WILEY

ENABLING DISCOVERY | POWERING EDUCATION | SHAPING WORKFORCES

# DBMS
# BIG DATA


Image source: nttdata

# AGENDA:

- What is Big Data?
- What Comes Under Big Data?
- Benefits of Big Data
- Big Data Technologies
- Analytical Big Data
- Operational vs. Analytical Systems
- Big Data Challenges

Data is life;
Data is process

# Data is the key to...

Understanding exactly what happened

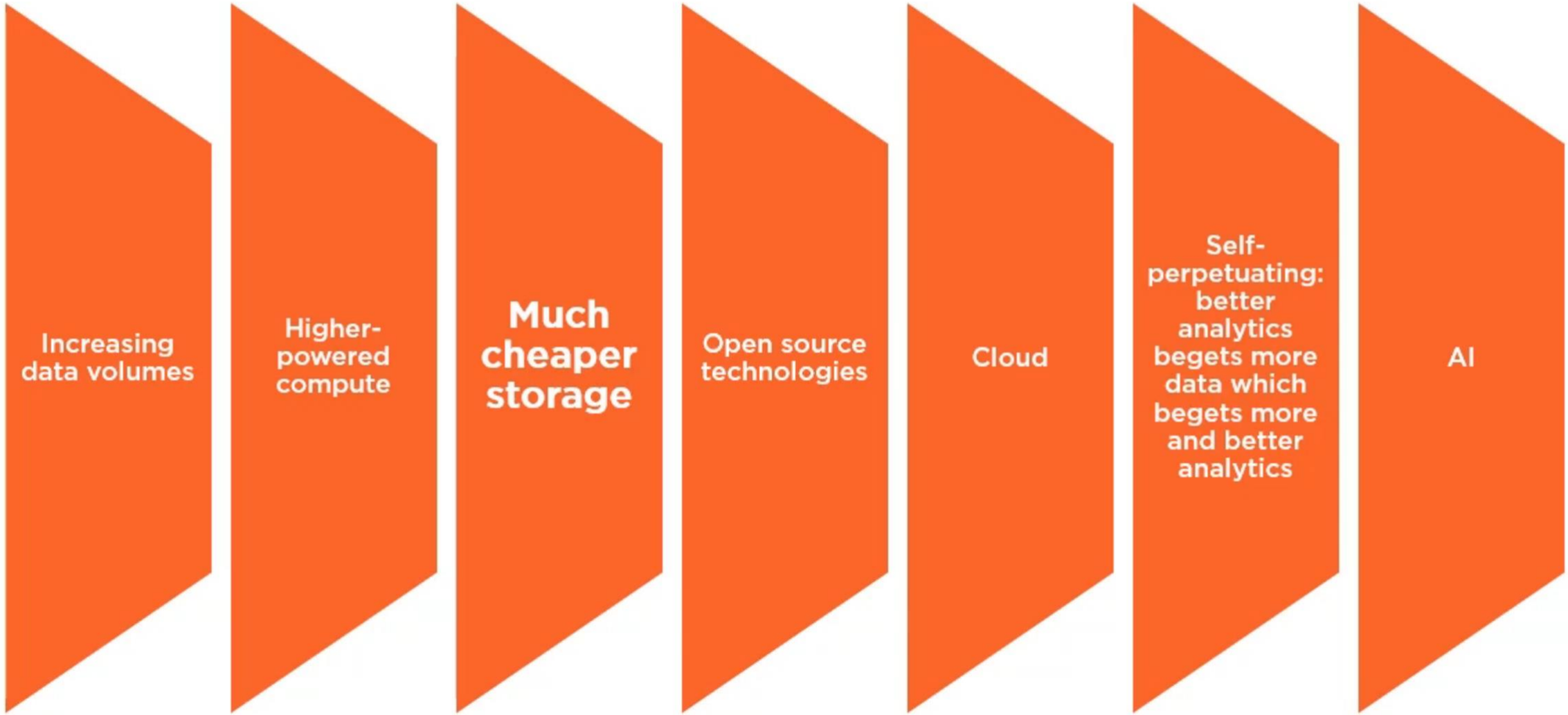
Making better decisions

Competitiveness

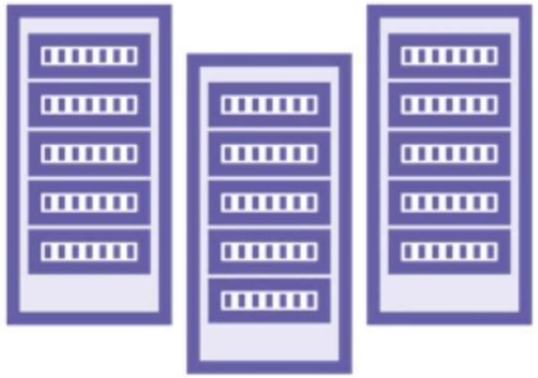
Artificial intelligence

**Digital transformation**

# What Gave Rise to Big Data?

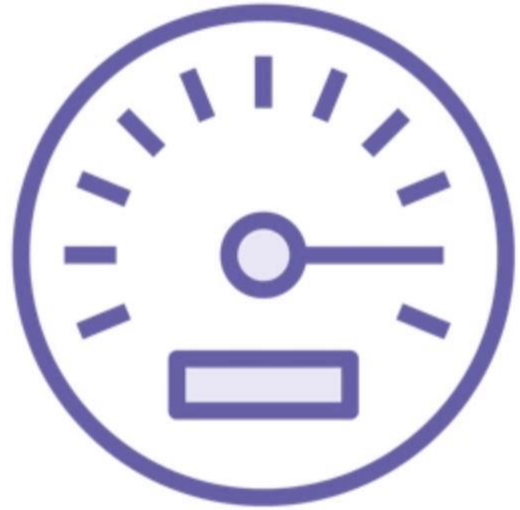


# The Three V's



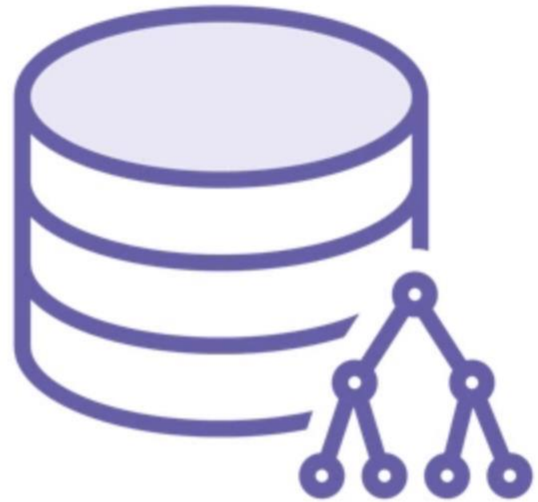
**Volume**

How much data you have



**Velocity**

How quickly data is accumulating



**Variety**

The diversity of data's structure, format, and content

# The Fourth V

**Veracity: Accuracy, quality, trustworthiness**

**Value: Big data value is a big motivator**

**Variability: Differences in data meaning**

**Visibility: Discoverability and accessibility**

**And**

- Validity
- Vulnerability
- Volatility
- Visualization

# Advanced Analytics Timeline

1990s-2000s
**Data Warehousing, BI**
Teradata, Essbase

2008
**Big Data 1.0**
Apache Hadoop

2014
**Post-Hadoop**
Apache Spark

2014
**Cloud data w'house**
Amazon Redshift

2018-Present
**Cloud data lake**
Presto, Athena

2018-Present
**AI/Machine learning**
Sklearn, TensorFlow

# The Big Change

| From | To |
| --- | --- |
| Transactional | Behavioral |
| Storage-bound | Limitless storage |
| Data exclusivity | Data inclusiveness |
| Proprietary hardware | Commodity hardware |
| Proprietary software | Open source software |
| Finite appliances | Infinite cloud resources |
| Highly structured | Schema-flexible |
| Bureaucratically curated | Governed and facilitated |

# Big Data Scenarios

Clickstream analytics: Web logs, ad-tech, A/B testing

Internet of Things (IoT): Sensor data, RFID, GPS

Social media analytics: Tweets, posts, mentions, tags

Customer: Unify marketing, clickstream, in-store, sentiment, weather

# Big Data, Data Gravity and the Cloud

Cloud services generate data and use data

Moving data is hard, and expensive

Data has “gravity”

This feeds on itself, in a virtuous (or vicious) cycle

The more data in a cloud, the more convenient that cloud is


Data gravity is the key to cloud competitiveness

# Big Data

# Big Data: A More Formal Definition

**Beyond the three/four V's and core scenarios, how can we define big data?**

**Data volume: 100s of TB into PB-scale and higher**

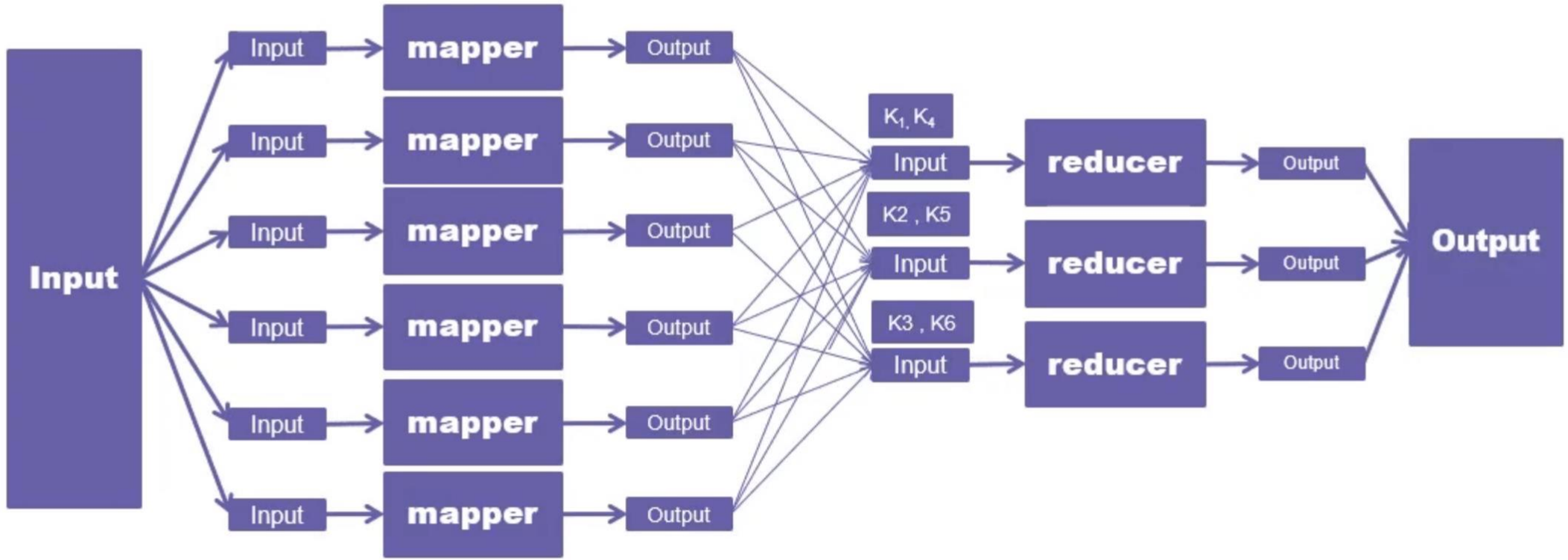
**Architecture: Parallel processing often involved**

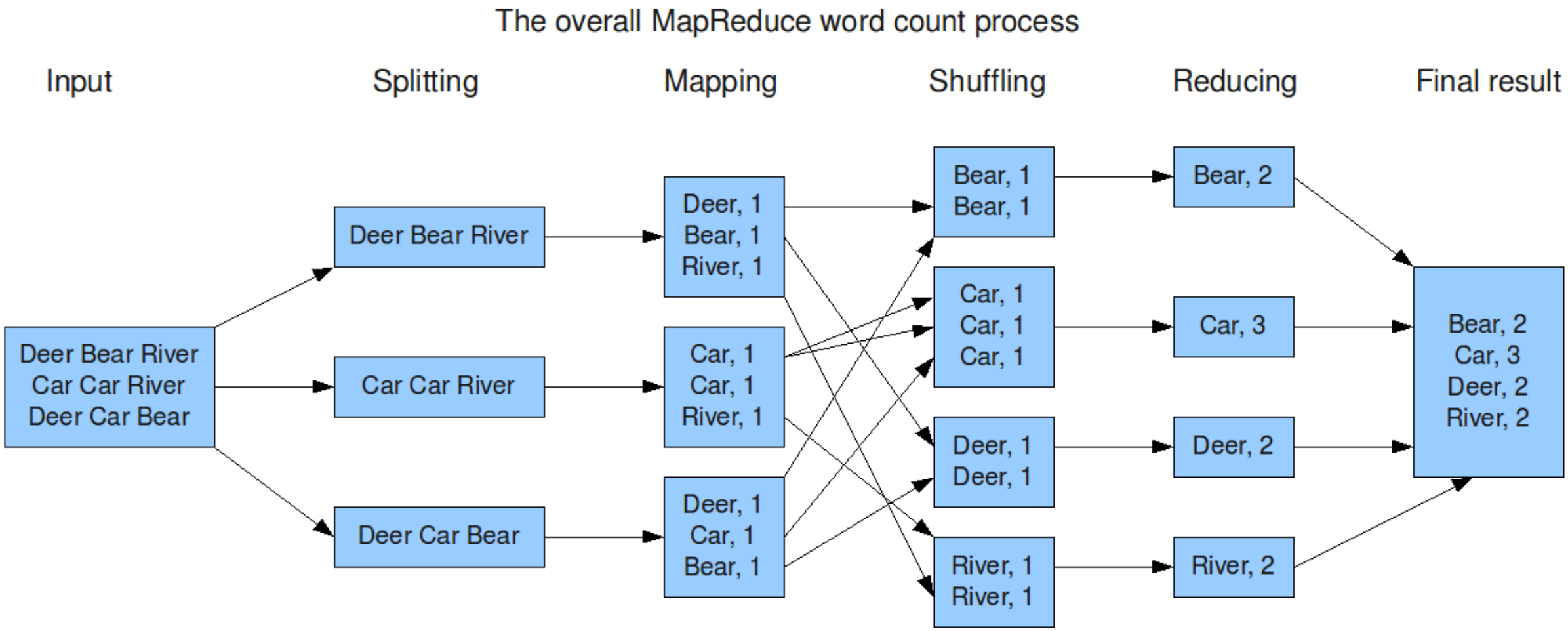
- Hadoop, Spark, data warehouse platforms

**Necessity: Processing of data sets too large for operational databases**

**Nominally: Big Data tech sometimes imposed on small data problems**

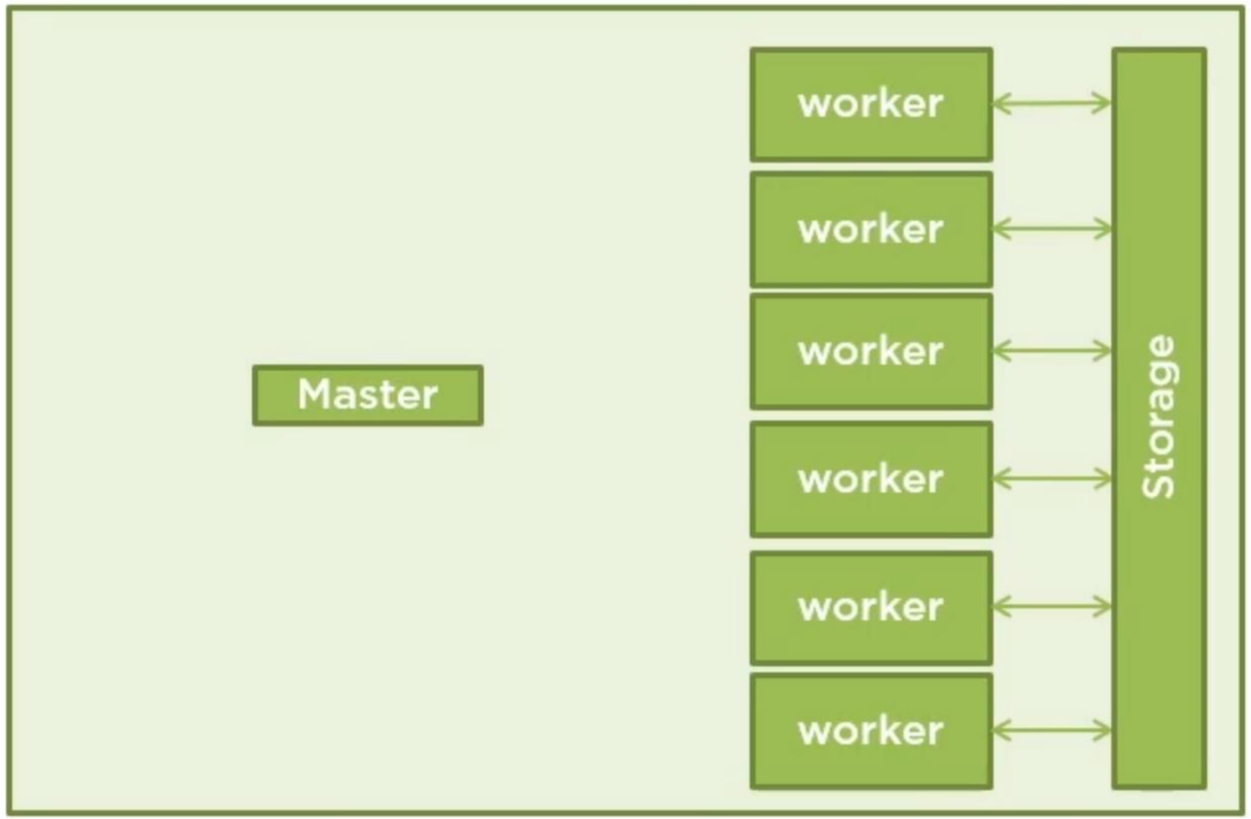
# MapReduce


Input
Input
Input
Input
Input
Input
Input
mapper
mapper
mapper
mapper
mapper
mapper
Output
Output
Output
Output
Output
Output
K1, K4
Input
K2 , K5
Input
K3 , K6
Input
reducer
reducer
reducer
Output
Output
Output
Output


The overall MapReduce word count process
Input
Splitting
Mapping
Shuffling
Reducing
Final result
Deer Bear River
Car Car River
Deer Car Bear
Deer Bear River
Car Car River
Deer Car Bear
Deer, 1
Bear, 1
River, 1
Car, 1
Car, 1
River, 1
Deer, 1
Car, 1
Bear, 1
Bear, 1
Bear, 1
Car, 1
Car, 1
Car, 1
Deer, 1
Deer, 1
River, 1
River, 1
Bear, 2
Car, 3
Deer, 2
River, 2
Bear, 2
Car, 3
Deer, 2
River, 2

# What is MPP?

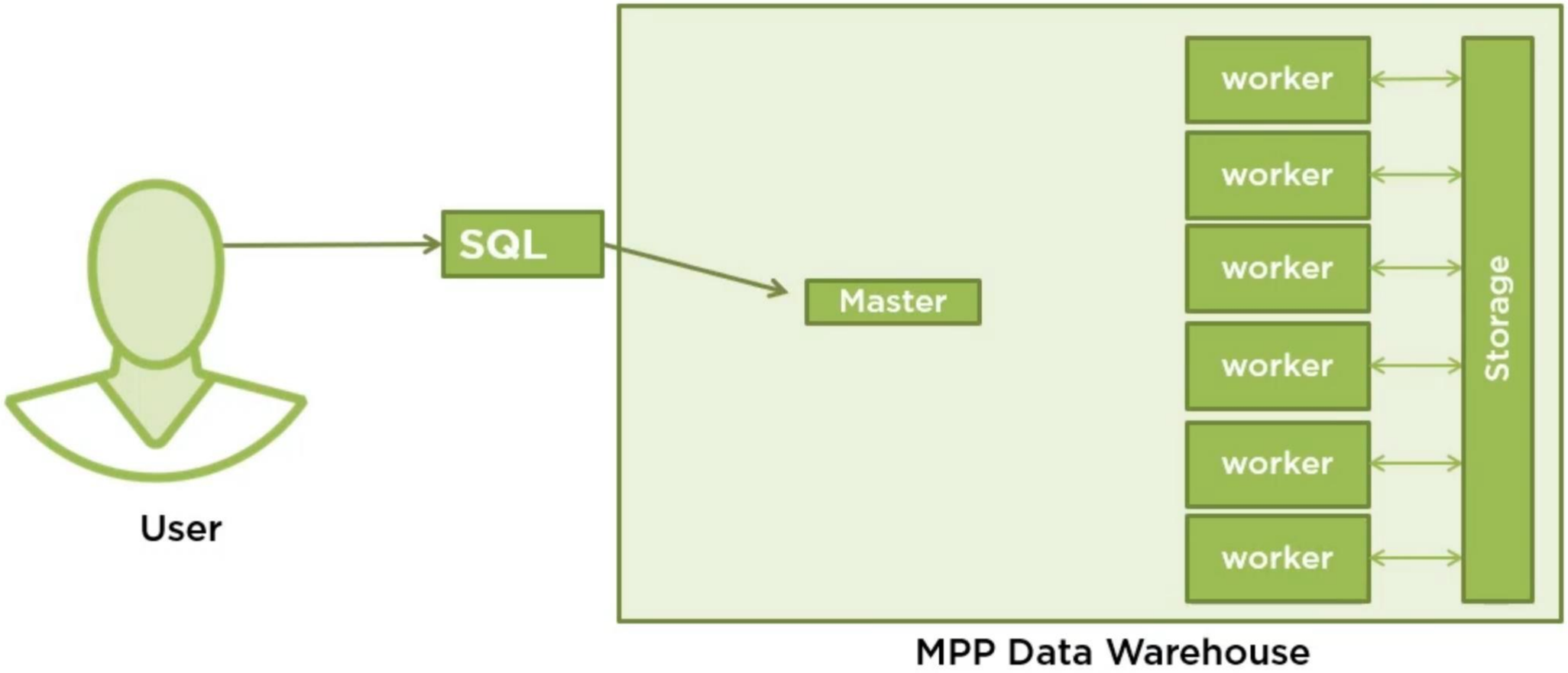
User



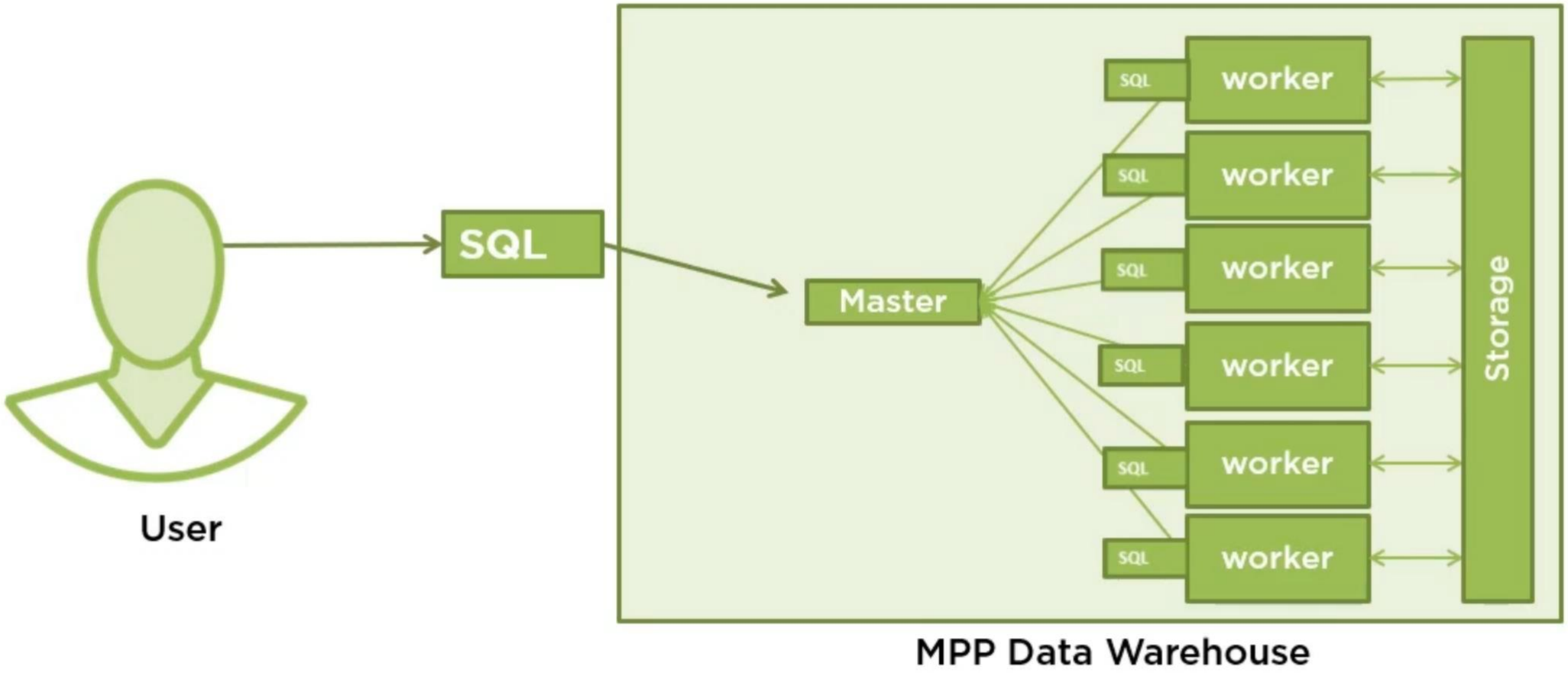

MPP Data Warehouse

massively parallel processing

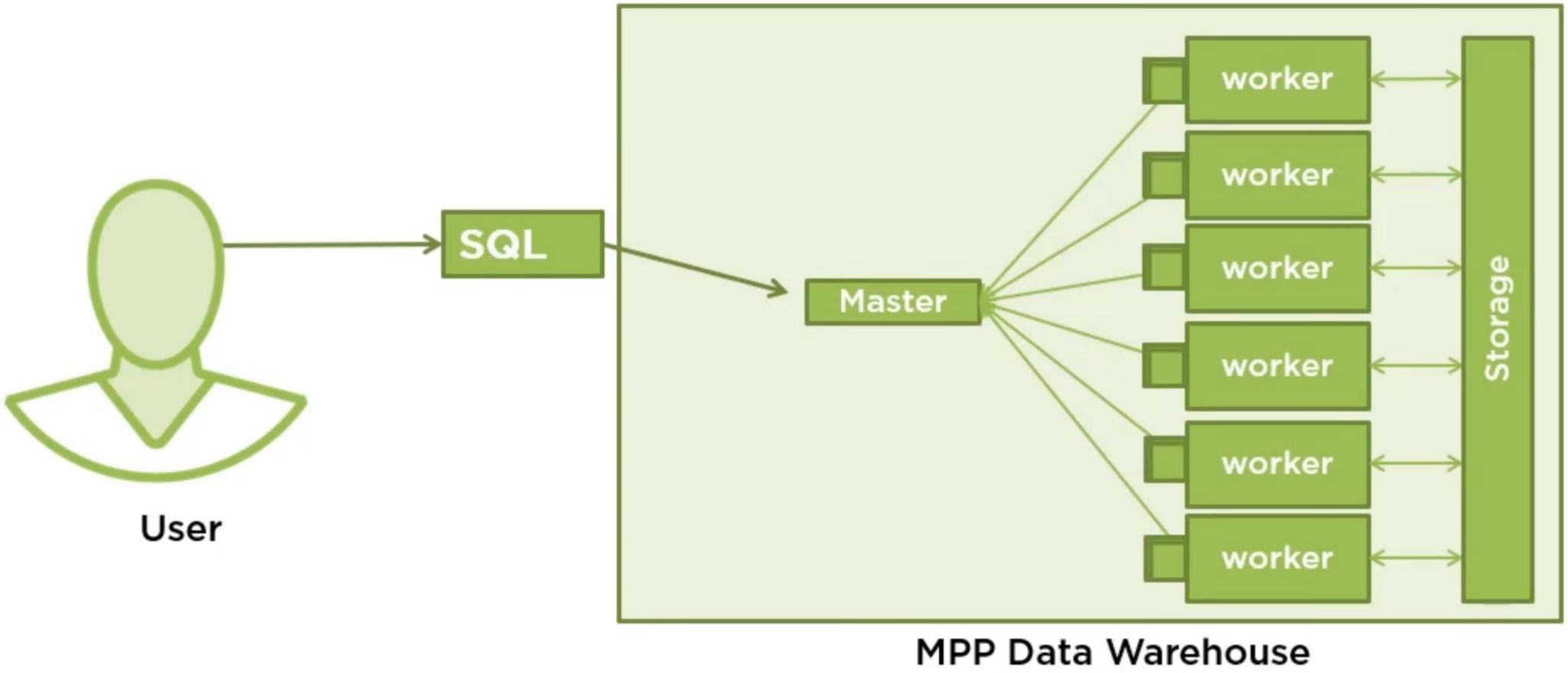
# What is MPP?


SQL
Master
worker
worker
worker
worker
worker
worker
Storage
User
MPP Data Warehouse

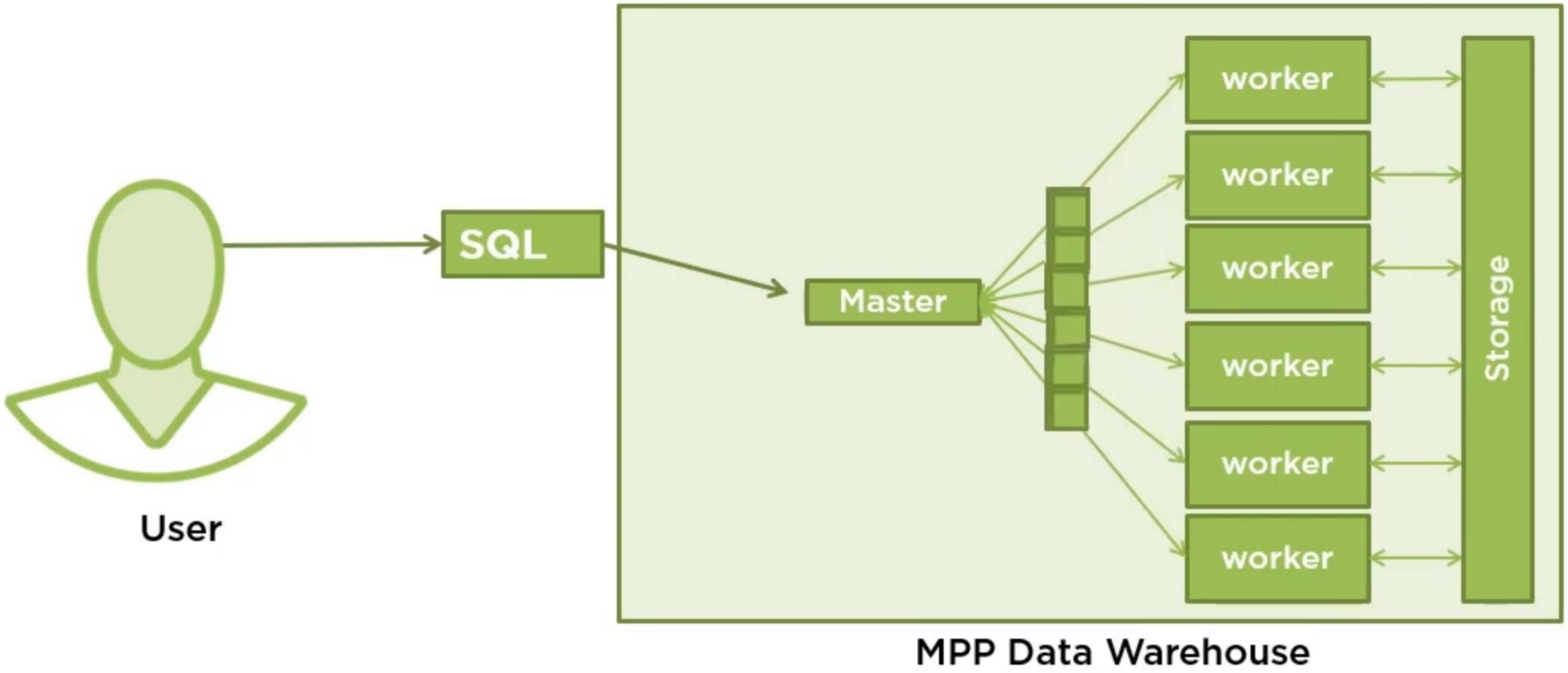
# What is MPP?


SQL
Master
SQL
worker
SQL
worker
SQL
worker
SQL
worker
SQL
worker
SQL
worker
Storage
User
MPP Data Warehouse

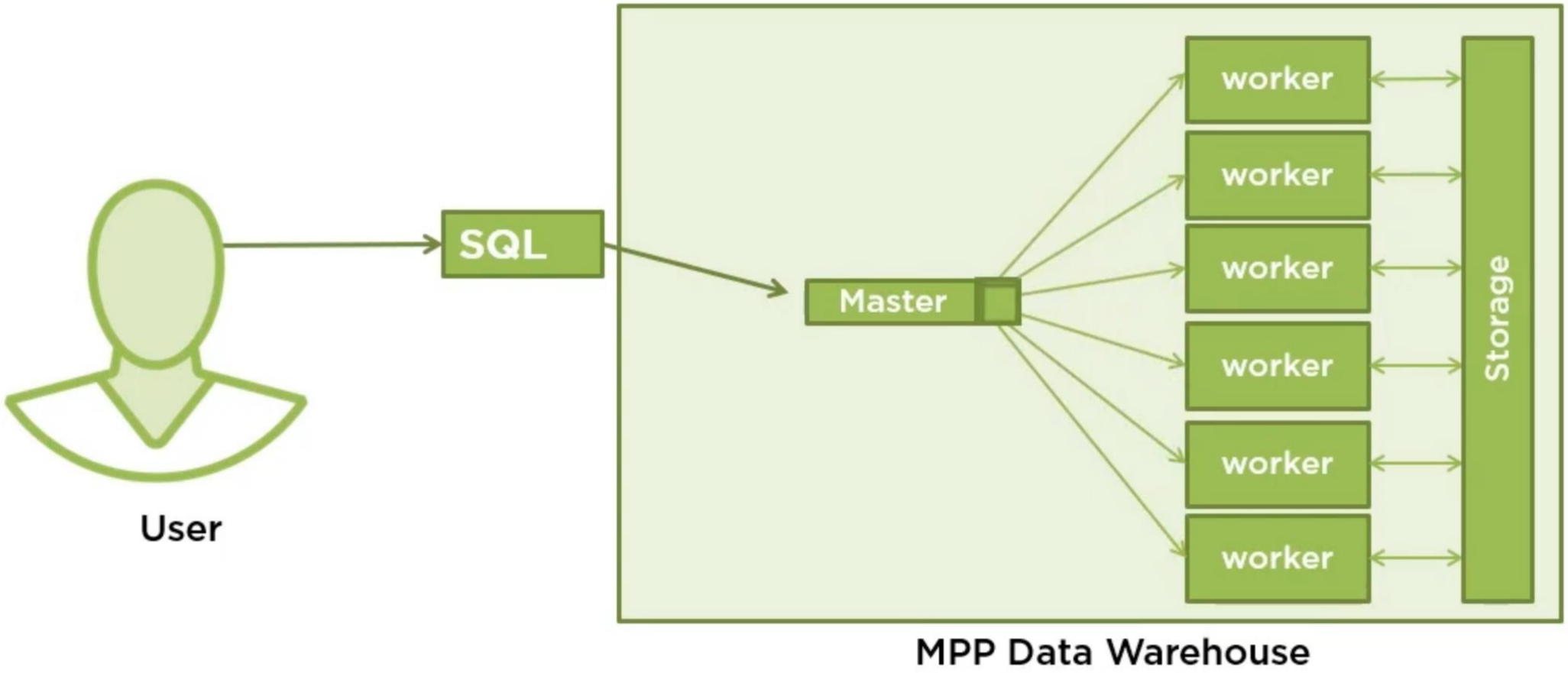
# What is MPP?


SQL
Master
worker
worker
worker
worker
worker
worker
Storage
User
MPP Data Warehouse

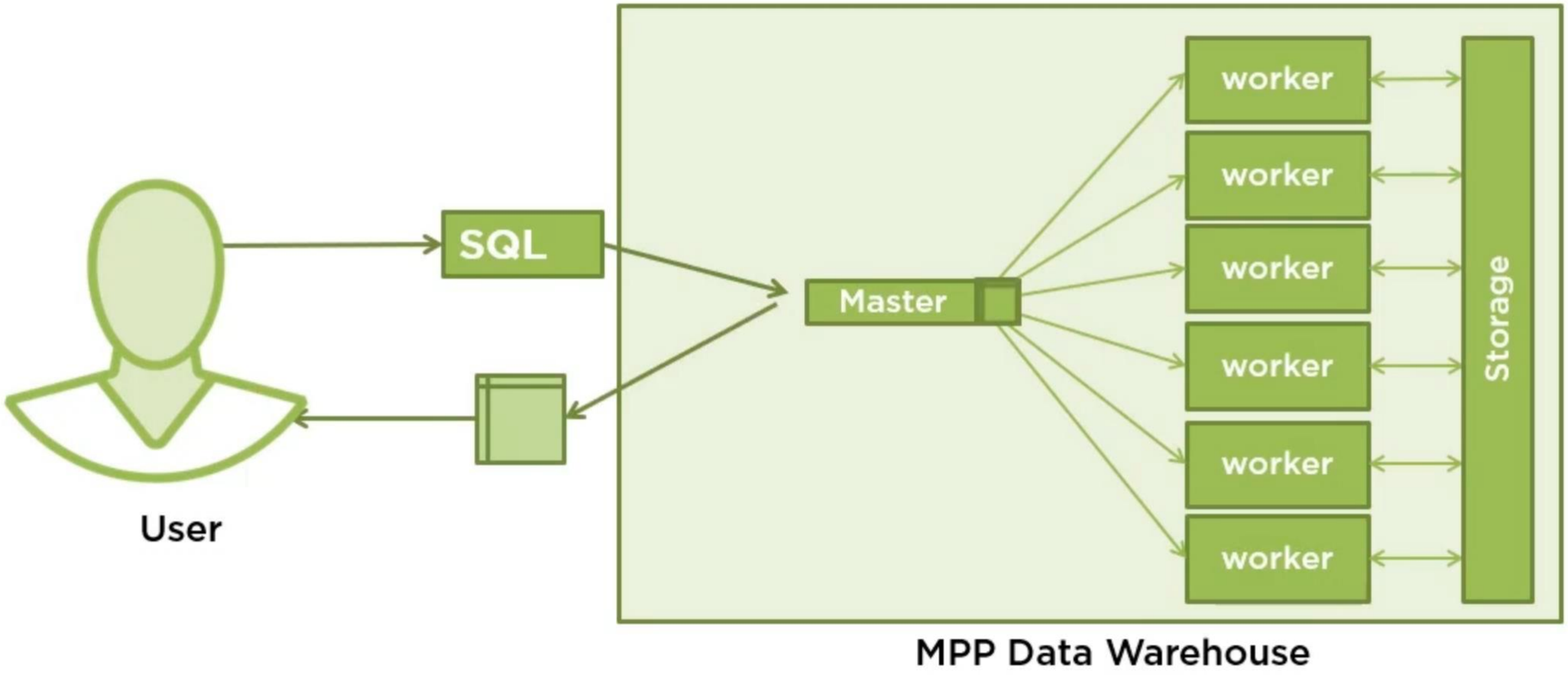
# What is MPP?


SQL
Master
worker
worker
worker
worker
worker
worker
Storage
User
MPP Data Warehouse

# What is MPP?


SQL
Master
worker
worker
worker
worker
worker
worker
Storage
User
MPP Data Warehouse

# What is MPP?


SQL
Master
worker
worker
worker
worker
worker
worker
Storage
User
MPP Data Warehouse

# Data Transformation and Pipelines

**Traditional: Extract, Transform and Load (ETL)**

- Old-school data warehousing approach
- Pull source data, load into memory, transform it, and push it into destination
- Design visually, use scripting, or a combination

**Variations: Data Prep and Internet Platform as a Services (iPaaS)**

- Self-service and internet application-oriented

**Related: Change Data Capture (CDC)**

**Newly Popular; Extract, Load and Transform (ELT)**

- Move data from source to destination first
- Then transform it on the destination platform

# Data Virtualization

**Too many data sources for convention ETL approach to be sustainable**

**Data movement too expensive**

- Regulations impact this

**Leave data where it is, process it there**

- Push down

**Federated query possible, but less important**

**Semantic model quite germane**

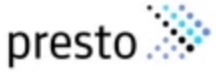
# Major Big Data Technologies


Hadoop: the big data trailblazer


Spark: In-memory big data, SQL, streaming, ML


Kafka: Event streaming FTW


Hive: The SQL-on-Hadoop original




Presto: MPP SQL-on-Anything

# Professional Roles in Big Data

Who are the people who implement, use, govern and strategize around big data?

# Analyst

**Business users**

**Actually work with the data**

**Understand it in context**

# Data Engineer

**Creates:**

Code

Visual pipelines

**For data:**

Cleaning

Transforming

Blending

Enrichment

# Data Steward

**Curates...**

**Organizes...**

**Governs...**

**Facilitates access to...**

**...the organizational data estate**

# Data Scientist

AI expert

Statistics ninja

Designs, trains, tunes

ML, deep learning models

# Machine Learning Engineer

**Perform more administrative data science tasks**

- Feature engineering
- Model deployment
- Monitoring
- Retraining

# Chief Data/Analytics/Data and Analytics Officer

**C-suite executive**

**In charge of**

Data strategy

Creating data culture

**Sometimes oversees data management and protection**

# Big Data

## TECHNOLOGIES

# Apache Hadoop


**The big data pioneer, now less dominant but ecosystem persists**

- Open source implementation of Google's MapReduce and Google File System (GFS)
- Developed at Yahoo, released: 2006

**Major vendors were Cloudera, Hortonworks and MapR**

- Cloudera and Hortonworks merged; MapR essentially folded

**Three public cloud providers each have own Hadoop service**

- Amazon EMR, Azure HDInsight, Google Cloud Dataproc

# Apache Spark


**Developed at AMPLab/UC Berkeley as an in-memory Hadoop alternative; released 2014**

**Similar in concept to Hadoop, but:**

- Lacks own clustering/file system
- Adds SQL, streaming, ML

**Available from:**

- Databricks (commercial entity founded by Spark creators)
- Cloudera, megavendors
- Cloud providers (in their Hadoop services and embedded Spark in many of others)

# The Importance of SQL

# Big Data and Data Lakes

**Big Data**

The large volumes of actual raw data

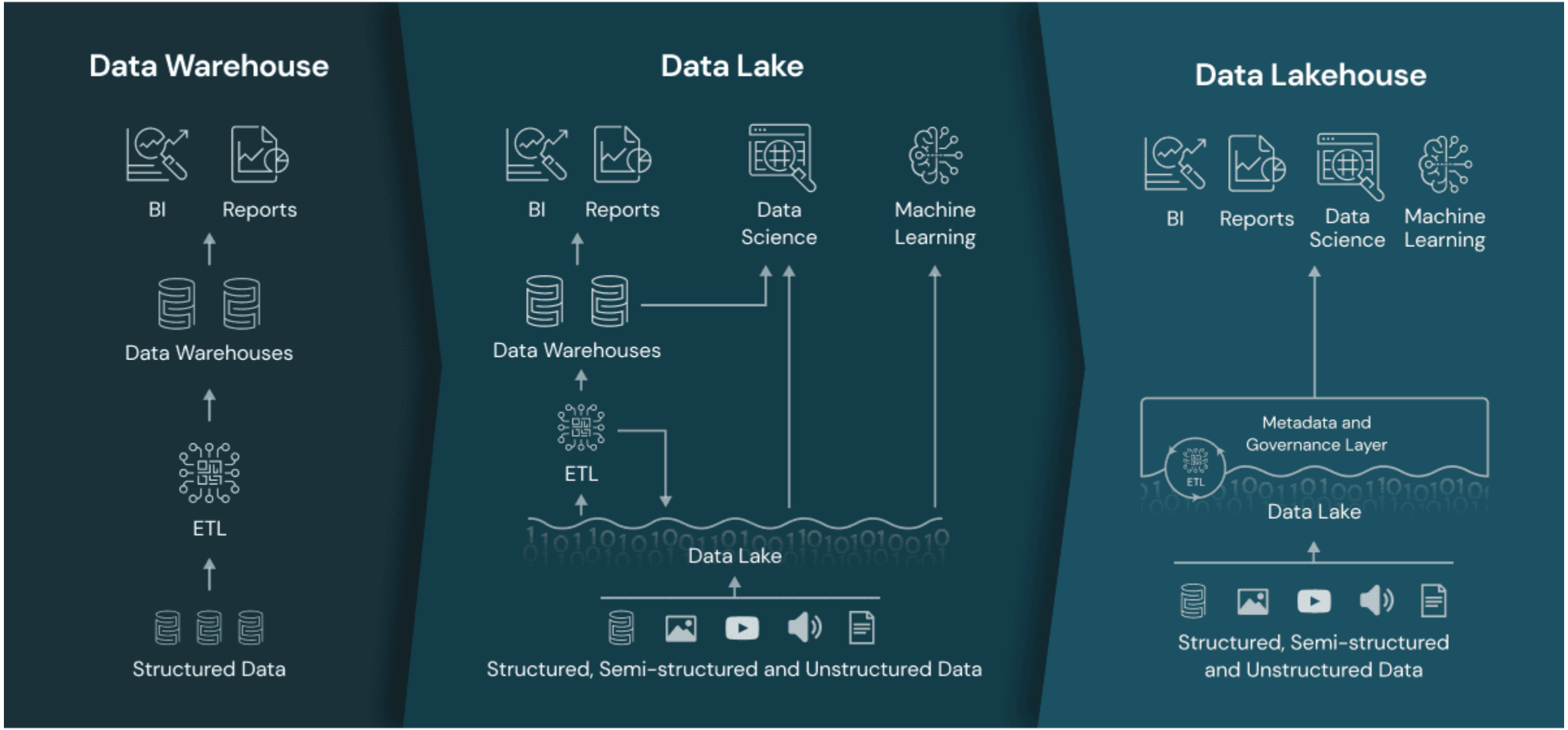
**Analytics**

The process of ad hoc query and aggregation of big data

**Data Lake**

The repository for storing big data.

Data lakes and data warehouses are both widely used for storing big data, but they are not interchangeable terms. A data lake is a vast pool of raw data, the purpose for which is not yet defined. A data warehouse is a repository for structured, filtered data that has already been processed for a specific purpose


Data Warehouse
BI
Reports
Data Warehouses
ETL
Structured Data
Data Lake
BI
Reports
Data Science
Machine Learning
Data Warehouses
ETL
Data Lake
Structured, Semi-structured and Unstructured Data
Data Lakehouse
BI
Reports
Data Science
Machine Learning
Metadata and Governance Layer
ETL
Data Lake
Structured, Semi-structured and Unstructured Data

# SQL/Data Lake Query Engines


Apache Hive: The SQL-on-Hadoop original


Spark SQL: Runs on Spark, creates DataFrames


Presto: **Data lake** darling


Apache Phoenix: Relational database in Hadoop

Data Warehouses?

# Apache Parquet


**Data lake file format of choice**

- Parquet is the new CSV

**Columnar**

- Provides compression; lowers storage and sometimes query bills

**Easily partitioned**

- Better performance

**Other columnar formats:**

- Apache ORC, RCFile

# Presto


**Created at Facebook**

- To address Hive/MapReduce being slow

**No storage engine**

- Data virtualization
- Connects to multiple back end data sources, including file formats/systems

**Works like a data warehouse engine**

- MPP, caching
- Part of warehouse/lake convergence

# Apache kafka

**Event Streaming**

Streaming + message queuing paradigms

**Created at LinkedIn**

Released: 2011

**Confluent**

Commercial entity founded by creators

**Confluent added**

SQL query, persistent event store

**Also available from**

Cloud providers, megavendors

# More Real-Time/Streaming: Open Source and Cloud-Proprietary


Apache Flink

Azure Event Hubs


Apache NiFi

Amazon Kinesis


Spark Streaming

Google Pub/Sub

# Cluster Computing

A cluster is a collection of servers (nodes)

One node is the "master" the others are "workers"

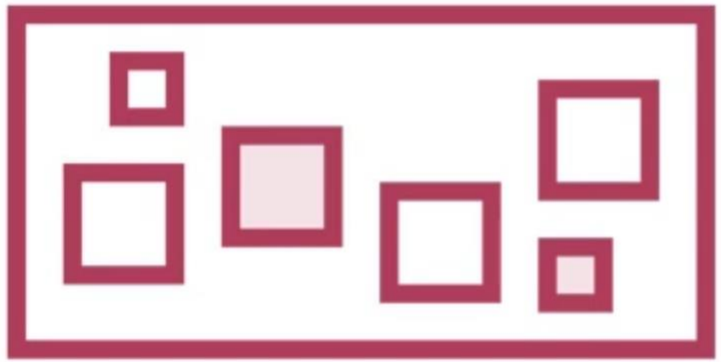
Applications talk to the master, which controls the workers

The workers do the...work, in parallel (simultaneously)

Failed nodes typically are replaced by others

Add/remove nodes to scale cluster up/down

# Containers



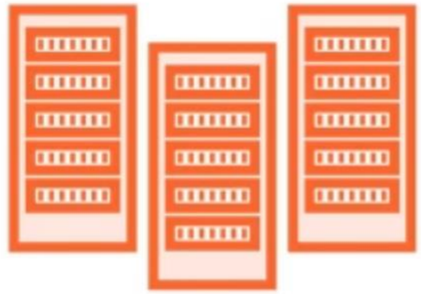
Virtual machines (VMs) are a full machine images with all software, OS, drivers

Containers are images with software, minor libraries or dependencies. Think of them as deployment packages

Containers can run on physical or virtual machines, deploy much faster than VMs can start up

Containers can be orchestrated and scripted. This makes multi/hybrid cloud work!

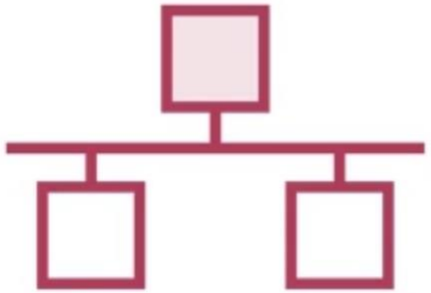
# Container Orchestration



Container deployment across clusters

Then: Mesos, Swarm


Now: K8s



K8s: *Clusters*, nodes, master, workers, pods

K8s: Deployments, ReplicaSets, services

Big Data: Hadoop/YARN, Spark

# Other Ecosystem Technologies

MPP SQL query: Apache Impala


Storage: Apache Kudu

NoSQL database: Apache HBase

In-memory columnar: Apache Arrow

Elastic stack (Elasticsearch, Logstash, Kibana)

# Data Governance/Catalog

Apache Ranger

Apache Atlas


Hive metastore is a standard

Cloud provider solutions

Third party solutions

# What is Machine Learning?

ML is based on the concept of observing an historical data set...

...and modeling how the value of a particular column (the label, or target)...

...is affected by the values of other columns (features)

# Machine Learning: Before (Big Data) and After

**Before Big Data, it was hard to store, process and train with large data sets**

- Storage was too expensive
- Servers were slow
- Clustered computing was uncommon and cost-prohibitive

**Training data had to be sampled**

- Less training data made models less accurate
- Scoring could take longer too

**Big data solves this**

- By making storage affordable
- By making compute effective

# The ML Stack


python


jupyter


NumPy


PyTorch


pandas


K
Keras


matplotlib

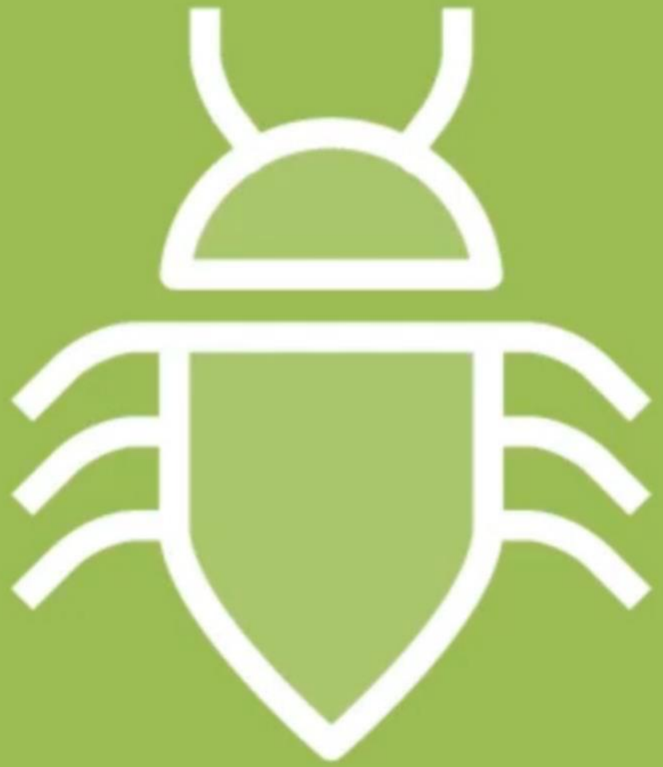

TensorFlow


scikit
learn


R

# Big Data

## VENDORS



# Vendors and technologies

Understanding the vendor and acquisition landscape helps map the technologies.

And it helps manage risks.

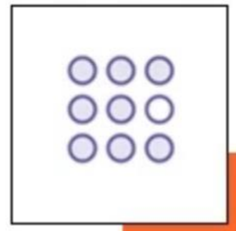
# Representative Business Intelligence Vendors

## Cloud

- Microsoft (Power BI)
- Google (Looker)
- Salesforce (Tableau)

## Enterprise

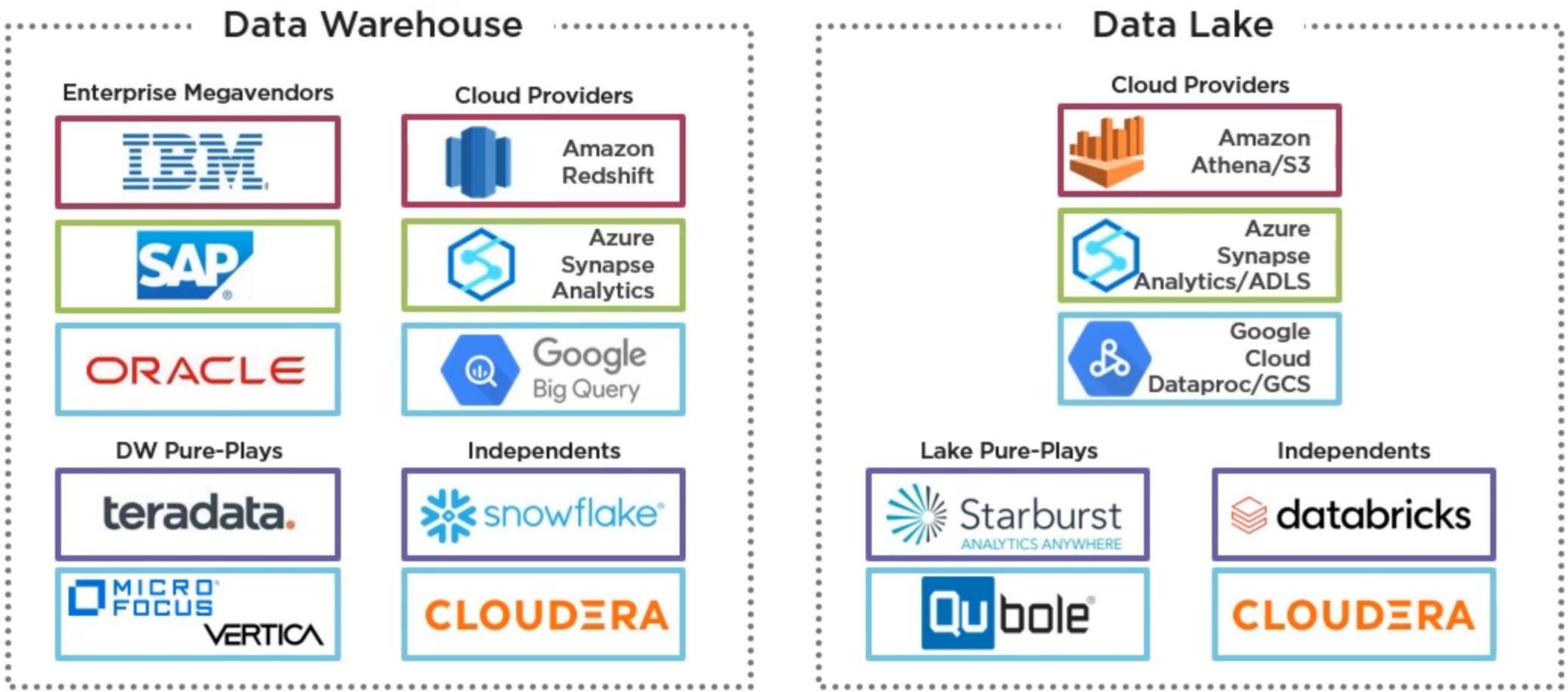
- IBM
- SAP
- Oracle



## Indies

- Qlik
- ThoughtSpot
- Sisense

# Warehouse and Lake



The data warehouse and data lake paradigms are starting to converge

# Unified Access: Presto

**SQL engine for querying lakes, warehouses and other data sources**

**As an added option: Amazon EMR, Qubole**

**As a service: Amazon Athena**

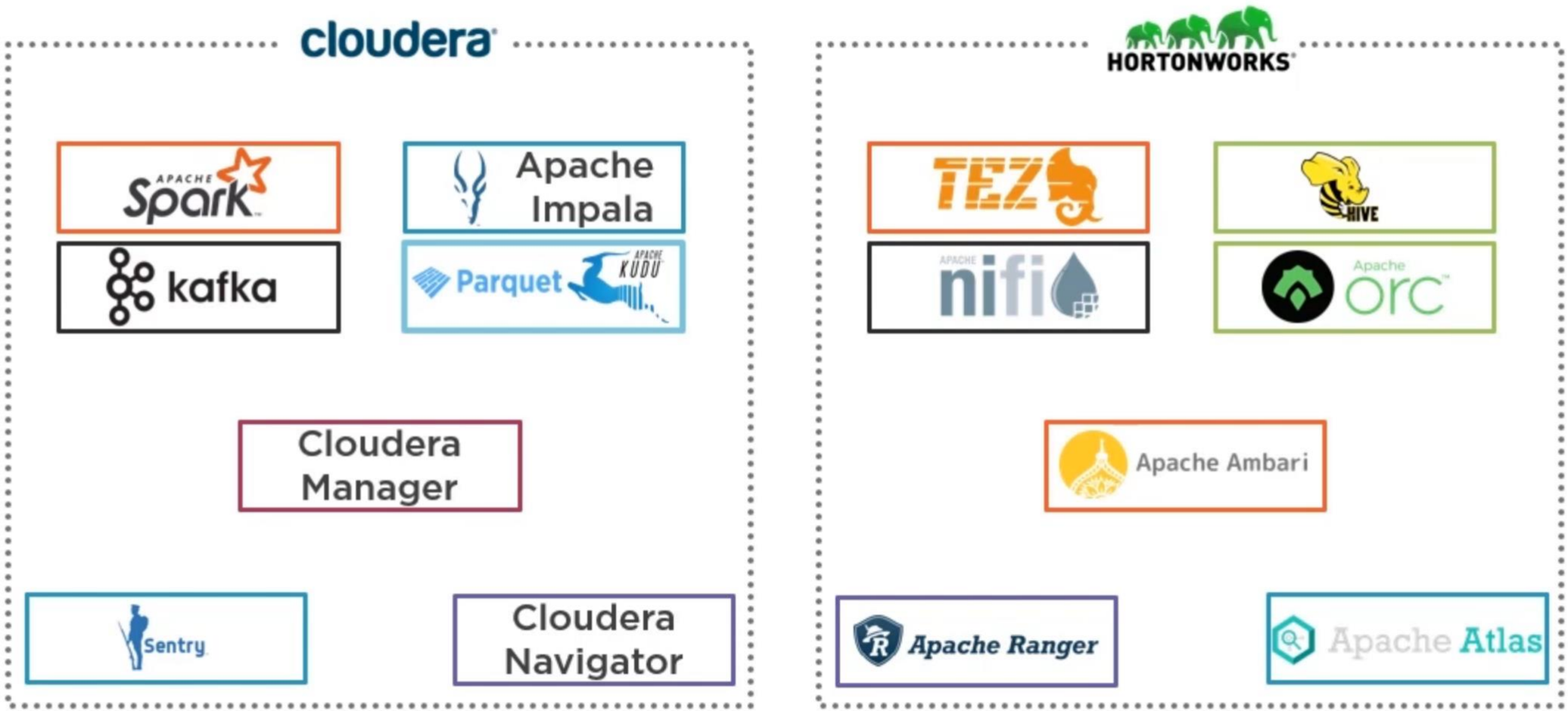
**Vendors:**

- Starburst, Ahana, Varada

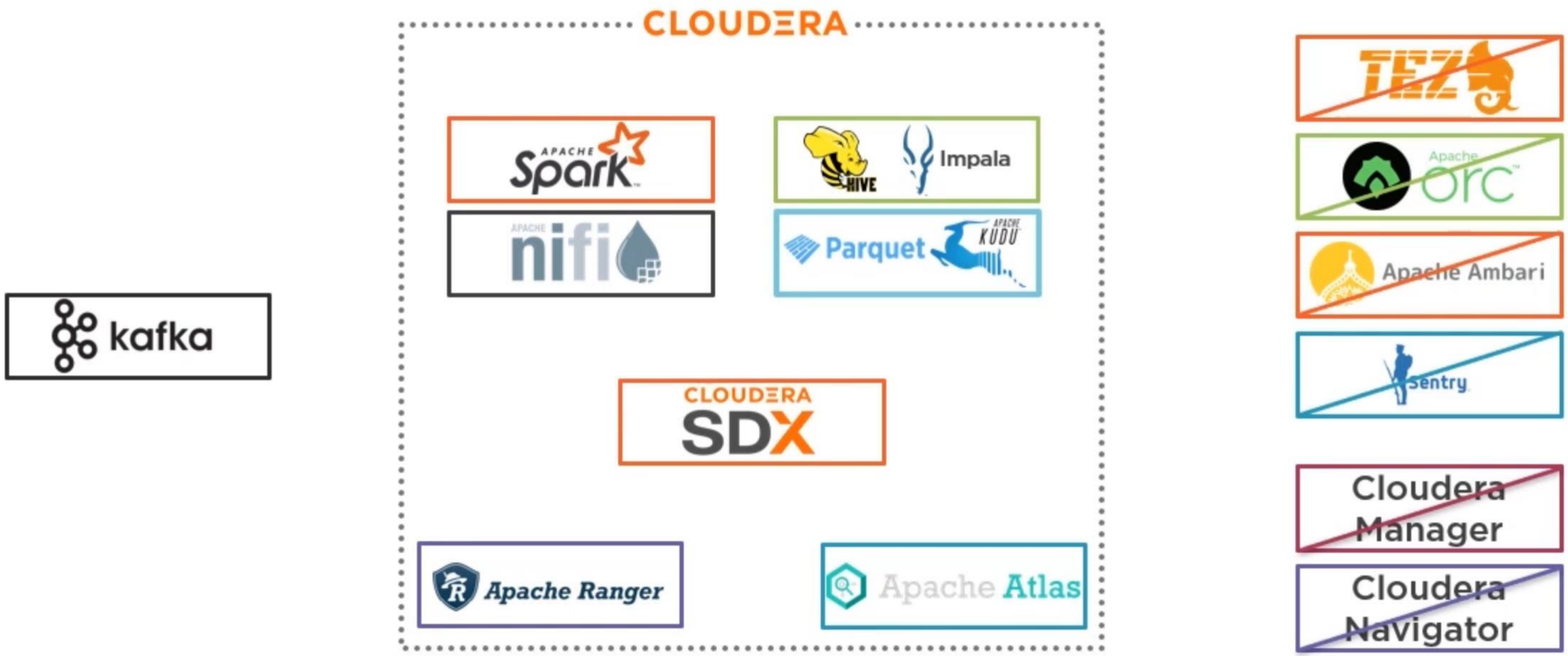
**Fork in the road**

- PrestoDB is the original project (championed by Ahana)
- Trino (formerly PrestoSQL) is a “fork” of the original project headed by Presto’s creators (championed by Starburst, with greater ecosystem momentum)

# Sibling Rivalry



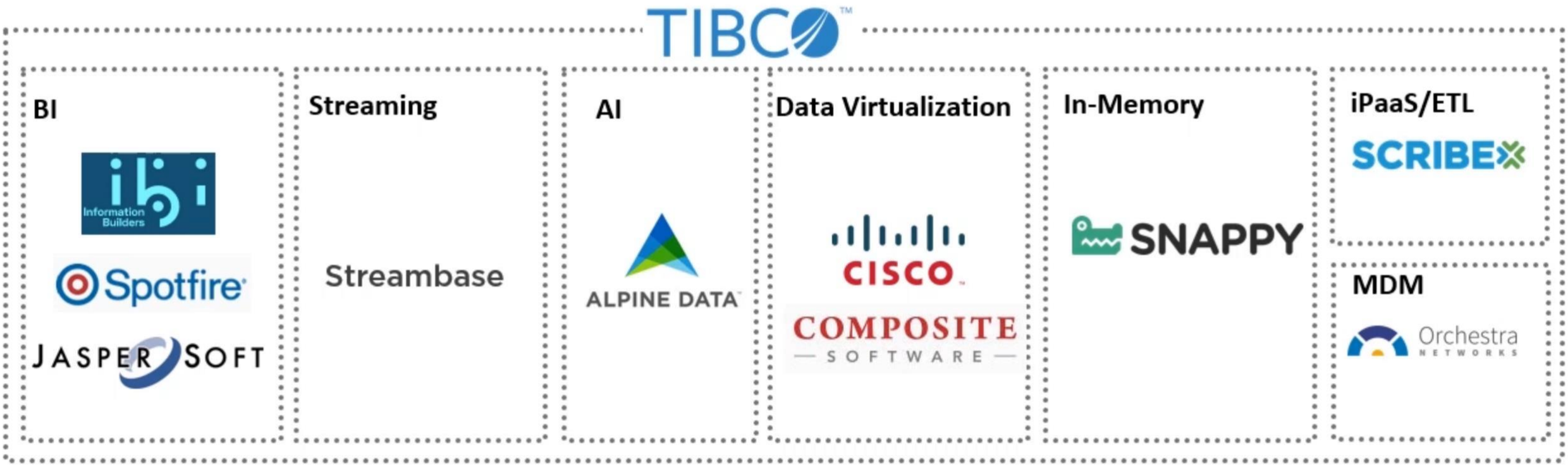
# Open Source Shakeout



# Vendors – Artificial Intelligence

# Acquisitions

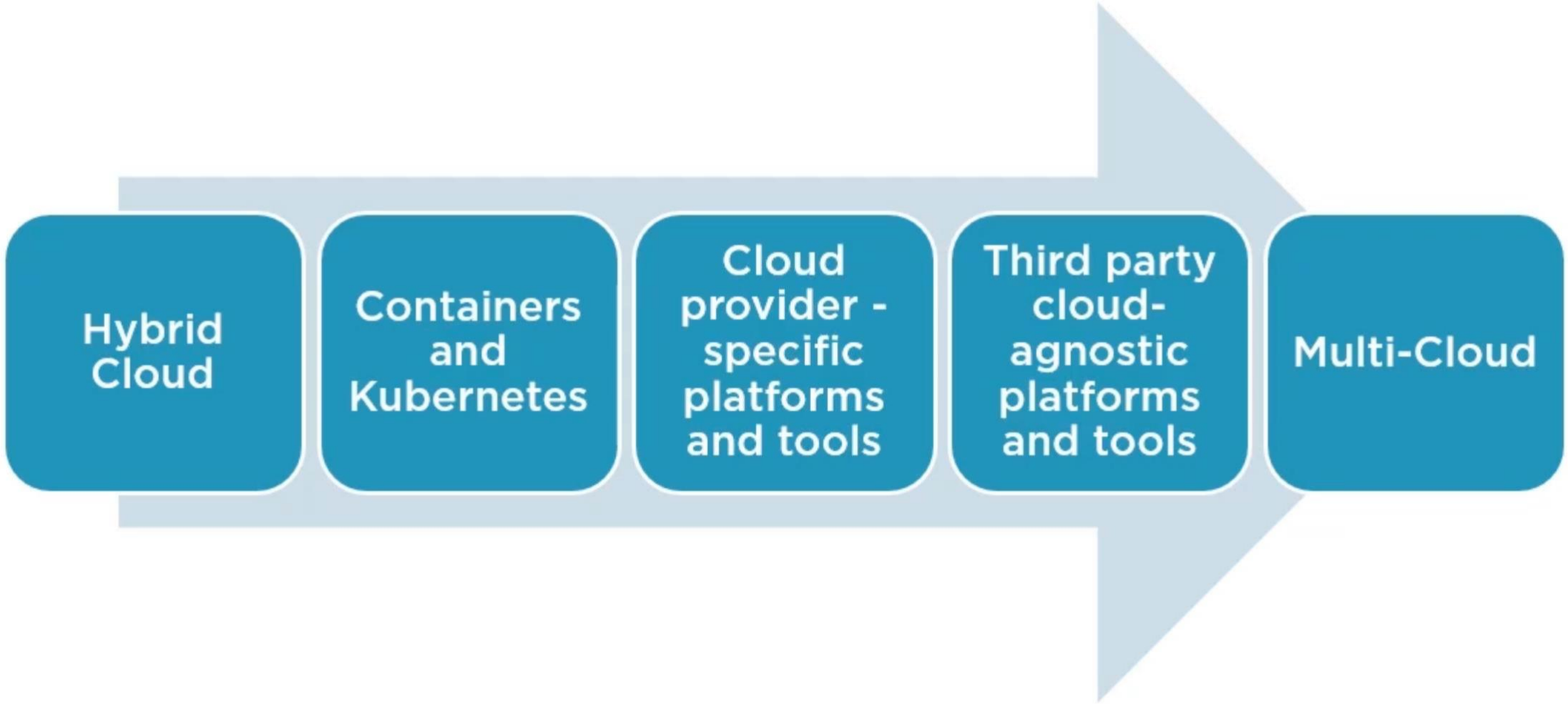
# TIBCO Acquisitions



# Big Data

## STRATEGY

# Cloud and Containers



# Workload Assessment

**Questions to ask when determining readiness and need**

## Batch Big Data

Conventional databases falling short? Exponentially more data expected?

## Streaming Data

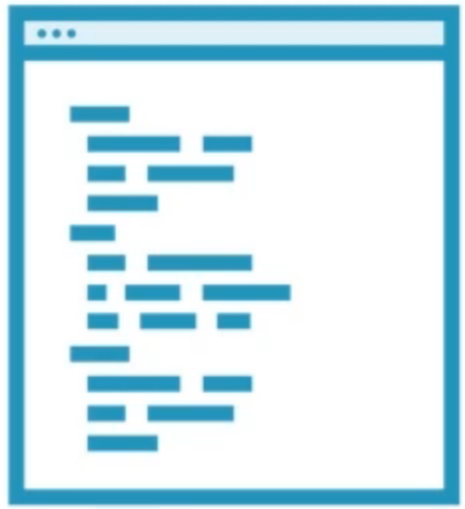
Monitoring? Sensors? IoT? Social? Logs? Time series?

## Machine Learning

Conventional analytics mature?

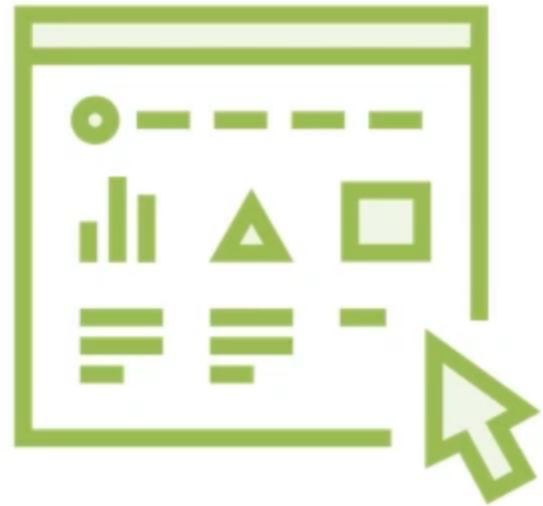
Data-driven decision making desired?

# Analytics Tooling Approach



**Code**

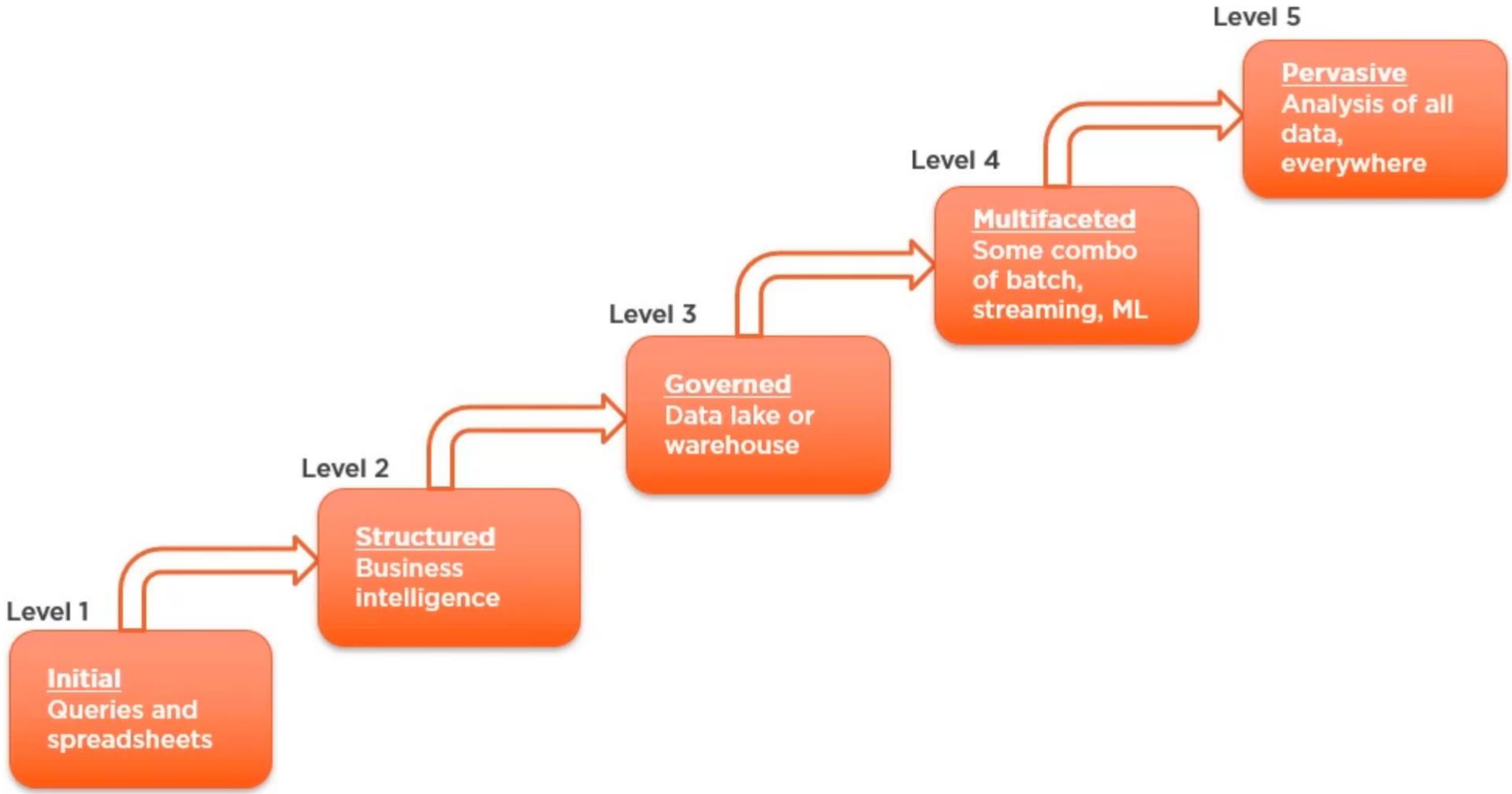
Procedural, imperative control



**Visual**

Declarative composition and specification

# Maturity Model



# Data Culture, Data-driven

# The Data Office(r)

**Data Office**

Creates data products, implements data strategy

**Chief Data Officer**

True C-level officer charged with strategy, not just technology

# Data Governance Approaches


Defense vs. offense

Regulatory compliance

Facilitation and outreach

Crowd-sourcing

Automation

# Adoption Matrix

| | Startup | Enterprise | Novice | Mature |
|---|---|---|---|---|
| BI | ++ | +++ | ++ | +++ |
| Data Warehouse | + | +++ | ++ | ++ |
| Data Lake | +++ | ++ | + | +++ |
| Streaming | ++ | + | | ++ |
| AI | + | ++ | | +++ |
| AutoML | + | ++ | ++ | + |
| Data Pipelines | ++ | ++ | + | +++ |
| Containers | ++ | +++ | | ++ |

**Cloud, containers and clusters are all related**

**Workload, tooling approaches depend on requirements and tech culture**

**Move up the maturity model, establish data culture**

**Data Office, data governance are keys to getting there**

**Making data work for you is within reach**

- Adopt technologies appropriately
- Deconstruct the big question and address each smaller one

# Concepts: Big Data

Cliché:
Volume, velocity, variety


Literal:
100s of TB or higher


Credo:
Aggregations/analysis on raw data, in standalone files

Business:
Analyze data that is:
Relevant & important;
not conformed to traditional systems

Technology:
Hadoop was foundational;
Spark is successor

# Concepts: Data Lakes

Euphemism for Hadoop and Big Data?

Storage systems and agnostic file formats together treated as virtual database

Multiple engines against same data

Initial importance of HDFS

New importance of cloud object storage

# Concepts: NoSQL

Big Data tie-in: semi-structured data + schema flexibility

Important producer of data for Big Data analytics

Big Data and NoSQL overlap, in HBase

Dominant indies: MongoDB, DataStax

Cloud providers taking market share

Most NoSQL platforms now support SQL!

# Concepts: Internet of Things

**IoT: Internet connectivity for low-powered devices**

**Provides telemetry, sensor data**

**Time series format works well for analytics**

**Broad use cases:**

- Maintenance, remote monitoring and asset tracking

**Common applications:**

- Preventive/proactive maintenance
- Usage/traffic data for municipalities
- Social media sentiment analysis
- Financial market data analysis
- Consumer: thermostats, appliances

# Concepts: Machine Learning/AI

**Historical data, relationships can be modeled to support predictions of future outcomes**

**Numerous algorithms and frameworks, open source and proprietary**

**Picking the right algorithm and "hyperparameter" value beyond most developers**

- But automation is emerging

**Development, deployment, monitoring and managing are all needed.**

- Some are more evolved than others

# Concepts: MapReduce & Massively Parallel Processing

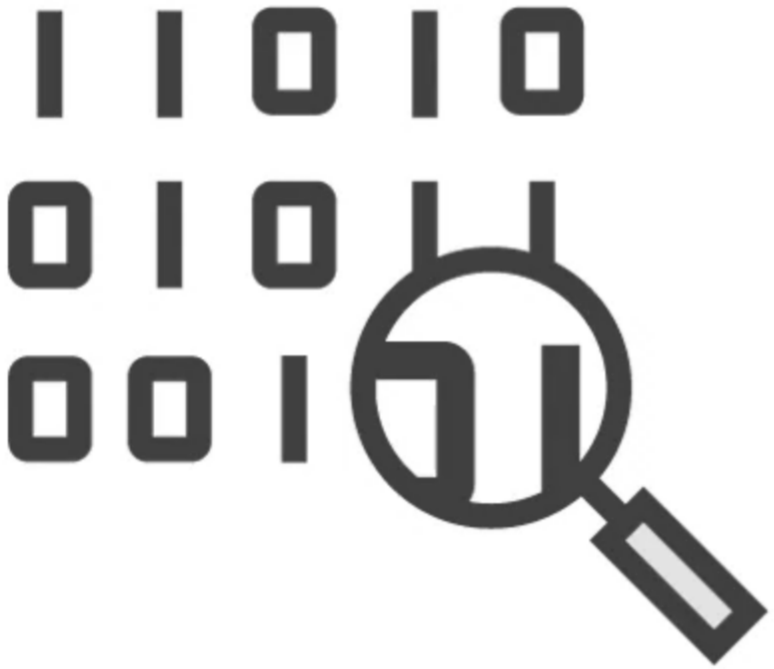
**Both algorithms based on divide and conquer for large data volumes**

- Create a cluster with lots of servers
- Node gets subset of data to work on
- Work in parallel; output quickly
- Got more data? Add more servers
- Cloud, elasticity work well here

**MR: two passes (parsing and aggregating)**

**MPP: partitions query across RDBMS nodes**

# Concepts: Streaming



**Streaming data produced is high-volume/ small-payload**

**Bread and butter of real-time analytics**

**Produced in various scenarios:**

- Internet of Things/sensors, financial markets, social media, Web analytics

**Small number of dominant open source technologies**

**Major cloud providers offer proprietary streaming platforms**

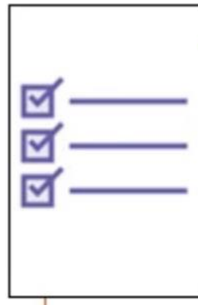
# Concepts: SQL


Structured Query Language, around since the 70s

Huge pool of technologists with basic competency


Newer startups tried to abandon it; failed

Universally understood, declarative query language too valuable to abandon



Used by:

- Operational relational databases
- Data Warehouses
- NoSQL platforms
- Big data, data lake query engines

# The Major Big Data Services

| | | | |
|---|---|---|---|
| Simple Storage Service (S3) | Athena | Elastic MapReduce (EMR) | Redshift |
| Glue | Data Pipeline | DynamoDB/ NoSQL | Relational Database Service (RDS) |

# Services: S3

Amazon's cloud object store...

...And its data lake, too

"Special relationship" with EMR

Virtually all AWS data services connect to S3

Buckets, folders, files

# Services: Athena


SQL query layer over S3

Based on Presto, Apache Hive

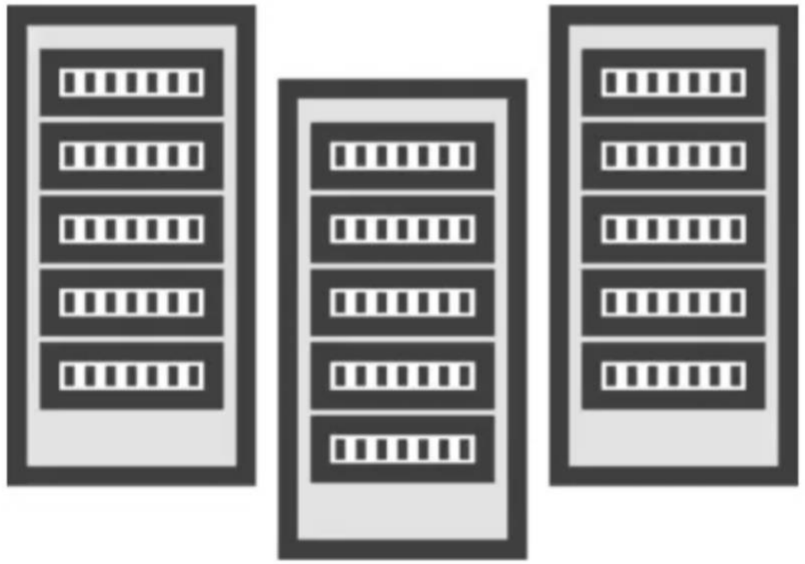
EMR = lots of options
S3 + Athena = all you need

Consumption–based pricing

Works great with columnar file formats, e.g. Parquet, ORC

Leverages Glue data catalog

# Services: Elastic MapReduce (EMR)



**The big daddy of AWS Big Data**

**Apache Hadoop, Spark, and lots more**

**Tightly integrated with S3, via EMRFS**

**Primarily about big data/data lake analytics but also handles other workloads:**

- Streaming data
- Data integration
- Machine learning/AI

**And then there's the ecosystem**

# Services: Redshift

AWS' cloud data warehouse; pioneer in category

Was AWS' fastest-growing data service for years

Elastically scalable but does not use S3 for storage

Clusters run 24/7, with associated costs

Huge ecosystem support

Big competitor is Snowflake, which can run on AWS *and* use S3

# Services: Glue

**Data catalog and integration/prep platform**

**Crawlers, Tables and Jobs**

**Tight integration with S3, DynamoDB, Redshift, RDS and external databases (last three via JDBC)**

**Jobs' visual interface generates code that runs on (serverless) Spark**

- Code uses high-level Glue API
- Editable, to a point

**Glue data catalog is strategic**

# Services: Lake Formation

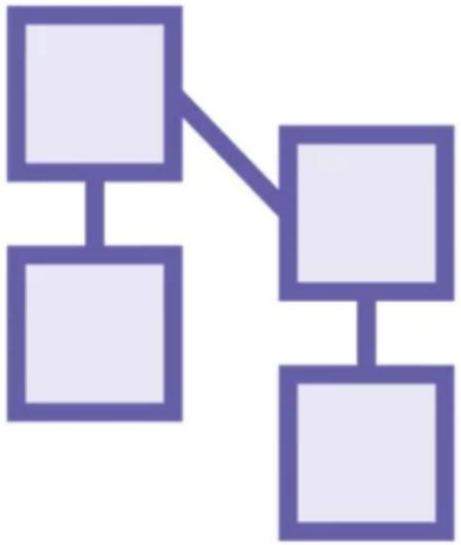
**Service for data lake creation**

**Automation of:**

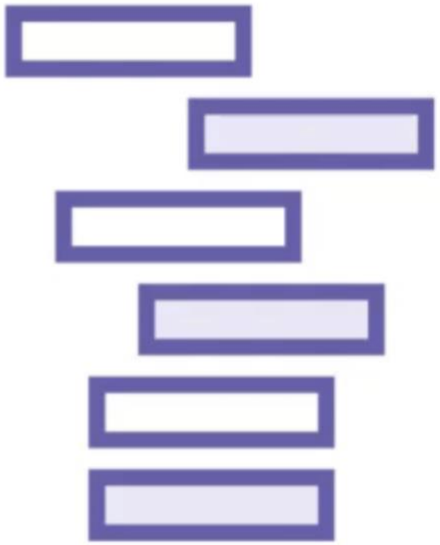
- Access controls
- Partitioning
- Deduplication (ML-based)
- Cleansing
- Classification

**Builds on/heavily leverages Glue**

# Services: Data Pipeline



Visual “boxes-and-lines” service for data integration



Tight integration with S3, Redshift, DynamoDB

Jobs can be scheduled or run on-demand

Code-free but not simple

# NoSQL: The Full AWS Story

## DynamoDB

Key-value store that integrates with numerous other AWS data services

## DocumentDB

Document store

## Neptune

Graph database

## Apache HBase on Elastic MapReduce

Column family store

## SimpleDB

Deprecated key-value store, superseded by DynamoDB

# What About Operational Databases?

**Relational Database Service (RDS)**

- Oracle, SQL Server
- MySQL, MariaDB
- PostgreSQL (aka “Postgres”)
- Aurora
  - Cloud-native/serverless
  - MySQL- and Postgres-compatible

# Mapping the Services

| Big Data Technology | Amazon Service |
| --- | --- |
| Data Warehouse | Redshift |
| Data Lake | S3, Athena |
| Batch Analytics | EMR |
| Relational | RDS |
| NoSQL | DynamoDB |
| Streaming Data | Kinesis, MSK |
| Data Integration | Glue, Data Pipeline |
| Artificial Intelligence | SageMaker |

# References

https://www.oracle.com/in/big-data/what-is-big-data/

https://aws.amazon.com/

https://www.ibm.com/in-en/analytics/big-data-analytics