ONEWAY

SOLUTION

One Way Solution

Batch-ETL

Data Engineering – [Day 2]

LUAN MORENO

CEO & CDO

Data Engineer & Data Platform MVP

Confluent Certified Developer for Apache Kafka [CCDAK]

Foundation, Apache Spark & Databricks

Test Data Engineering – [Day 1]

Data Engineering – Foundation, Apache Spark & Databricks

1

Big Data
Cloud Computing
Apache Hadoop
Big Data as a Service [BDaaS]
Apache Fundamentals
Apache Spark 3.0 Features
Databricks
Big Data Architectures

Agenda

2

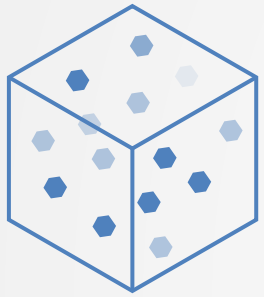
Data Lake
Spark Lifecycle
Storage Solutions for Spark
Data Lakehouse
Delta Architecture

One Way Solution


Everyone has talent. What is rare is the courage to follow the talent to the dark place where it leads.
Erica Jong
quotefancy

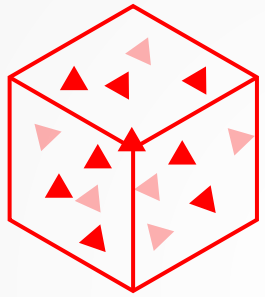
RDBMS

SQL Server | Oracle | PostgreSQL | MySQL



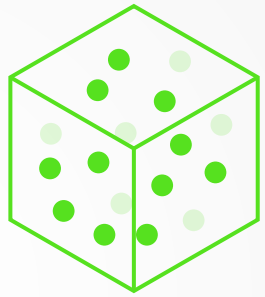
NoSQL

MongoDB | Cassandra | Redis



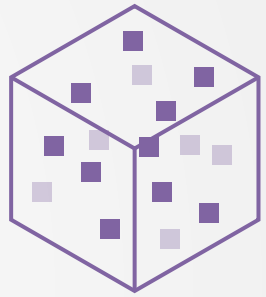
Web

Web App | Mobile App

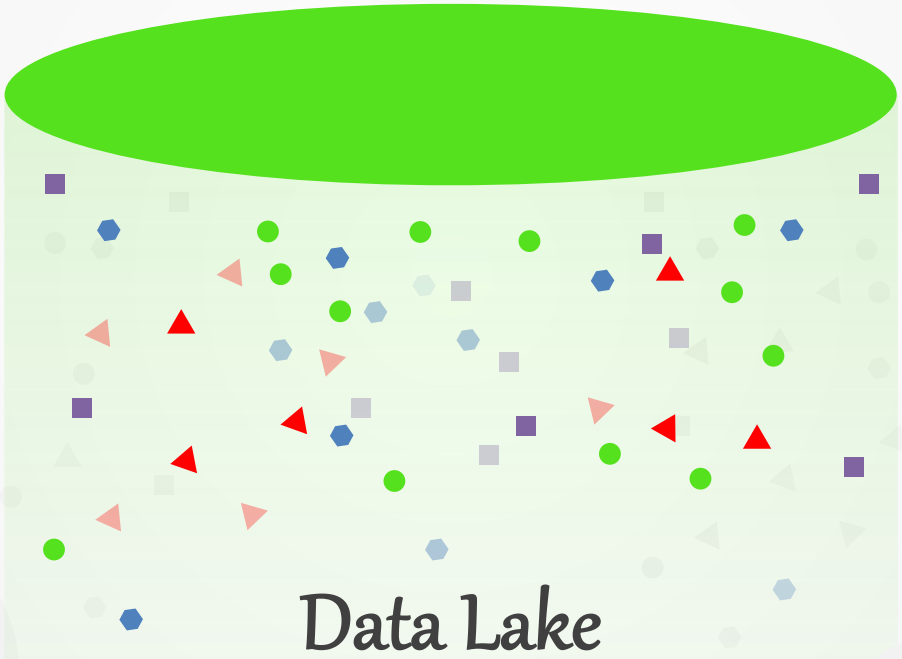


Files

CSV | XML | JSON

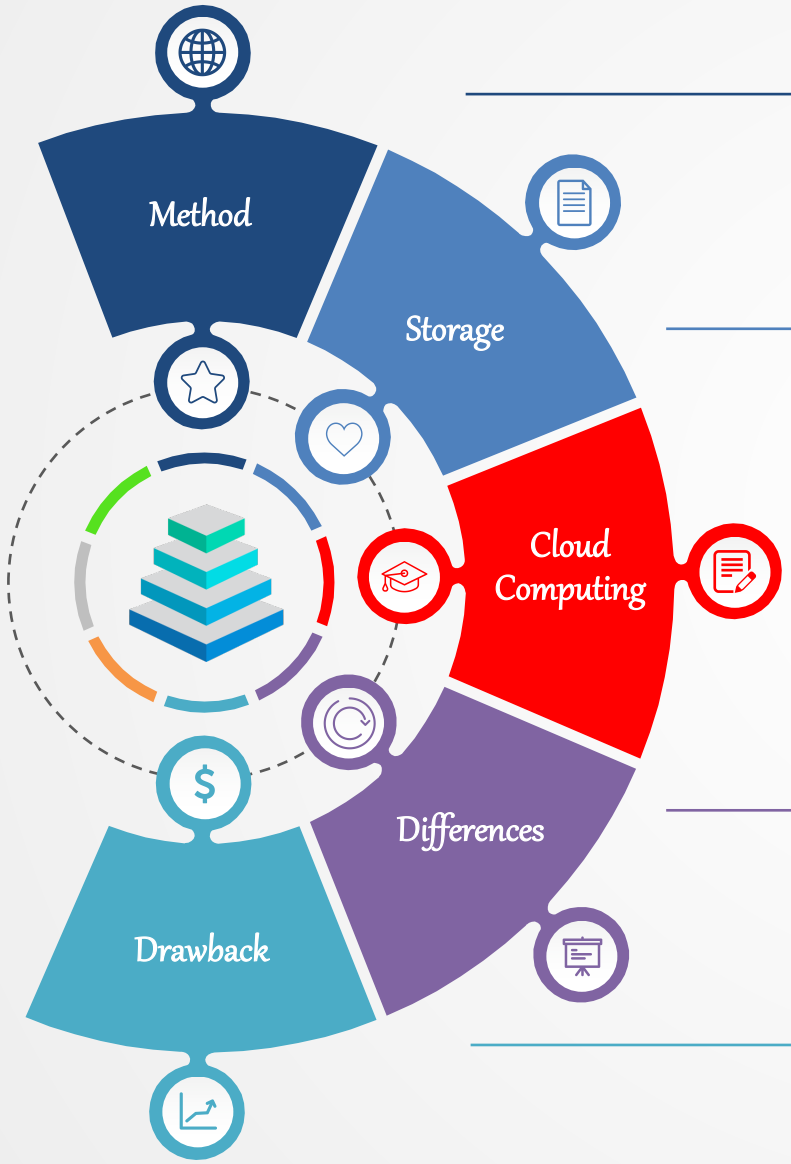


Process



repository of raw data

Data Lake [Concepts]



Repository of Raw Data [Data Democratization]
Data Lake – James Dixon in 2010
Democratization of Data – [Unsiloed Data]

Structured – Relational Databases
Semi-Unstructured – CSV, Logs, XML & JSON
Unstructured – Emails, Documents, Binaries, Audio & Video

Azure – Azure Blob Storage & Azure Data Lake Storage Gen2
AWS – Amazon S3 & AWS Lake Formation
GCP – Google Cloud Storage

Data Mart – Subset of Decision Support for Departments
Data Warehouse – Decision Making Support for Enterprise-Level
Data Lake – Raw Data [Source of Truth]

Data Swamp – Highly Disorganized Data
Data Governance – Management & Control of Data
Data Quality – Accuracy and Data Veracity
Data Security – Protecting Digital Data

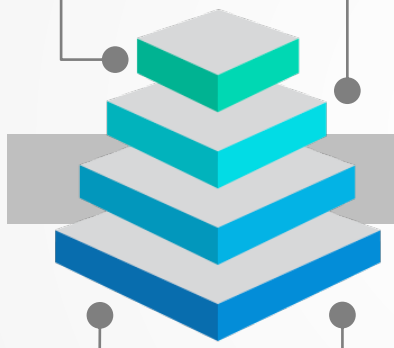
Data Lake [Use-Cases]

Mobile Application
Sending Data Periodically
Storing Data in **TXT** Format

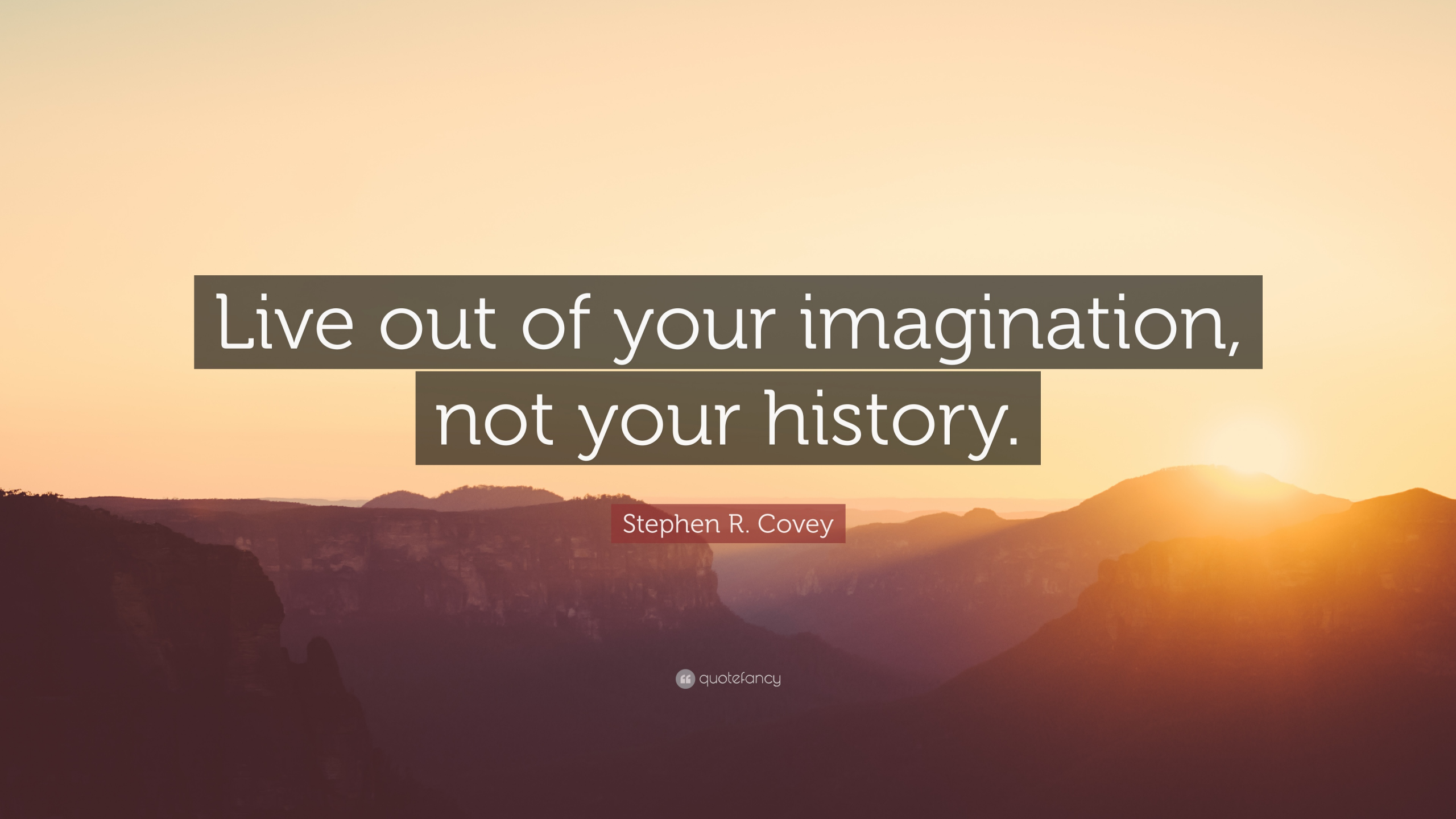
MongoDB
Sending Data in Near Real-Time
Storing Data in **JSON** Format

SQL Server
Sending Data in Batches [2 Hs]
Storing Data in **Parquet** Format

SFTP
Sending Data in Batches [12 Hs]
Storing Data in **CSV** Format



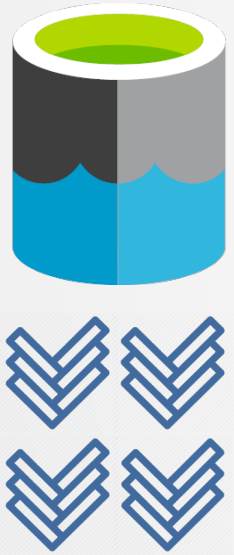
- Data Processing [DP]
- Data Filtering & [ELT]
- Machine Learning [ML]
- Data Warehouse [Dw]
- Data Visualization [DataViz]


Live out of your imagination, not your history.
Stephen R. Covey
quotefancy

The Spark Lifecycle

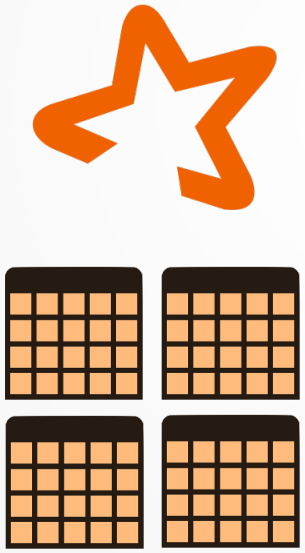
Data Lake

Repository of Raw Data
Without Schema Enforcement



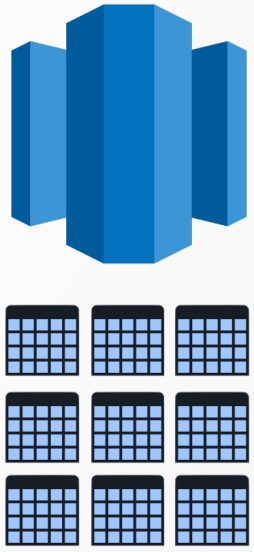
Apache Spark

Distributed Cluster-Computing Framework
Optimized for Memory Computation



Data Warehouse

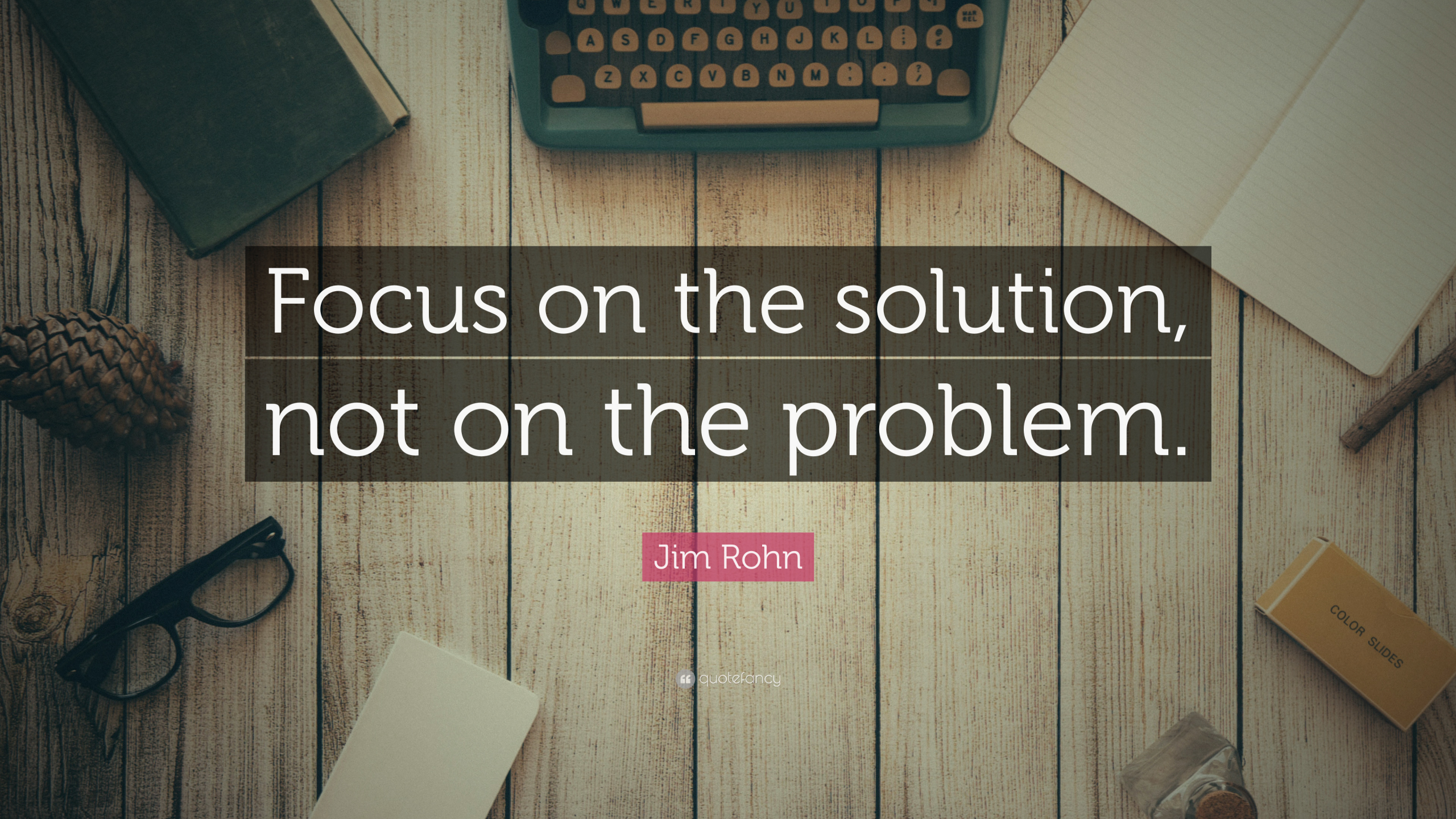
Analytics Platform for Enterprises
Scalability – Horizontally & Vertically



Raw Ingestion

Transformations

Business-Level


Focus on the solution,
not on the problem.
Jim Rohn
quotefancy

Storage Solutions for Spark

Next Evolution of Storage Solutions

Scalability & Performance

the storage solution should be able to scale to the volume of data and provide the read & write throughput and latency that the workload requires.

Support for Diverse Data Formats

the storage solution should be able to store unstructured data (e.g., text files like raw logs), semi-structured data (e.g., JSON data), and structured data (e.g., tabular data).

Openness

supporting a wide range of workloads often requires the data to be stored in open data formats. standard apis allow the data to be accessed from a variety of tools and engines. this allows the business to use the most optimal tools for each type of workload and make the best business decisions.

Transaction Support

complex workloads are often reading and writing data concurrently, so support for acid transactions is essential to ensure the quality of the end results.

Support for Diverse Workloads

the storage solution should be able to support a diverse range of business

- SQL Workloads ~ Traditional BI Analytics
- Batch Workloads ~ Traditional ETL Jobs Processing Unstructured Data
- Streaming Workloads ~ Real-Time Monitoring & Alerting
- ML & AI Workloads ~ Recommendations and Churn Predictions

Storage Solutions for Spark ~ [Database]

Information

- Online Transaction Processing [OLTP]
- Online Analytical Processing [OLAP]

Limitations

- Growth in Data Sizes
- Growth in Diversity Analytics
- Expensive for Scale-Out
- Poor Support for Non-SQL Based Analytics

Online Transaction Processing

Online Analytical Processing

Storage Solutions for Spark ~ [Data Lakes]

Information

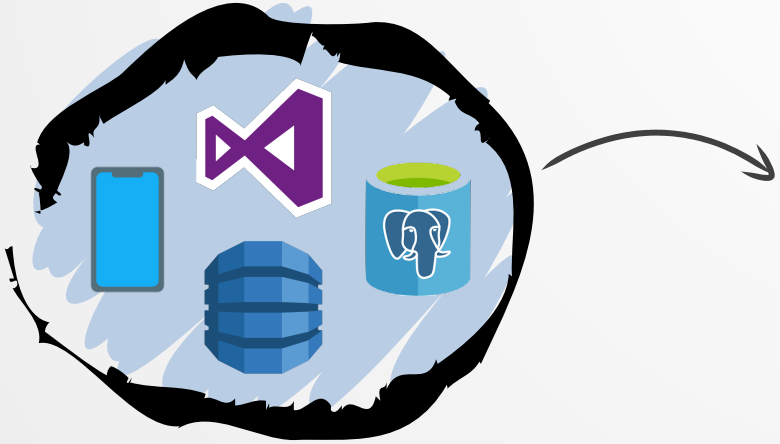
- Support for Diverse Use-Cases
- Support for Diverse File Formats
- Support for Diverse Filesystems

Limitations

- Atomicity & Isolation ~ Corrupted Data
- Consistency – Inconsistency View of Data

External Data

Different Data Sources
Periodically Storing Data

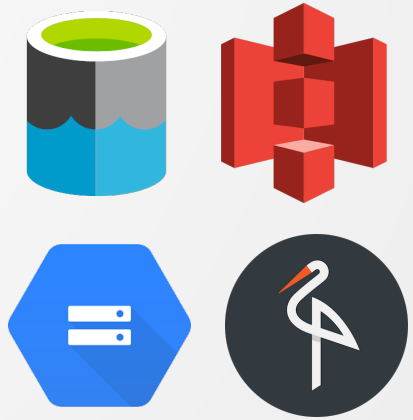


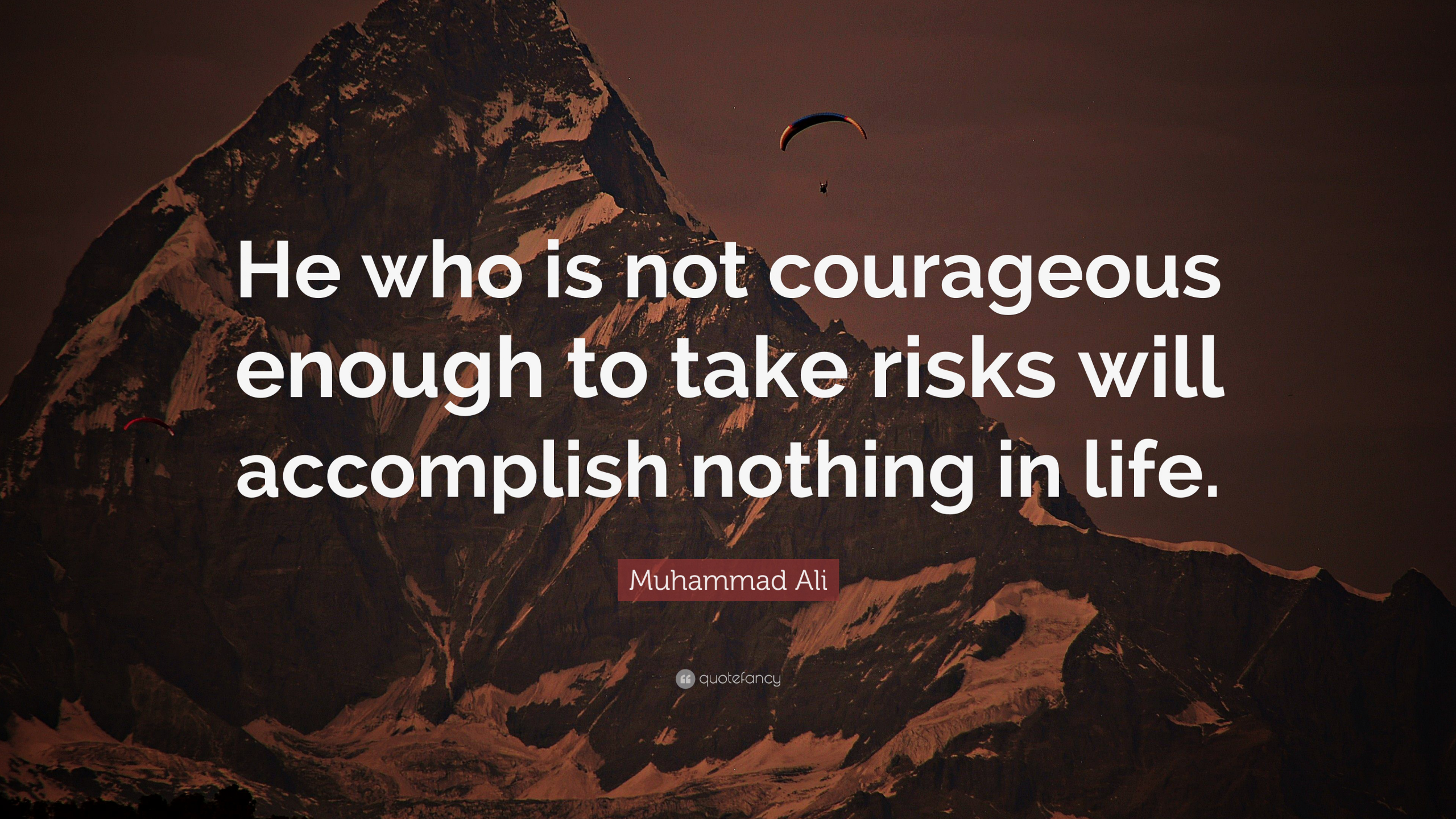
Data Lake

Repository of Raw Data
Without Schema Enforcement

Object Storages

Storages Optimized for Big Data Use-Cases
Scalability & Reliability




He who is not courageous enough to take risks will accomplish nothing in life.
Muhammad Ali
quotefancy

Data Lakehouse

Best of a Data Warehouse and Data Lake

Data Warehouse [Dw] ~ 90s

Decision Support for BI Applications
Since **1980s** evolving & MPP Architecture
Structured Data & No Cost Efficient

OLTP System
RDBMS

ETL
Extract | Transform | Load

Data Mart [DM]
Departmental Business Decision

Data Warehouse [Dw]
Ralph Kimball [Bottom-Up]

Data Lake [DL] ~ 2000s

Large Amount of Data [DV] from Different Sources
Since **2000s** being a Data Raw Repository
Without Transactions, Data Enforcement, Lack of Consistency
Batch & Streaming Jobs not Performing as Expected
AI Advances based on Unstructured Data (Text, Images, Video, Audio)

Data Sources
Applications, RDBMS & NoSQL

ELT
Extract | Load | Transform

Data Lake
Structured, Semi-Structured & Unstructured

Lakehouse [LH] ~ 2020s

Similar Structure of **Dw** with a **Low-Cost Storage** with Features:

- Transaction Support
- Schema Enforcement and Governance
- BI Support
- Decoupled Storage & Computation
- Openness with Structured & Unstructured Data
- End-to-End Streaming

Data Sources
Structured, Semi-Structured & Unstructured Data

ELT
Extract | Load | Transform

Lakehouse
BI, ML, Streaming Analytics & Data Science

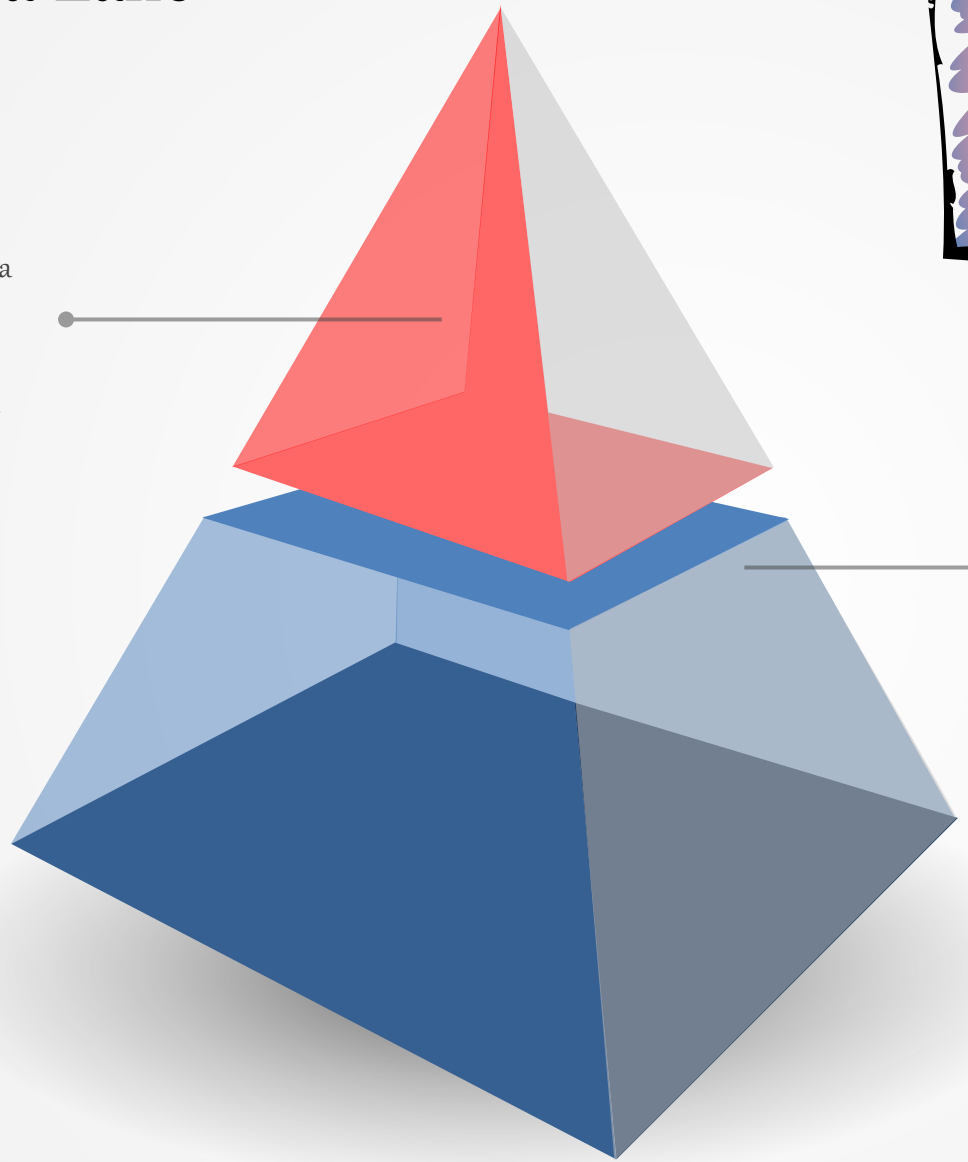
Being deeply loved by someone gives you strength, while loving someone deeply gives you courage.

Lao Tzu

Data Lake vs. Delta Lake

Delta Lake

Repository of **Cleaned** & **Sanitized** Data
ACID Transactions
Schema Enforcement & Evolution
100% Compatible with Apache Spark



Data Lake

Repository of **Raw** Data
Unsiloed Data
Without Schema Enforcement
Data Swamp & Data Quality Issues

Delta Lake [Features]

unified analytics engine for big data

storage layer [format] with acid capabilities

most common data lakes

Delta Lake

Is an Open-Source Storage Layer ~ ACID Transactions to Apache Spark & Big Data Workloads

ACID Transactions

data lakes typically have multiple data pipelines reading and writing data concurrently, and data engineers must go through a tedious process to ensure data integrity, due to the lack of transactions. it provides serializability, the strongest level of isolation level.

Open Format

all data in delta lake is stored in apache parquet format enabling delta lake to leverage the efficient compression and encoding schemes that are native to parquet.

Audit History

delta lake transaction log records details about every change made to data providing a full audit trail of the changes.

Scalable Metadata Handling

delta lake treats metadata just like data, leveraging spark's distributed processing power to handle all its metadata. as a result, delta lake can handle petabyte-scale tables with billions of partitions and files at ease.

Unified Batch & Streaming

a table in delta lake is both a batch table, as well as a streaming source and sink. streaming data ingest, batch historic backfill, and interactive queries all just work out of the box.

Updates & Deletes

delta lake supports Scala, Java, Python, SQL APIs to merge, update and delete datasets. this allows you to easily comply with GDPR and CCPA and simplifies use cases like change data capture.

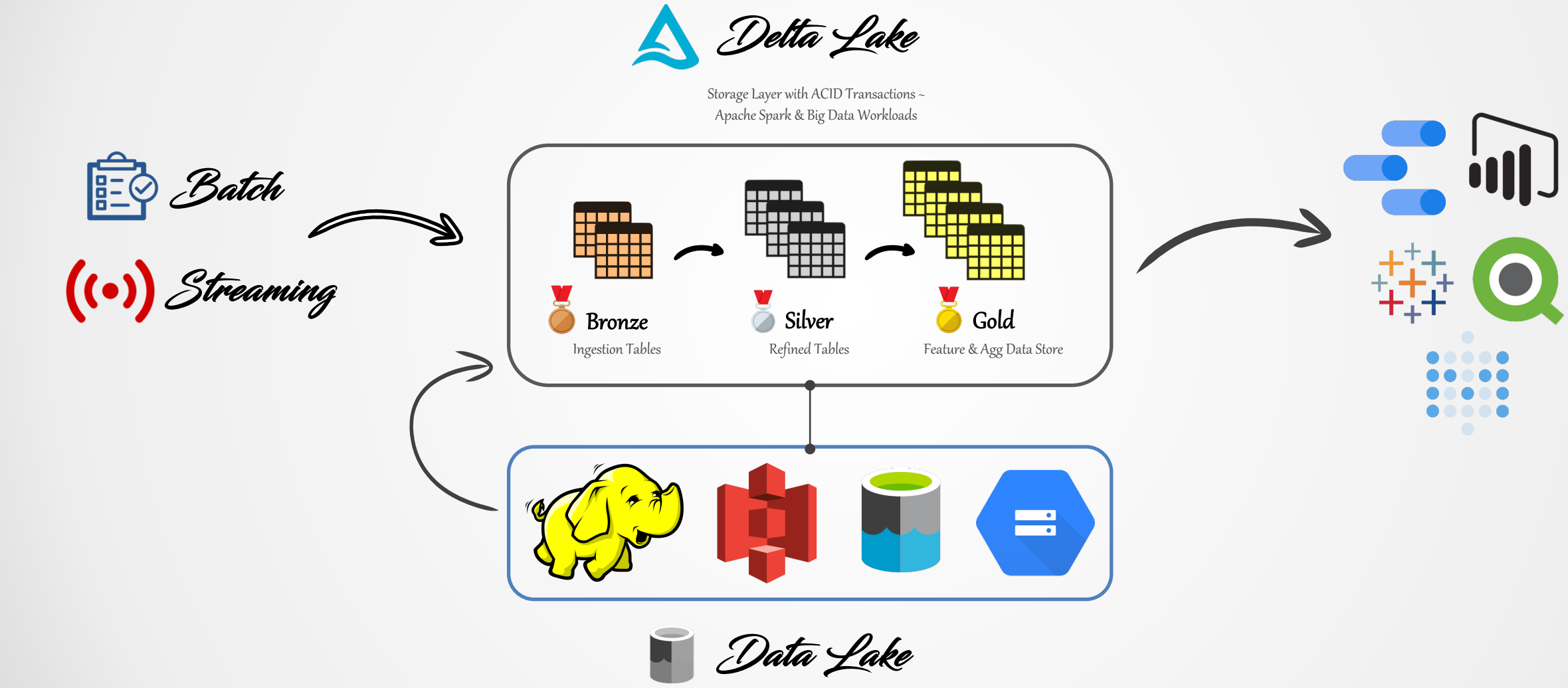
Time Travel [Data Versioning]

delta lake provides snapshots of data enabling developers to access and revert to earlier versions of data for audits, rollbacks or to reproduce experiments.

Schema Enforcement & Evolution

delta lake provides the ability to specify your schema and enforce it. this helps ensure that the data types are correct and required columns are present, preventing bad data from causing data corruption. delta lake enables you to make changes to a table schema that can be applied automatically, without the need for cumbersome dll.

The Delta Architecture



Data Ingestion into Data Lakehouse Best Practices [Delta Lake]

Network of Partners [Connectors]

essential ecosystem of connectors to bring data into **Delta Lake**
Azure Data Factory, Fivetran, Qlik, Infoworks, StreamSets, Syncsoft

Cloud Storage [Auto Loader]

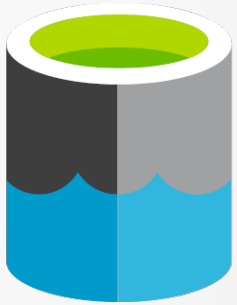
loading data continuously from cloud stores with **Exactly-Once** guarantees at low cost, low latency and minimal DevOps work. List is one of the most expensive operations. **Auto Loader** is an optimized file source that uses **Structured Streaming**

Streaming Load [Structured Streaming]

loading data continuously from **Apache Kafka** with **Exactly-Once** using the Structured Streaming API

Delta Lake

storage system with ACID capabilities that is designated to build a **Data Lakehouse**



Building a Data Lakehouse using Batch-ETL


What you focus on grows.
Esther Hicks
quotefancy

Use Case ~ Near Real-Time ETL

Data Lake

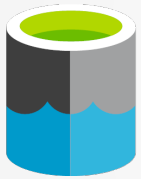
Repository of Raw Data
Without Schema Enforcement

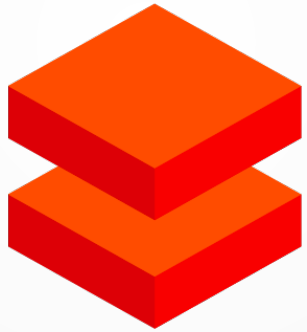
Apache Spark

Distributed Cluster-Computing Framework
Optimized for Memory Computation

Delta Lake

Storage Layer with ACID Transactions ~
Apache Spark & Big Data Workloads





Apache Kafka

Distributed Streaming Platform
Publish & Subscribe, Process & Store Events

ONEWAY

SOLUTION