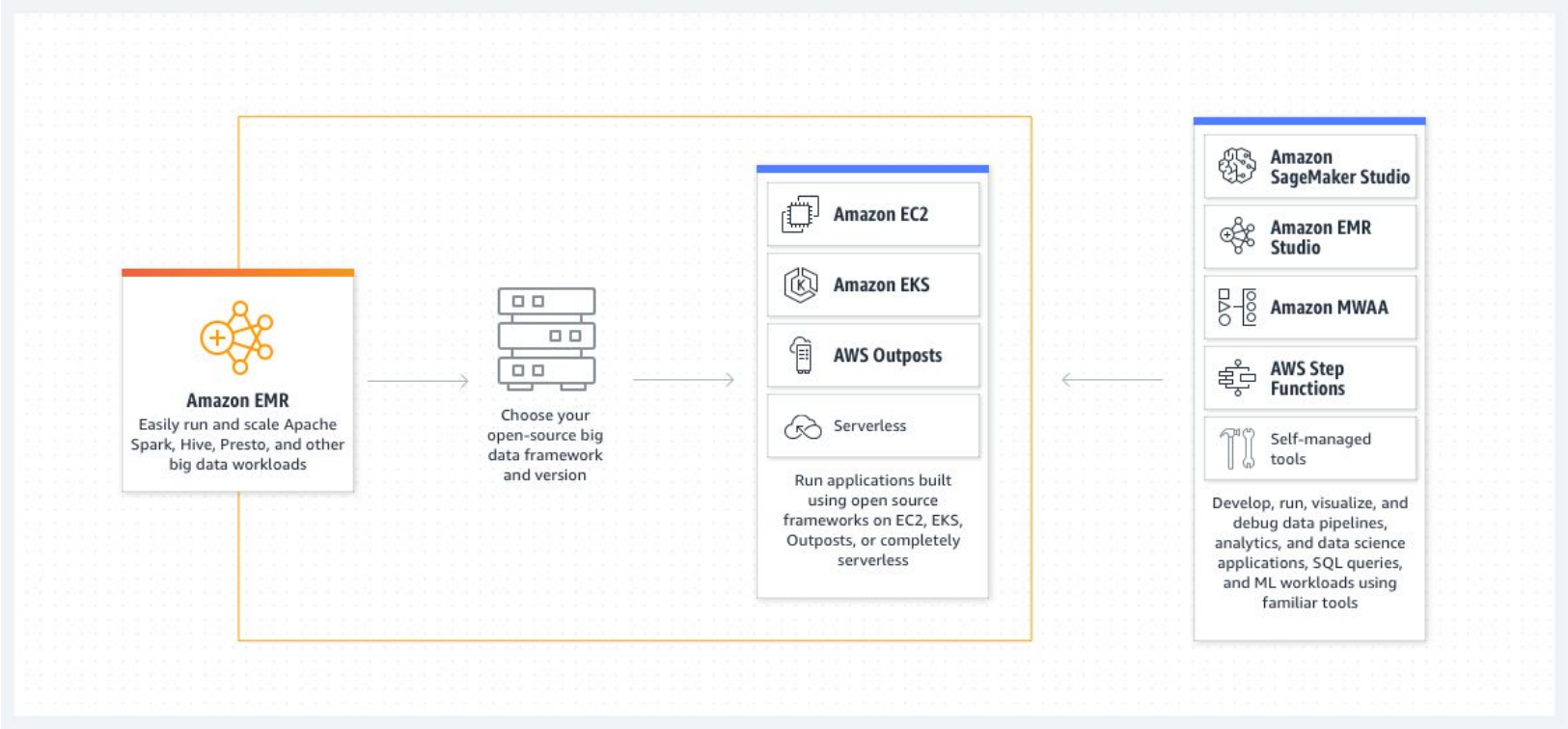
Stack

Amazon EMR



Stack

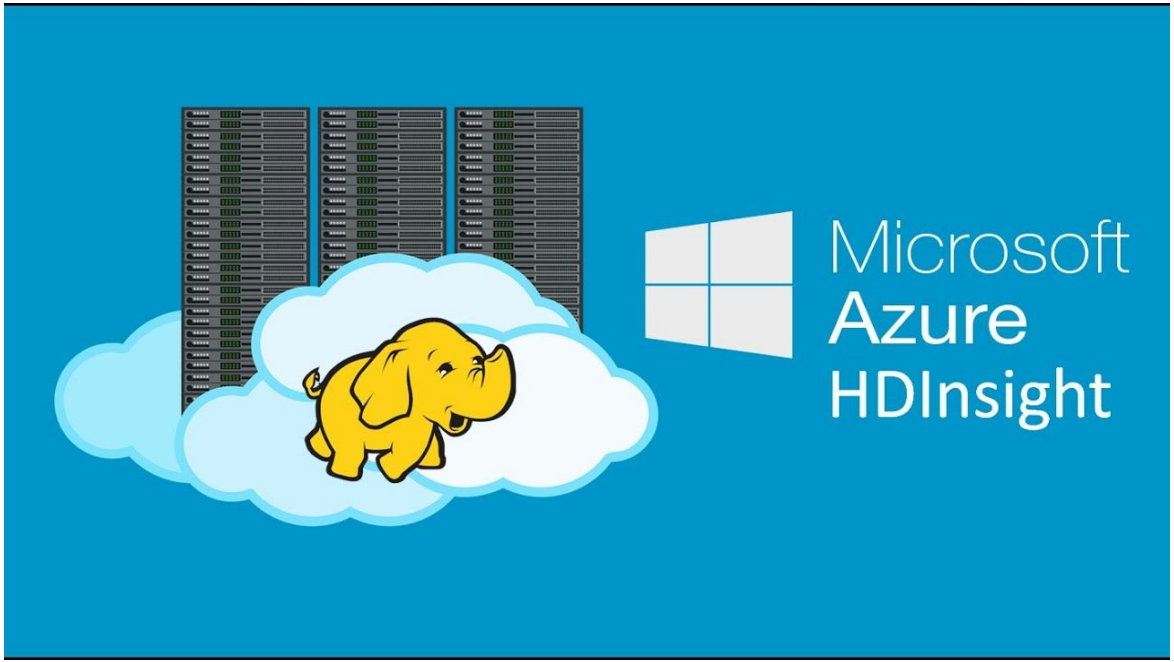
Amazon EMR



Stack

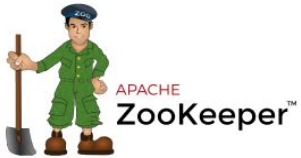
Hands on!

HDinsight



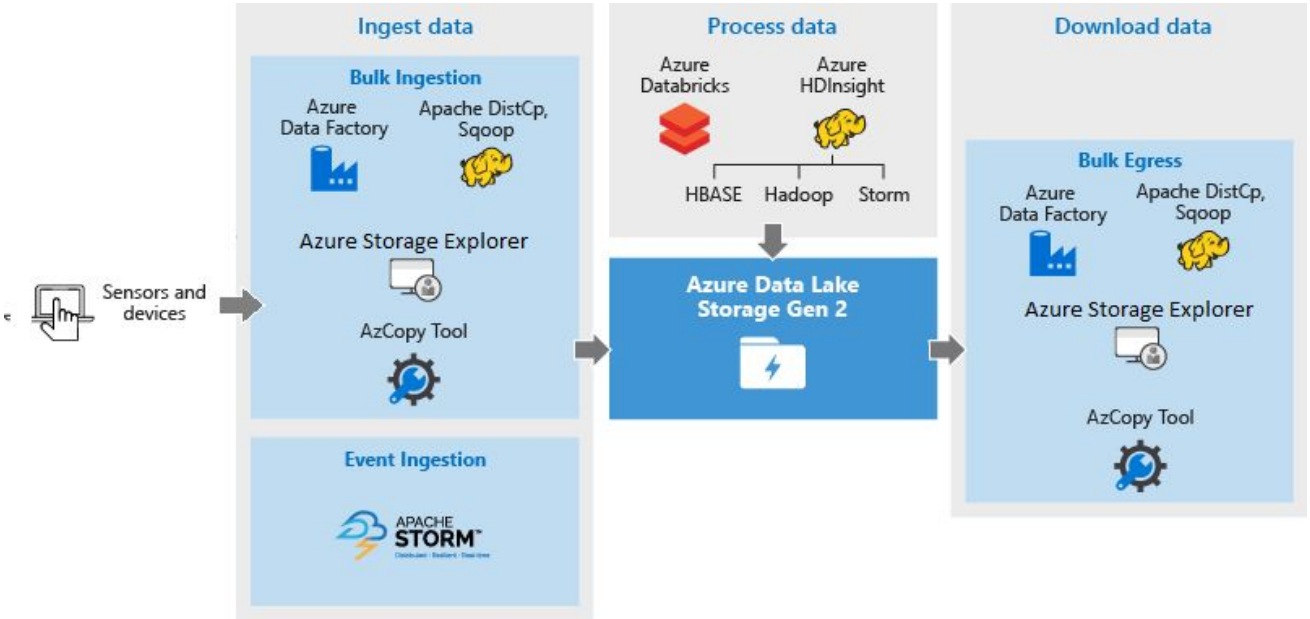
Stack

HDinsight



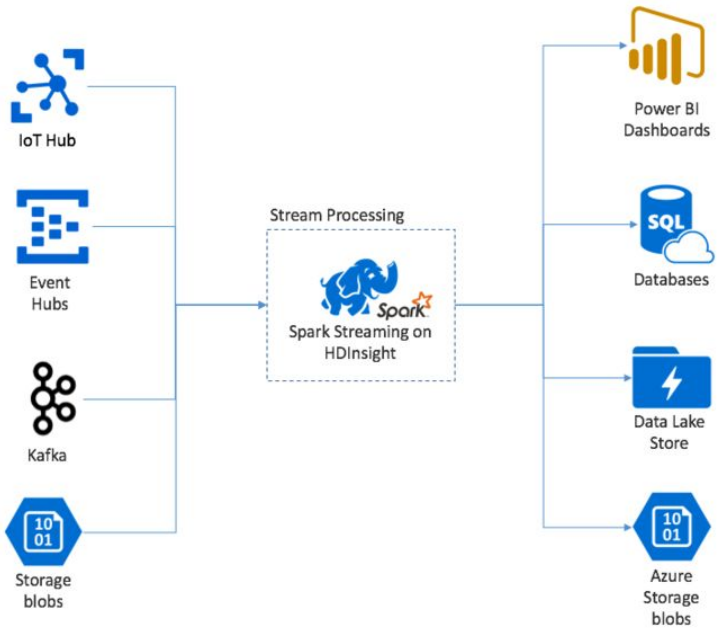
Stack

Data Lake



Stack

HDinsight



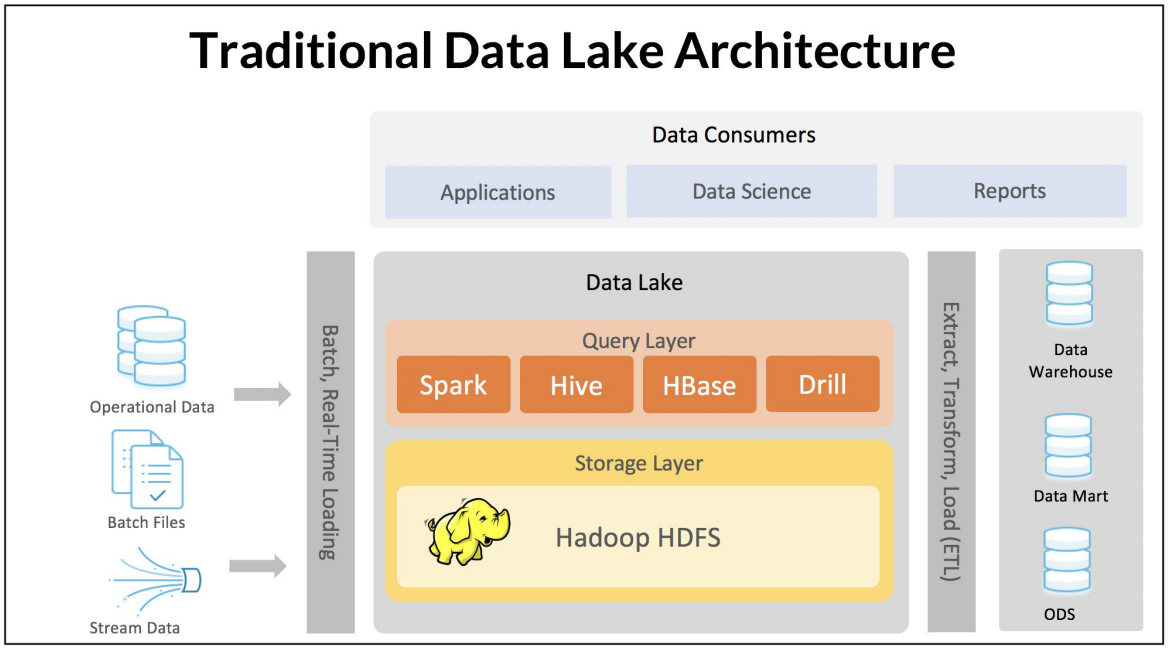
Stack

Hands on!

Hadoop

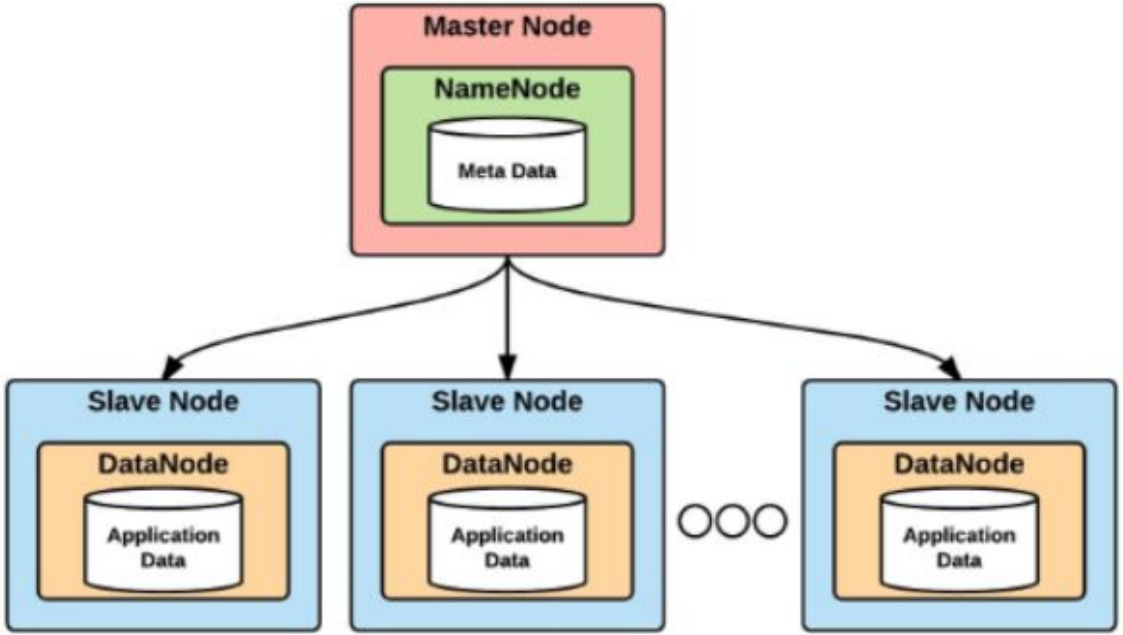
Stack

Hadoop



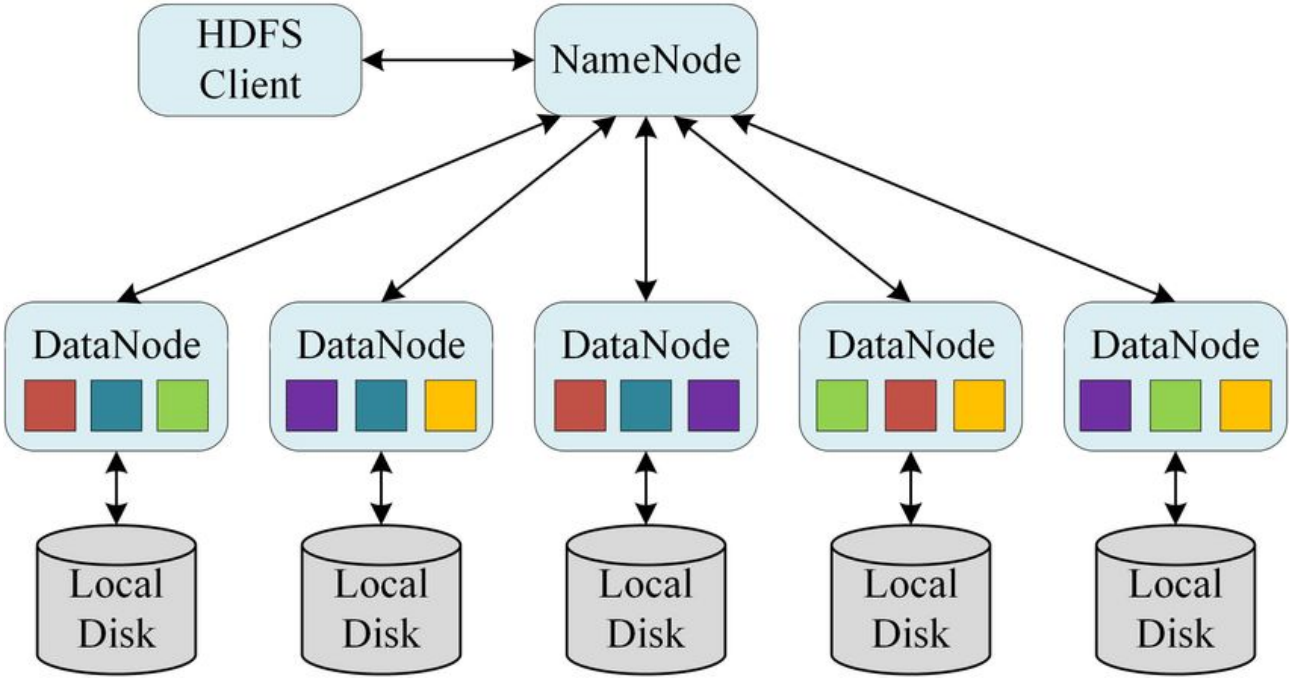
Stack

Hadoop



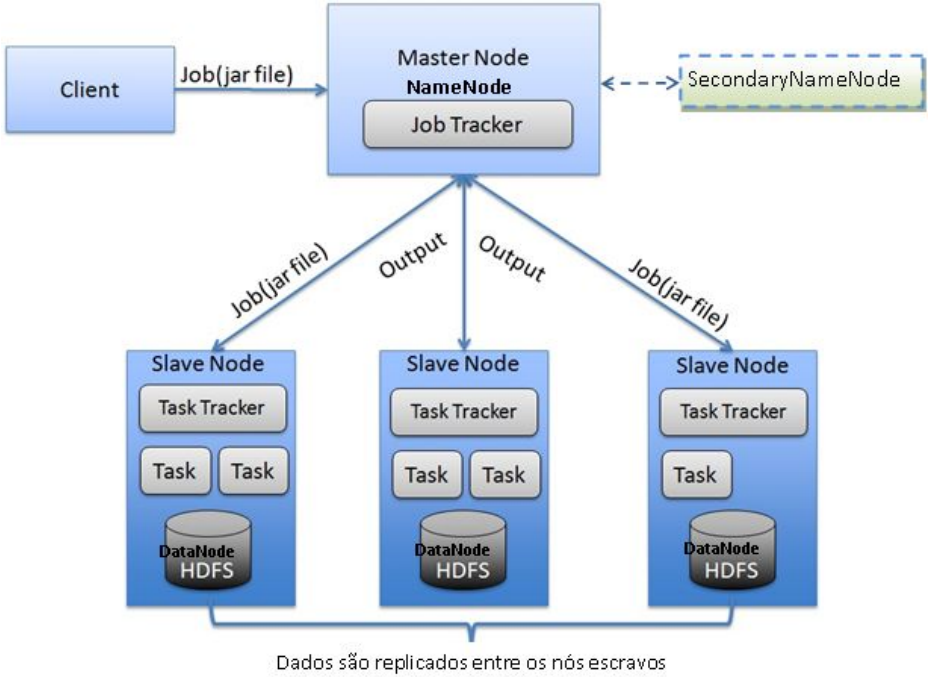
Stack

Hadoop



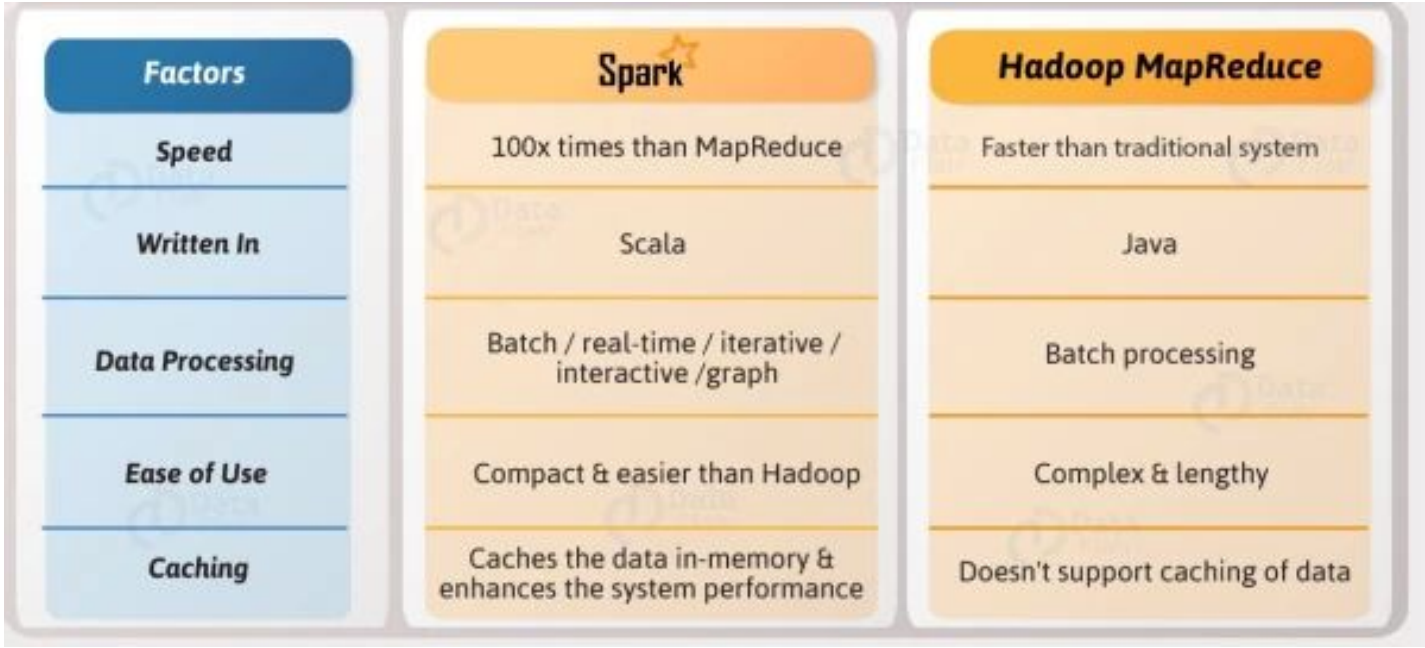
Stack

Hadoop



Stack

Hadoop



Factors	Spark	Hadoop MapReduce
Speed	100x times than MapReduce	Faster than traditional system
Written In	Scala	Java
Data Processing	Batch / real-time / iterative / interactive /graph	Batch processing
Ease of Use	Compact & easier than Hadoop	Complex & lengthy
Caching	Caches the data in-memory & enhances the system performance	Doesn't support caching of data

Stack

Hands on!

Hive

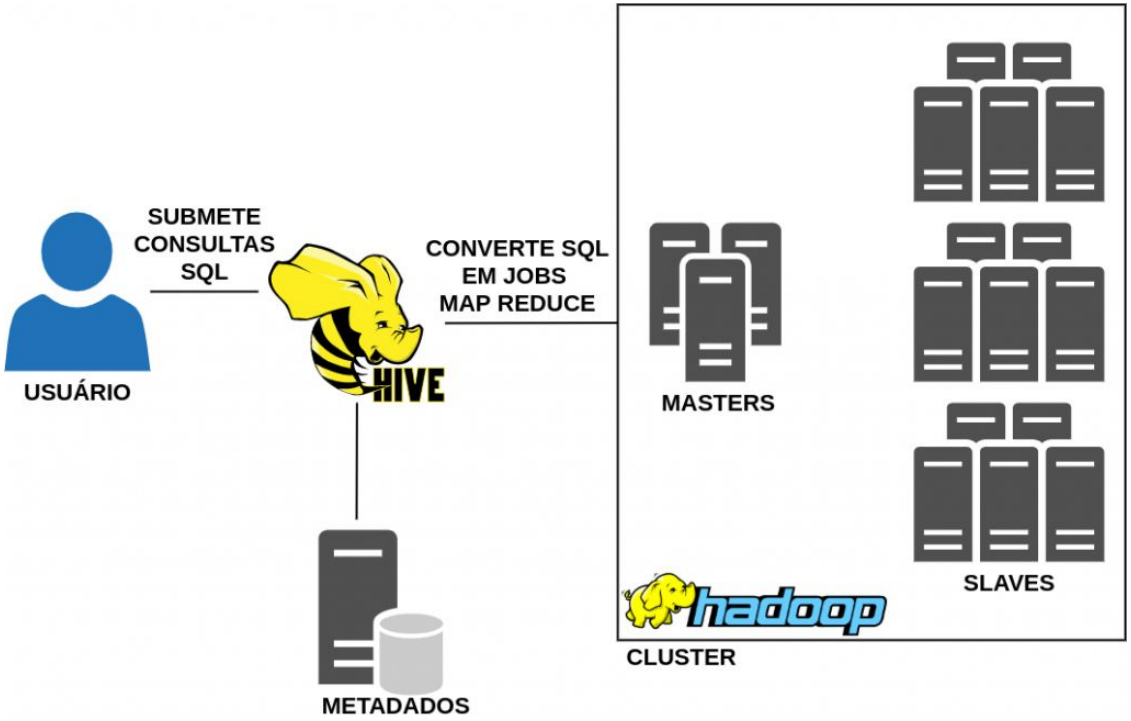
Stack

Hive


Challenge...
>950 Million Users
>500 Terabytes per day
>70k queries per day
>300 million photos per day
Traditional RDBMS...
Solution...
hadoop MapReduce
Hard to program
Users Know SQL well
HIVE
Tables can be partitioned & bucketed
Schema flexibility & evolution
Easy to plug-in custom mapper/ reducer code
JDBC/ODBC drivers are available
HIVE tables can be befined directly on HDFS
Extensible: Types, Formats, Functions, Scripts

Stack

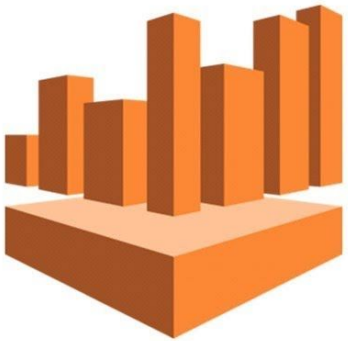
Hive


SUBMETE
CONSULTAS
SQL
USUÁRIO
HIVE
CONVERTE SQL
EM JOBS
MAP REDUCE
METADADOS
MASTERS
SLAVES
hadoop
CLUSTER

Stack

AWS Athena

Overview of AWS Athena



Amazon Athena

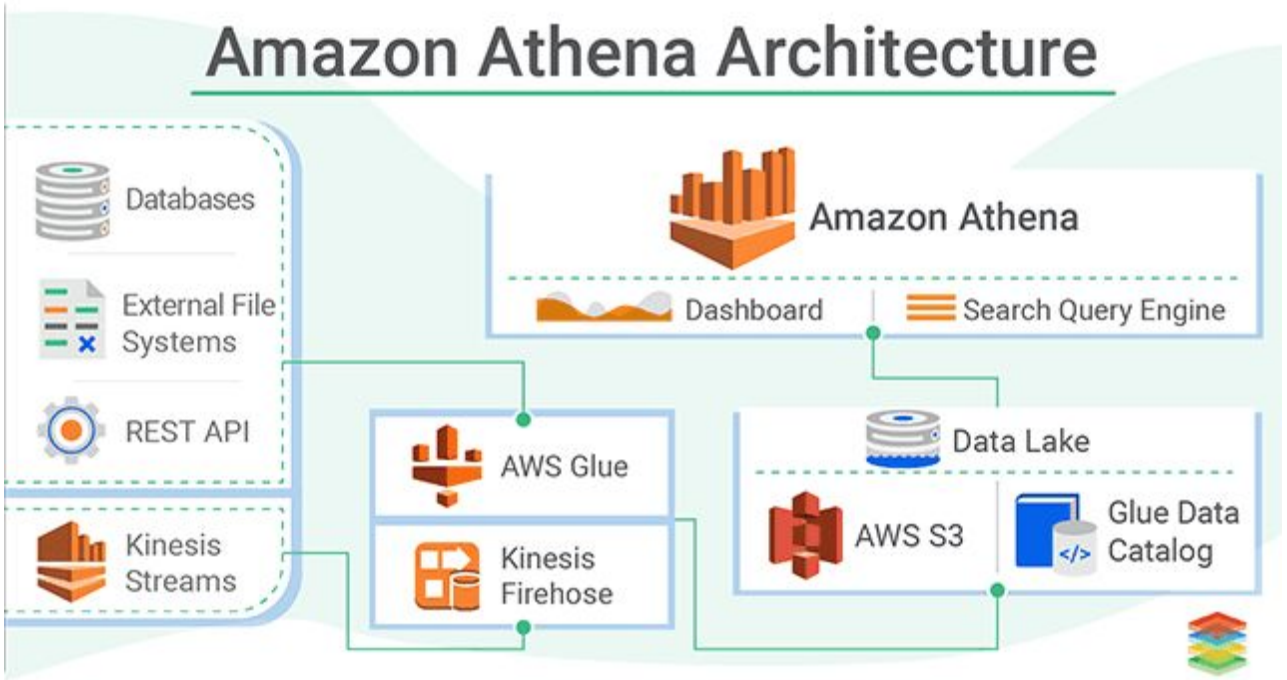
Serverless
Interactive
Query

Stack

AWS Athena

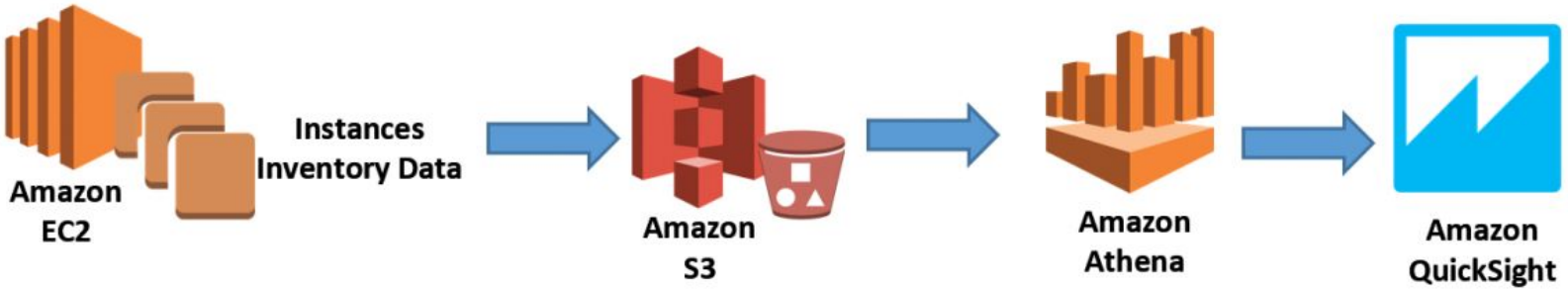
Stack

Arquitetura



Stack

Arquitetura



Stack

AWS Athena

Athena

- Menos utilizado para processamento de dados em larga escala.

- Mais utilizado como uma ferramenta para executar queries ad-hoc.

- Serverless - não precisa gerenciar o cluster.

EMR

- É muito mais do que execuções de queries.

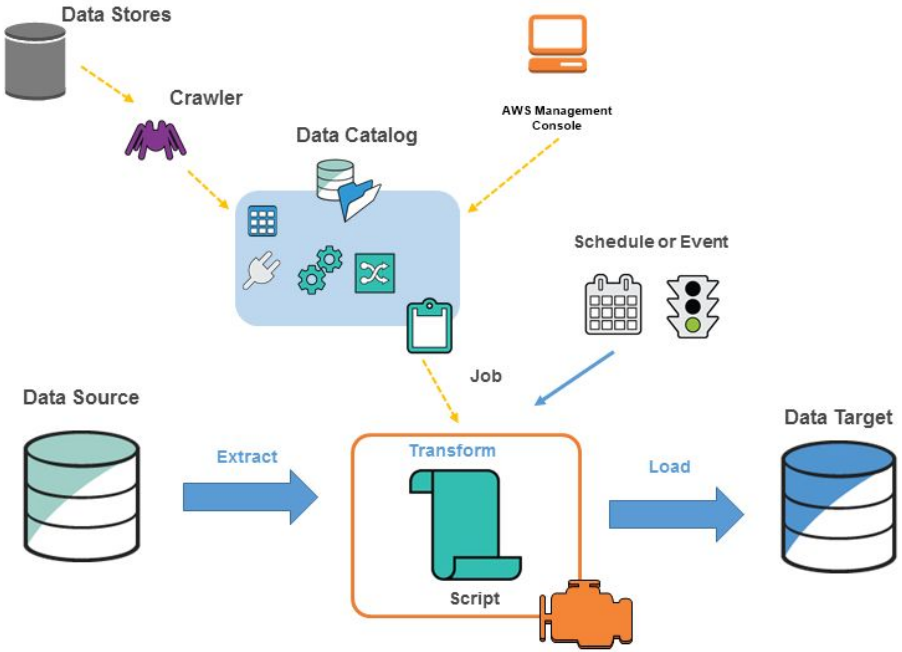
- Utiliza scripts/códigos customizados para processar e analisar grande volume de dados.

- Necessário gerenciar o cluster do EMR.

Stack

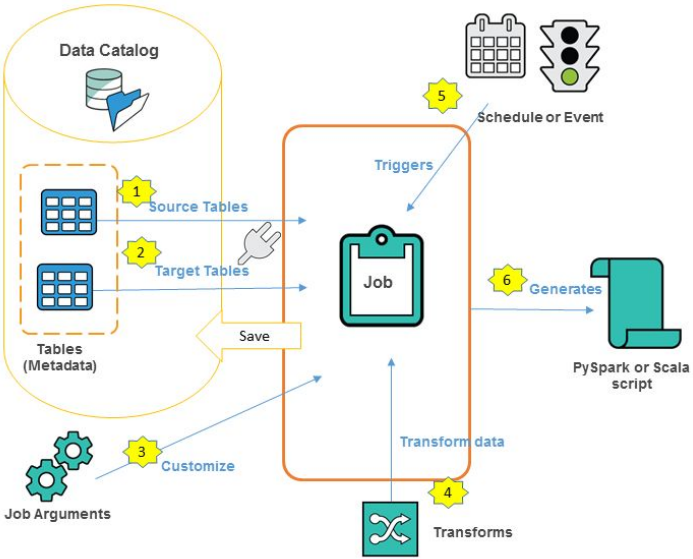
Hands on!

AWS Glue + Athena

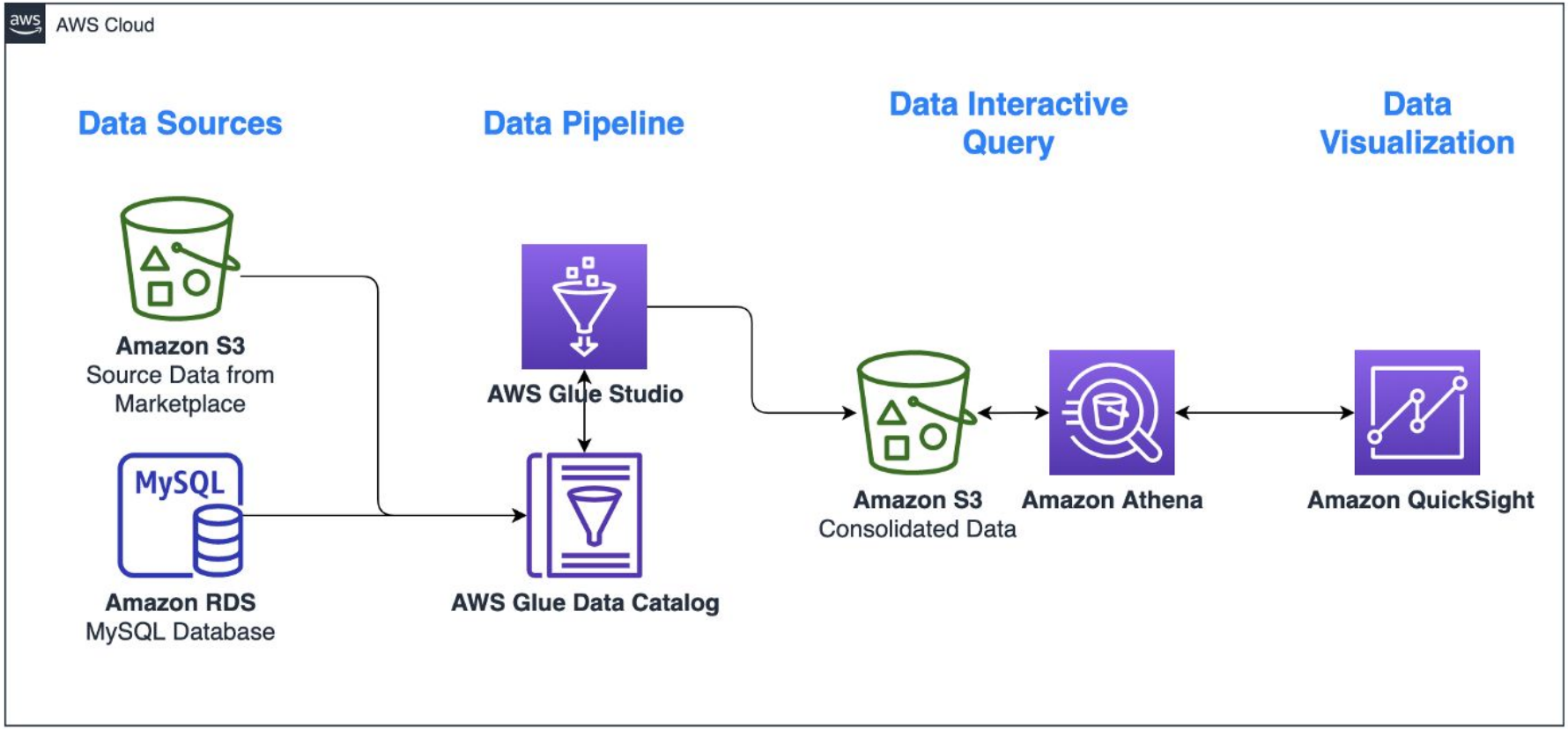


Stack

AWS Glue Studio



Stack

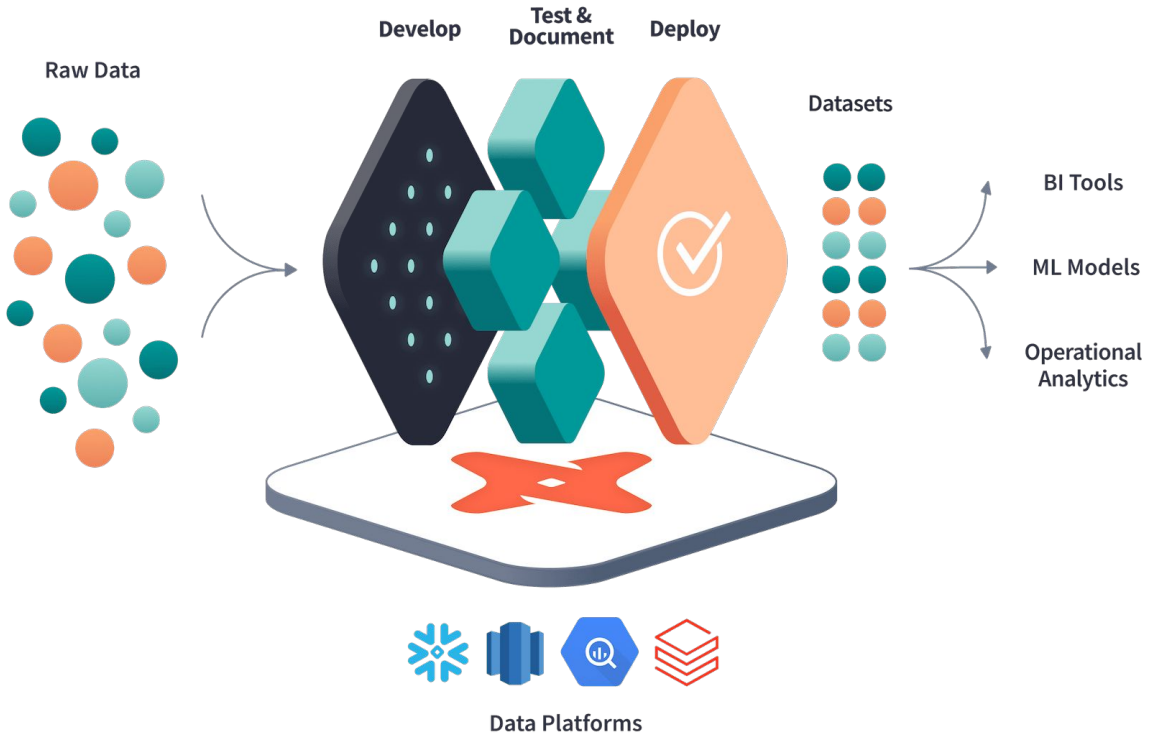

AWS Cloud
Data Sources
Data Pipeline
Data Interactive Query
Data Visualization
Amazon S3
Source Data from Marketplace
MySQL
Amazon RDS
MySQL Database
AWS Glue Studio
AWS Glue Data Catalog
Amazon S3
Consolidated Data
Amazon Athena
Amazon QuickSight

Stack

DBT - Data Build Tool

Stack

DBT - Data Build Tool



Stack

Data build tool

Version Control and CI/CD

Deploy safely using dev environments. Git-enabled version control enables collaboration and a return to previous states.

Test and Document

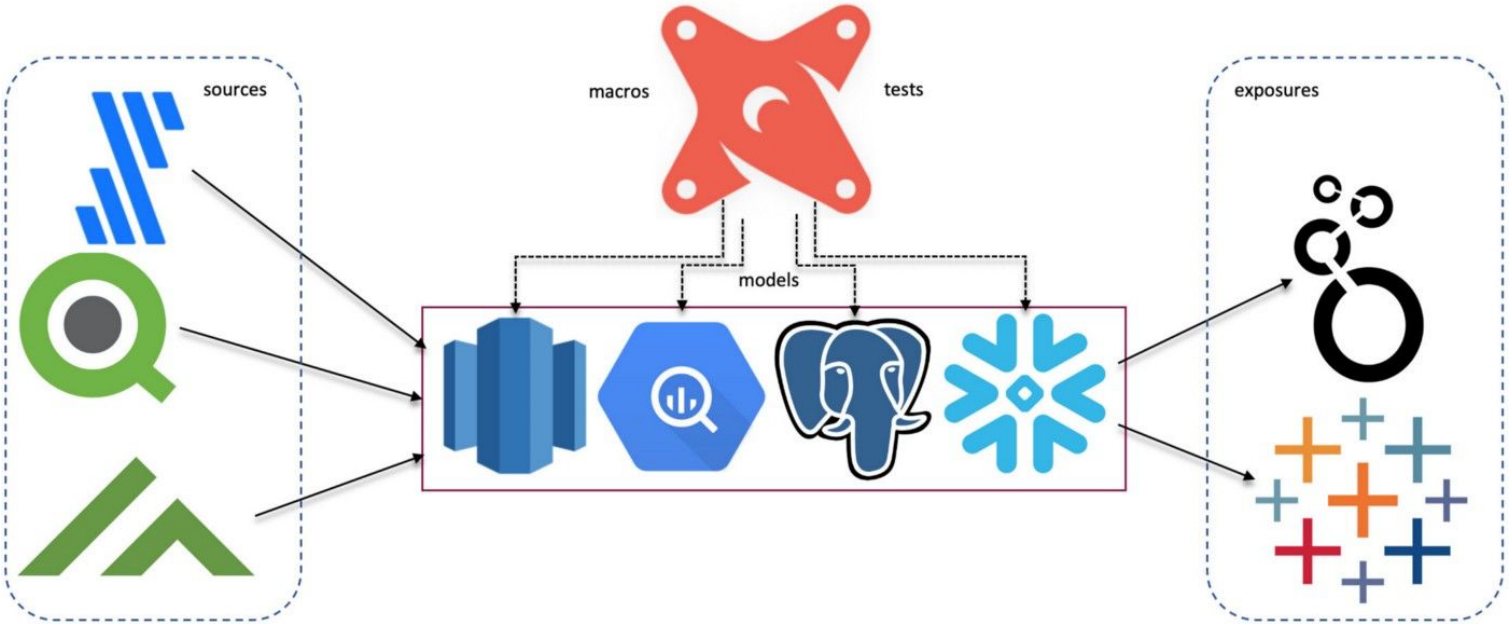
Test every model prior to production, and share dynamically generated documentation with all data stakeholders.

Develop

Write modular SQL models with SELECT statements and the ref() function– dbt handles the chore of dependency management.

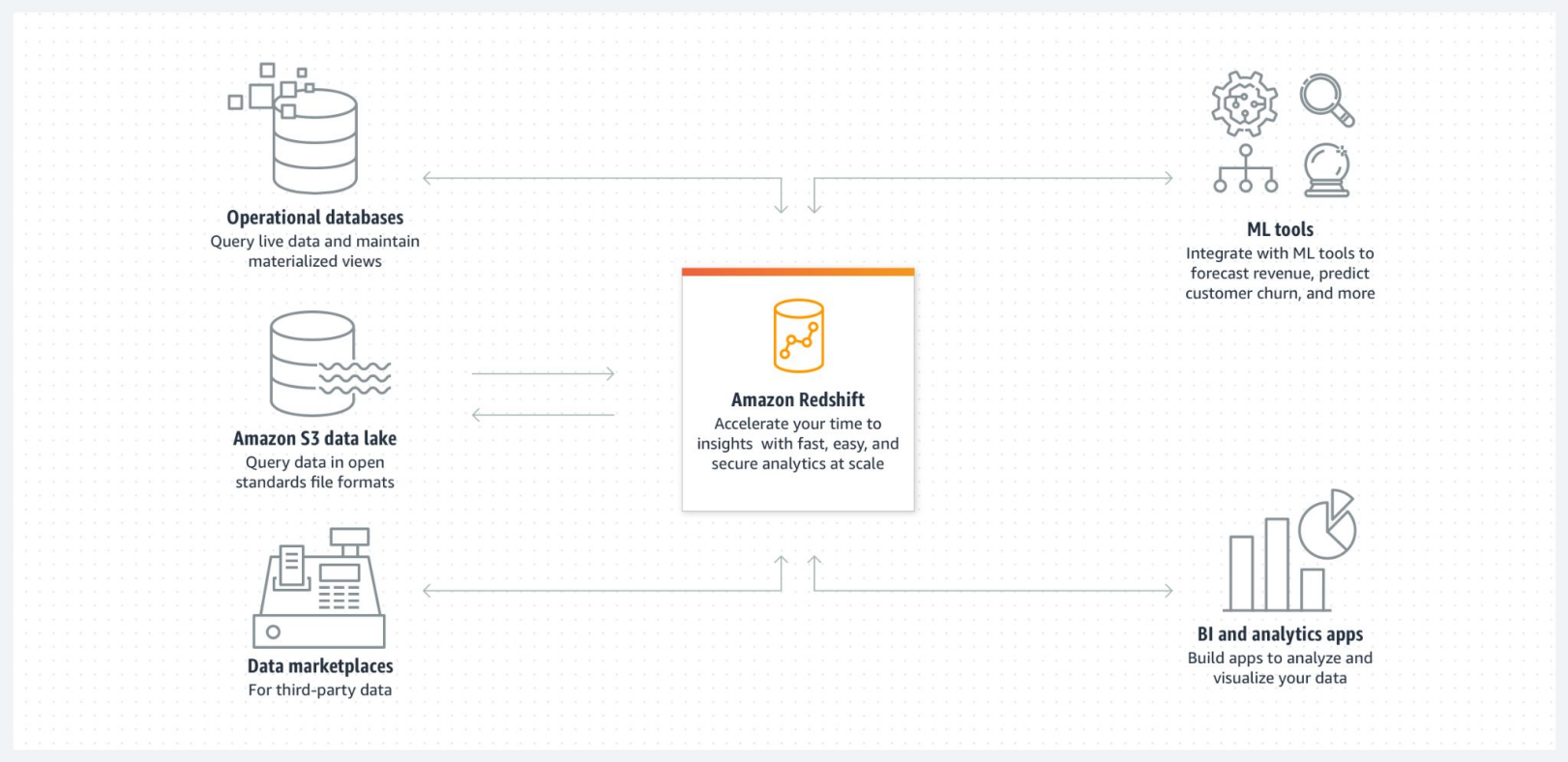
Stack

Data build tool



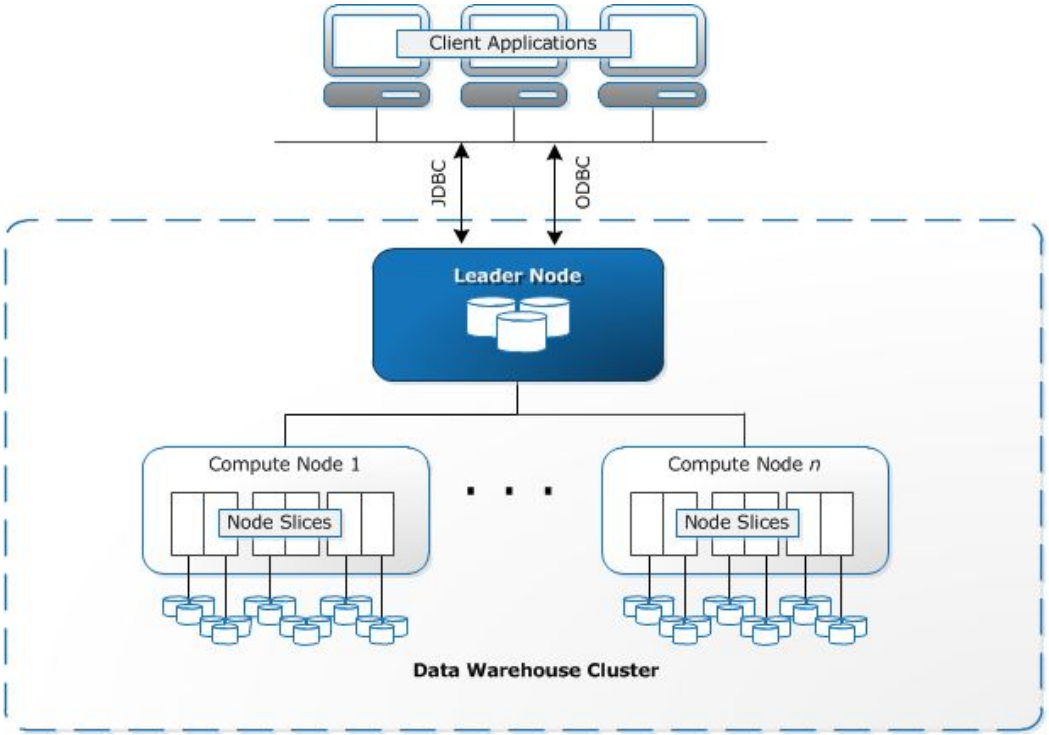
Stack

Redshift



Stack

Redshift



Stack

Redshift

client_id	client_name	client_email	client_state
1	ABC	a@abc.com	MH
	DEF	d@def.com	RJ
3	GHI	g@ghi.com	MP

row below

1|ABC|a@abc.com|MH	2|DEF|d@def.com|RJ	3|GHI|g@ghi.com|MP
Block 1	Block 2	Block 3

Columnar Storage

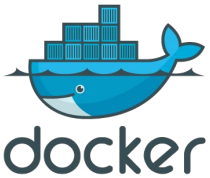
1|2|3|4|5|6|7|8|9|10	11|12|13|14|15|16|17|18|19	20|21|22|23|24|25|26|27
Block 1	Block 2	Block 3

Stack

Redshift

- Compression
- Massively Parallel Processing
- Machine Learning Optimization
- AWS SageMaker

Setup local

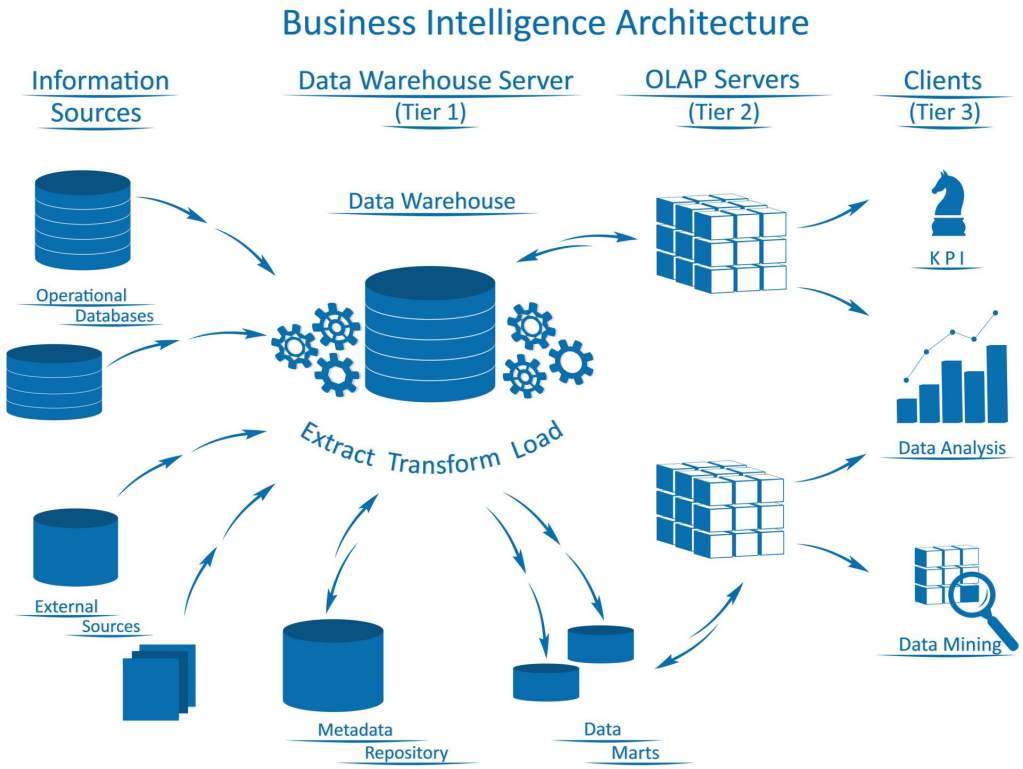


amazon
REDSHIFT

Stack

Hands on!

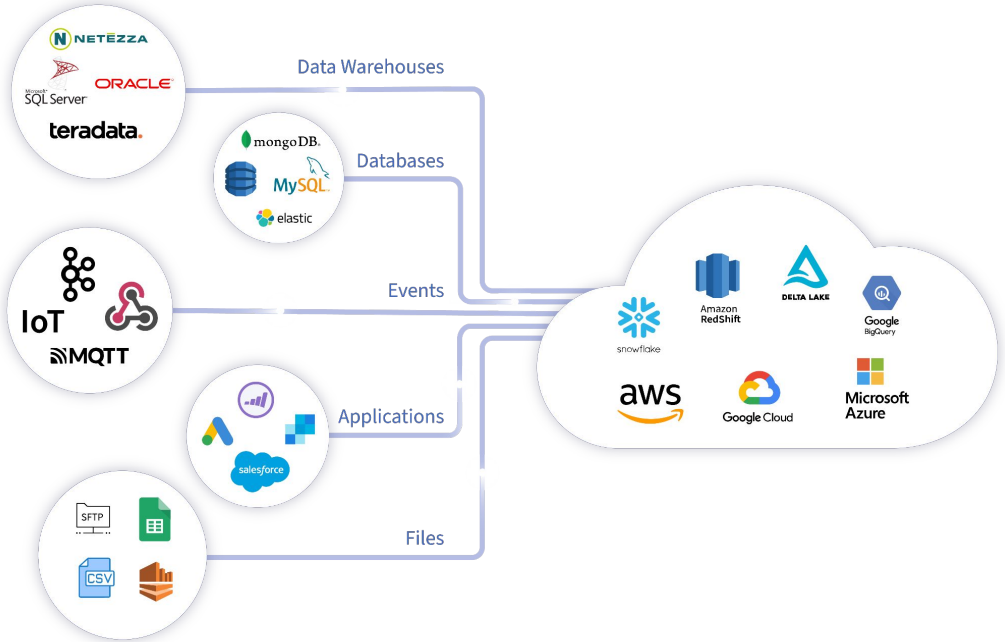
Ingestion process



Stack

Ingestion types

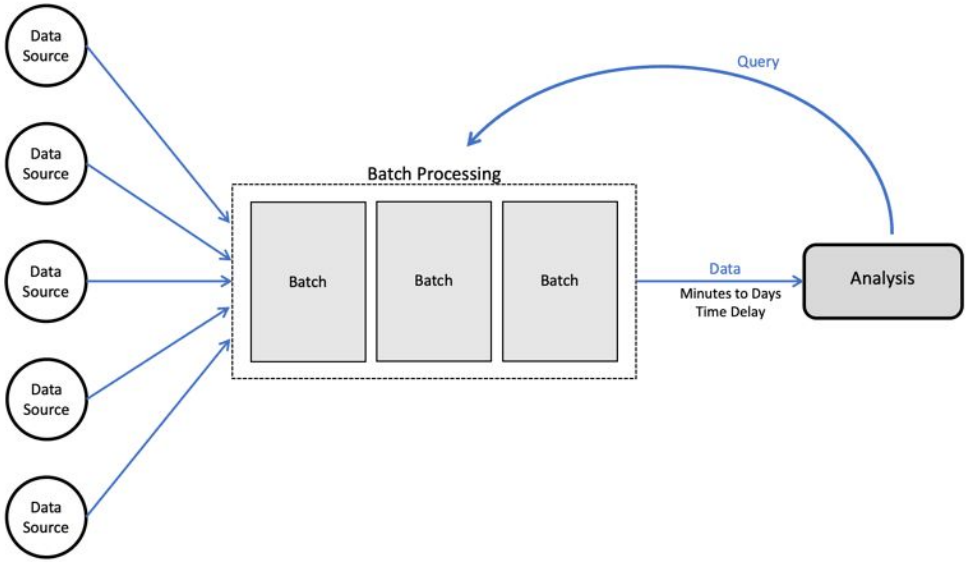
- Batch
- Real-time
- Lambda architecture



Stack

Ingestion types

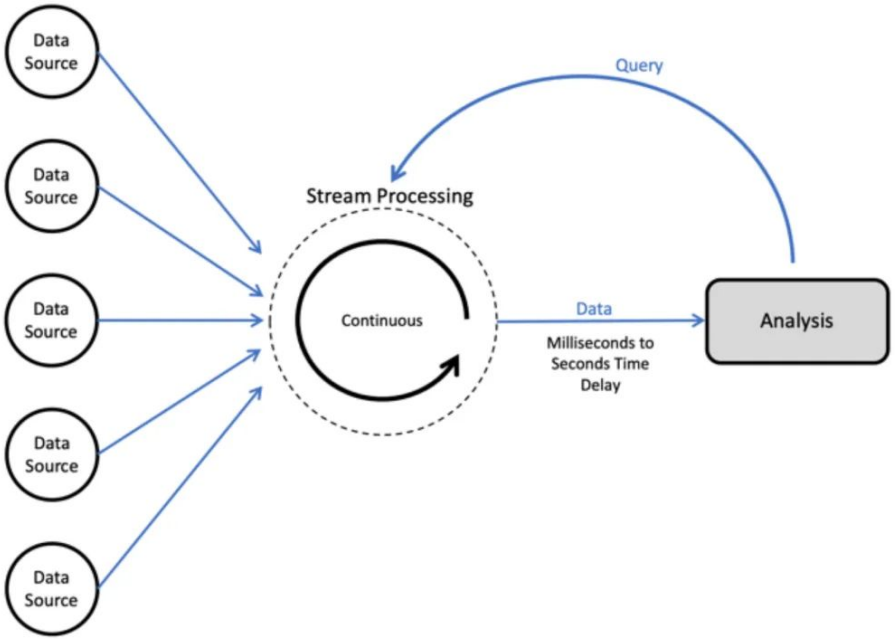
- Batch
- Real-time
- Lambda architecture



Stack

Ingestion types

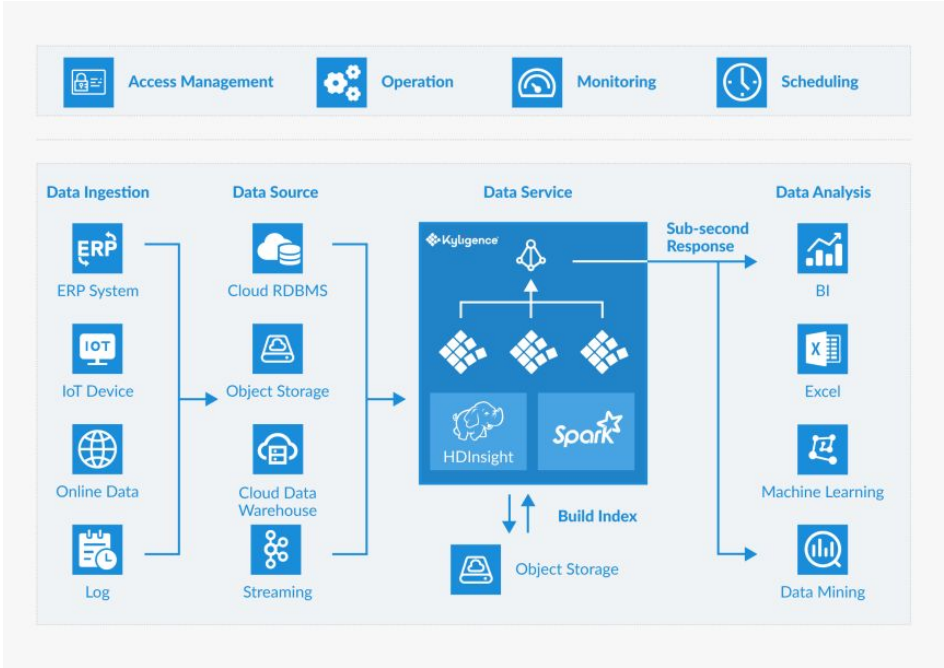
- Batch
- **Real-time**
- Lambda architecture



Stack

Benefits of Data Ingestion

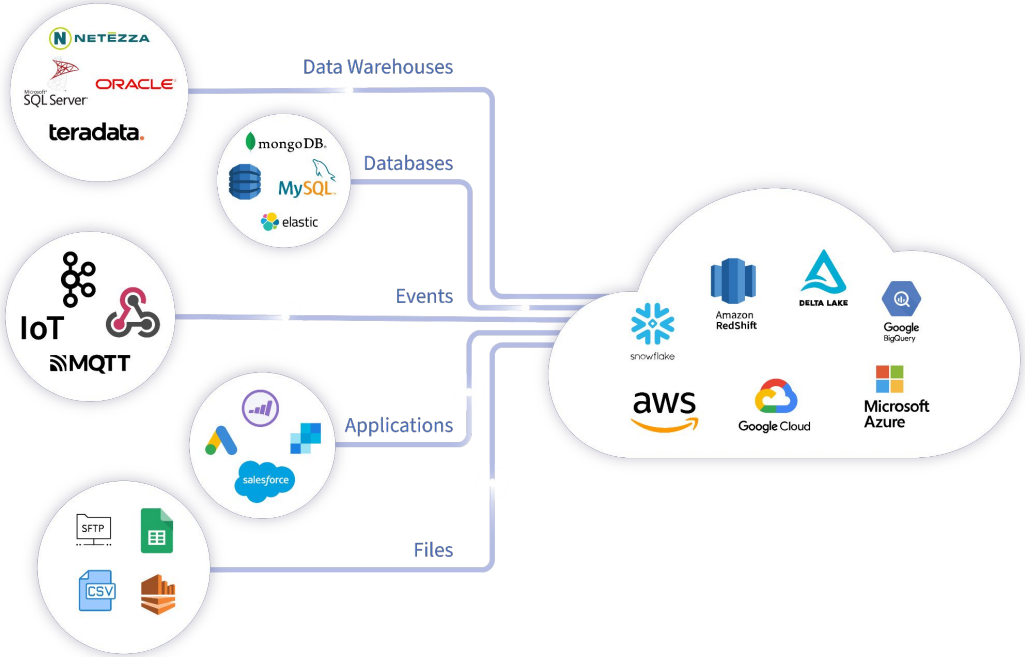
- Data is readily available
- Data is less complex
- Teams save time and money
- Companies make better decisions
- Teams create better apps and software tools



Stack

Considerations

- Data Velocity
- Data Size
- Data Frequency (Change Data Capture?)
- Data Format

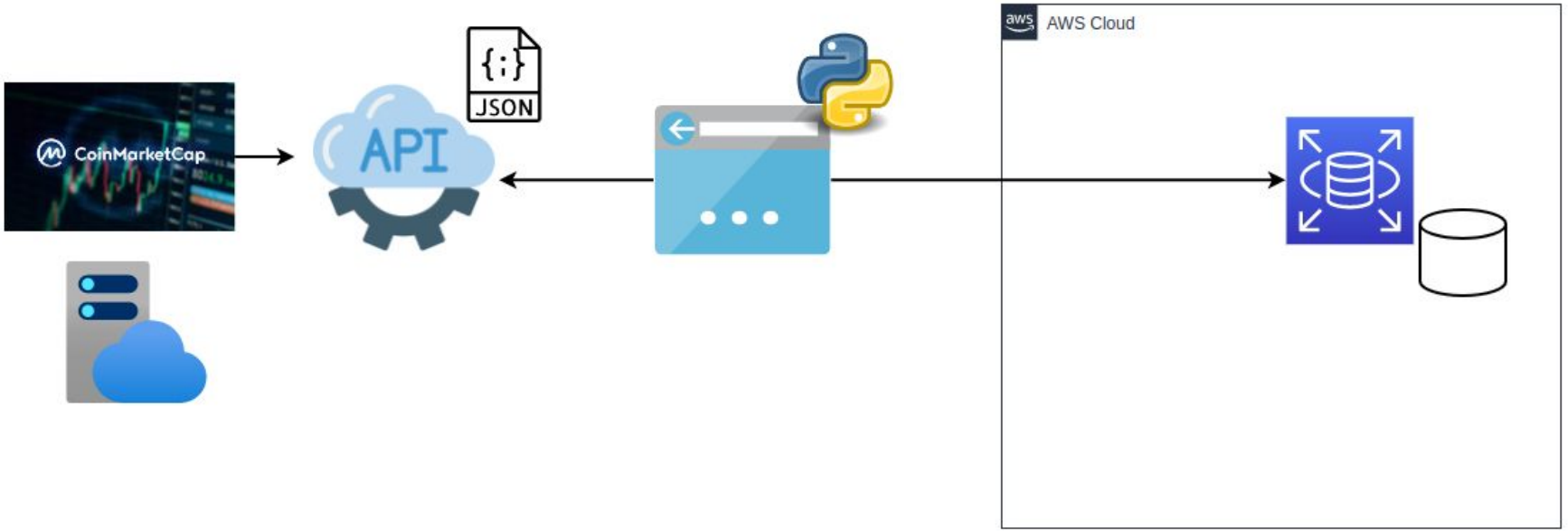


Stack

Data Pipeline

Stack

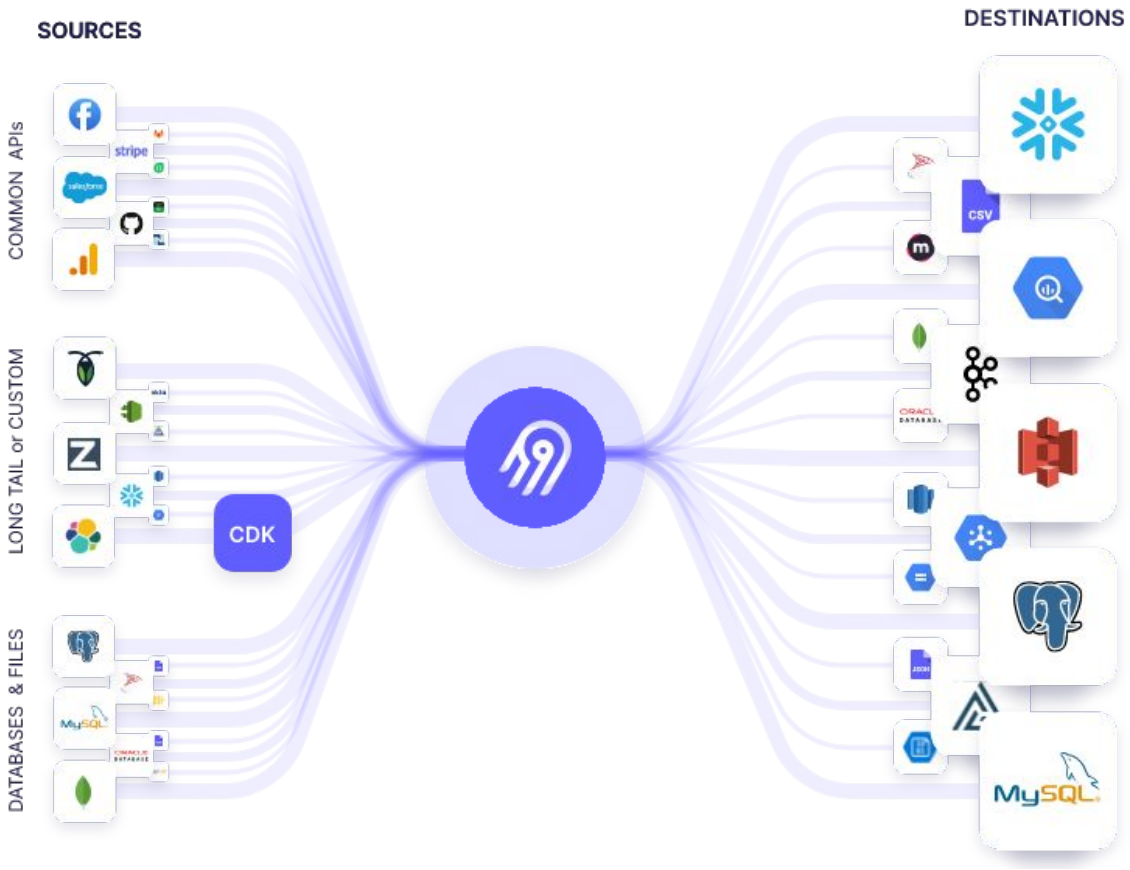
Pipeline



Stack

Hands on!

Airbyte



Stack

Hands on!