IT 775
Database Technology

Data Stores

Data Lakes

What is a Data Lake?

Definition:

A Data Lake is a system or storage repository that can accomodate large amounts of structured, semi-structured, and unstructured data. It is a place to store every type of data in its native format with no fixed limits on account or file sizes. It offers high data quantity to increase analytic performance and native integration.

What is a Data Lake?

Definition:

It is a cost-effective way to store all of an organization’s data for later processing. Unlike a hierarchal Data Warehouse where data is stored in Files and Folders, it has a flat architecture. Every data element is given a unique identifier and tagged with a set of metadata information. A Data Lake is usually a single store of data including -

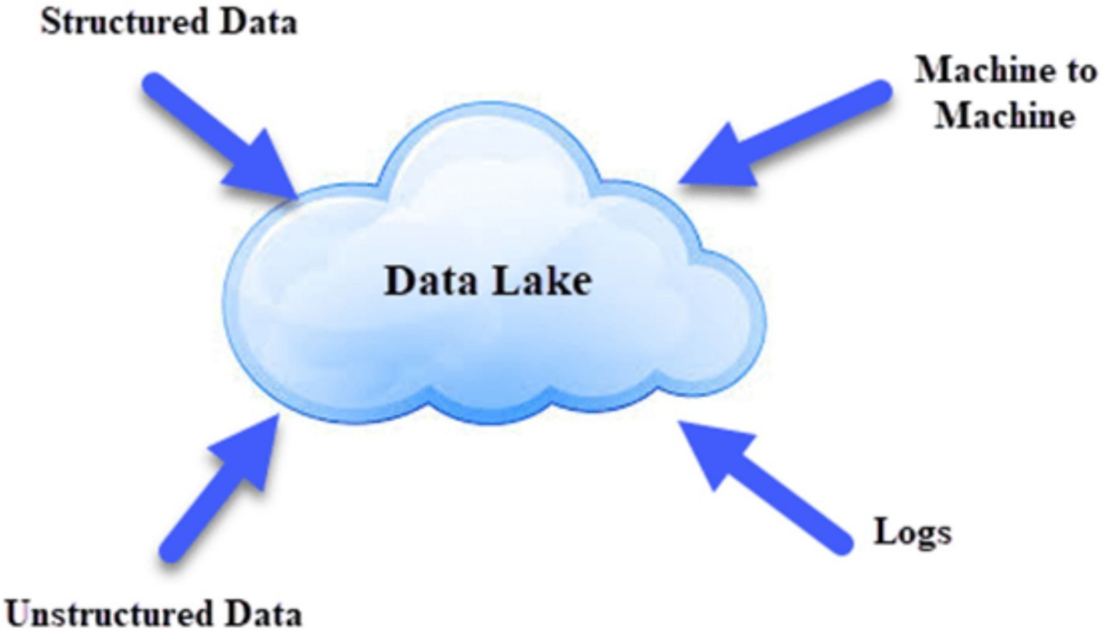
- Raw copies of source system data, sensor data, social data, etc.
- Transformed data used for tasks such as reporting, visualization, advanced analytics and machine learning.

A data lake can include structured data from relational databases (rows and columns), semi-structured data (CSV, logs, XML, JSON), unstructured data (emails, documents, PDFs) and binary data (images, audio, video).

What is a Data Lake?

Definition:

A Data Lake is like a large container which is very similar to a real lake and rivers. Just like in a lake you have multiple tributaries coming in, a data lake has structured data, unstructured data, machine to machine, logs flowing through in real-time.



Why use a Data Lake?

The main objective of building a data lake is to offer an unrefined view of data to data scientists. Reasons for using Data Lake are:

- With the onset of storage engines, like Hadoop, storing disparate information has become easier. There is no need to model data into an enterprise-wide schema with a Data Lake.
- With the increase in data volume, data quality, and metadata, the quality of analyses also increases.
- Data Lake offers business Agility
- Machine Learning and Artificial Intelligence can be used to make predictions.
- It potentially offers a competitive advantage to the implementing organization.
- There is no data silo structure. Data Lake gives 360 degrees view of customers and makes analysis more robust.

Data Lake Architecture

There are three main architectural principles that distinguish data lakes from conventional data repositories:

1) All Data Accepted: No data needs to be turned away. Everything collected from source systems can be loaded and retained in a data lake if desired.

2) Original Data State Preserved: Data can be stored in an untransformed or nearly untransformed state, as it was received from the source system.

3) Transformation of Data As Needed: Data is transformed and fit into a schema as needed based on specific analytics requirements, an approach known as schema-on-read.

Data Lake Architecture

Whatever technology is used in a data lake deployment, some other elements should also be included to ensure that the data lake is functional and that the data it contains doesn't go to waste. That includes the following:

1) A common folder structure with naming conventions.

2) A searchable data catalog to help users find and understand data.

3) A data classification taxonomy to identify sensitive data, with information such as data type, content, usage scenarios and groups of possible users.

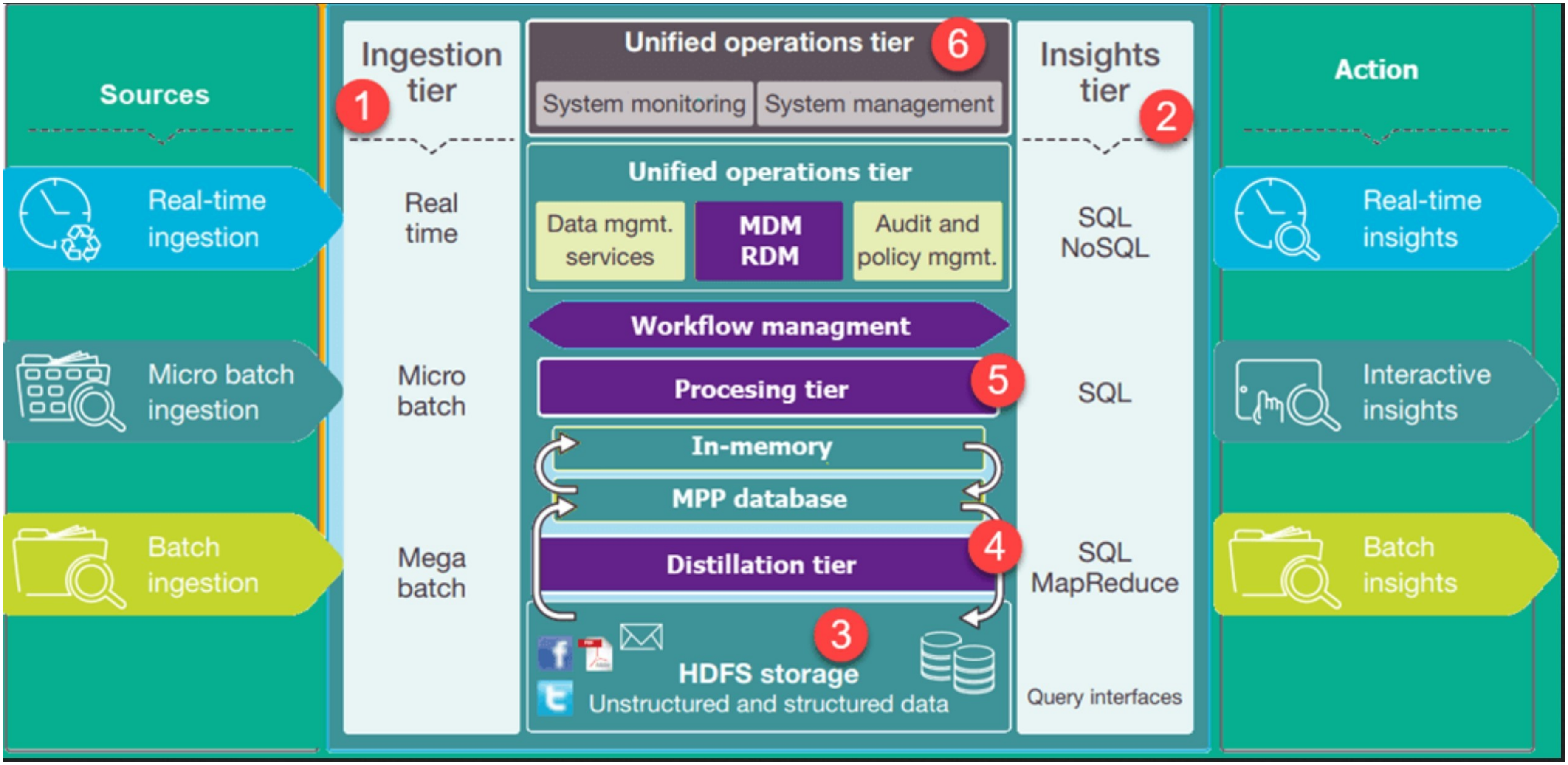
4) Data profiling tools to provide insights for classifying data and identifying data quality issues.

5) A standardized data access process to help control and keep track of who is accessing data.

6) Data protections, such as data masking, data encryption and automated usage monitoring.

Data Lake Architecture

The lower levels represent data that is mostly at rest while the upper levels show real-time transactional data. This data flows through the system with no or little latency.



Data Lake Architecture

The following are important tiers in Data Lake Architecture:

1) Ingestion Tier: The tiers on the left side depict the data sources. The data could be loaded into the data lake in batches or in real-time.

2) Insights Tier: The tiers on the right represent the research side where insights from the system are used. SQL, NoSQL queries, or even excel could be used for data analysis.

3) HDFS is a cost-effective solution for both structured and unstructured data. It is a landing zone for all data that is at rest in the system.

4) Distillation Tier: takes data from the storage tire and converts it to structured data for easier analysis.

5) Processing Tier: run analytical algorithms and users queries with varying real time, interactive, batch to generate structured data for easier analysis.

6) Unified Operations Tier: governs system management and monitoring. It includes auditing and proficiency management, data management, workflow management.

Data Lake Key Concepts

Following are Key Data Lake concepts that one needs to understand to completely understand the Data Lake Architecture:

Data Ingestion: Ingestion allows connectors to get data from a different data source(s) and load into the Data lake.

- All types of Structured, Semi-Structured, and Unstructured data.
- Multiple ingestions like Batch, Real-Time, One-time load.
- Many types of data sources like Databases, Webservers, Emails, IoT, and FTP.

Data Storage: Storage should be scalable, offers cost-effective storage and allow fast access to data exploration. It should support various data formats.

Data Governance: Governance is a process of managing availability, usability, security, and integrity of data used in an organization.

Data Lake Key Concepts

Continued:

Security: Security needs to be implemented in every layer of the Data lake. It starts with Storage, Unearthing, and Consumption. The basic need is to stop access for unauthorized users. It should support different tools to access data with easy to navigate GUI and Dashboards. Authentication, Accounting, Authorization and Data Protection are some important features of data lake security.

Data Quality: Quality is an essential component of Data Lake architecture. Data is used to exact business value. Extracting insights from poor quality data will lead to poor quality insights.

Data Discovery: Discovery is another important stage before you can begin preparing data or analysis. In this stage, tagging technique is used to express the data understanding, by organizing and interpreting the data ingested in the Data lake.

Data Lake Key Concepts

Continued:

Data Auditing: Two major Data auditing tasks are tracking changes to the key dataset.Tracking changes to important dataset elements captures how/ when/ and who changes to these elements. Auditing helps to evaluate risk and compliance.

Data Lineage: This component deals with the data’s origins. It mainly deals with where it moves over time and what happens to it. It eases error corrections in a data analytics process from origin to destination.

Data Exploration: It is the beginning stage of data analysis. It helps to identify the right dataset which is vital for starting the analysis process.

All given components need to work together to play an important part in Data lake building easily evolve and explore the environment.

Data Lake Stages

Continued:

Data Auditing: Two major Data auditing tasks are tracking changes to the key dataset.Tracking changes to important dataset elements captures how/ when/ and who changes to these elements. Auditing helps to evaluate risk and compliance.

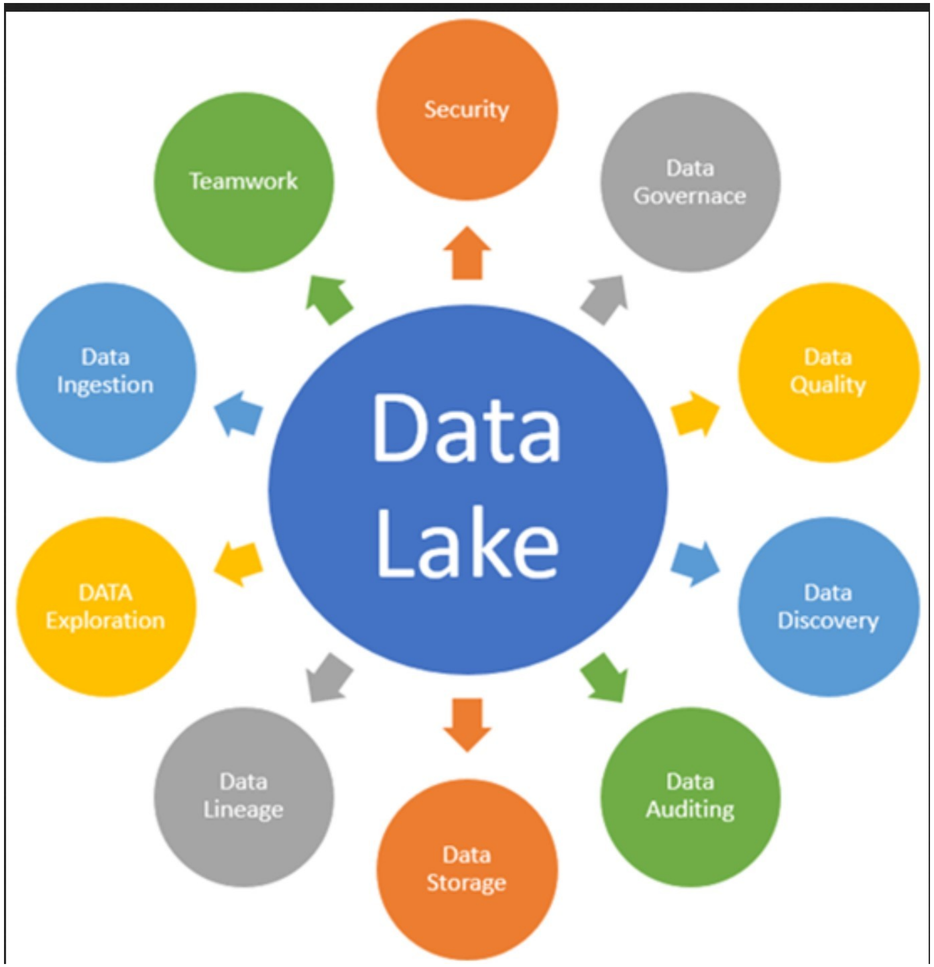
Data Lineage: This component deals with the data's origins. It mainly deals with where it moves over time and what happens to it. It eases error corrections in a data analytics process from origin to destination.

Data Exploration: It is the beginning stage of data analysis. It helps to identify the right dataset which is vital for starting the analysis process.

All given components need to work together to play an important part in Data lake building easily evolve and explore the environment.

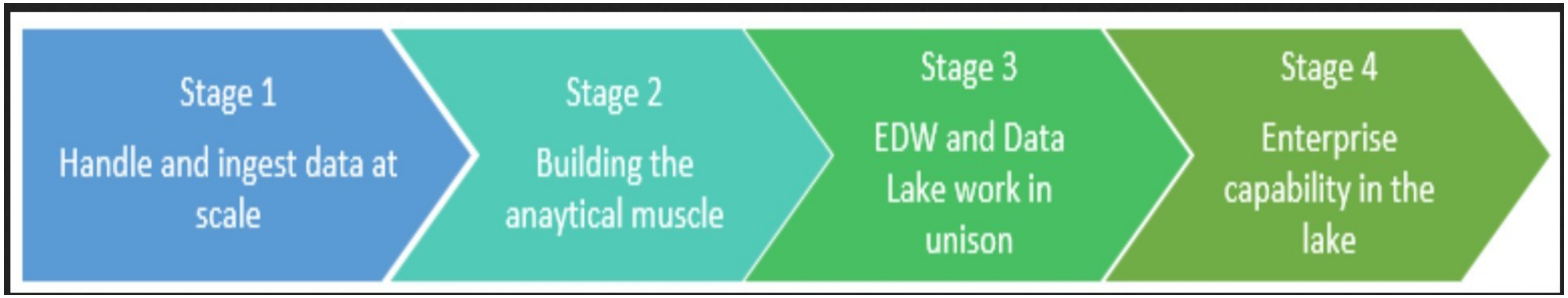
Data Lake Stages

All given components need to work together to play an important part in Data lake building easily evolve and explore the environment.



Data Lake Stages

The Definition of Data Lake Stages differs among various sources, but the basic concepts remain the same. The following is a basic stages definition:



Stage 1

Handle and ingest data at scale

Stage 2

Building the anaytical muscle

Stage 3

EDW and Data Lake work in unison

Stage 4

Enterprise capability in the lake

Data Lake Stages

Stage 1: Handle and ingest data at scale -

This first stage of Data Maturity Involves improving the ability to transform and analyze data. Here, business owners need to find the tools according to their skillset for obtaining more data and build analytical applications.

Data Lake Stages

Stage 2: Building the analytical muscle -

This is a second stage which involves improving the ability to transform and analyze data. In this stage, companies use the tool which is most appropriate to their skillset. They start acquiring more data and building applications. Here, capabilities of the enterprise data warehouse and data lake are used together.

Data Lake Stages

Stage 3: EDW and Data Lake work in unison -

This step involves getting data and analytics into the hands of as many people as possible. In this stage, the data lake and the enterprise data warehouse start to work in a union. Both playing their part in analytics.

Data Lake Stages

Stage 4: Enterprise capability in the lake -

In this maturity stage of the data lake, enterprise capabilities are added to the Data Lake. Adoption of information governance, information lifecycle management capabilities, and Metadata management. However, very few organizations can reach this level of maturity, but this tally will increase in the future.

Data Lake - Best Practices

Best practices for Data Lake Implementation:

- Architectural components, their interaction and identified products should support native data types.
- Design should be driven by what is available instead of what is required. The schema and data requirement is not defined until it is queried.
- Design should be guided by disposable components integrated with APIs.
- Data discovery, ingestion, storage, administration, quality, transformation, and visualization should be managed independently.
- The architecture should be tailored to a specific industry. It should ensure that capabilities necessary for that domain are inherent.
- Fast on-boarding of newly discovered data sources is important.
- Helps customized management to extract maximum value
- Should support existing enterprise data management techniques and methods

Data Lake - Challenges

Challenges for Data Lake Implementation:

- In Data Lakes, data volume is higher, so the process must be more reliant on programmatic administration.
- It is difficult to deal with sparse, incomplete, volatile data.
- Wider scope of dataset and source needs imply a larger data governance & support need.

Data Lake - Benefits

Benefits:

- Helps fully with product ionizing & advanced analytics
- Offers cost-effective scalability and flexibility
- Offers value from unlimited data types
- Reduces long-term cost of ownership
- Allows economic storage of files
- Quickly and adaptable to changes
- Centralization of different content sources
- Users, from various departments, may be scattered around the globe can have flexible access to the data.

Data Lake - Risks

Risks:

- After some time, Data Lake may lose relevance and momentum.
- There is larger amount risk involved while designing Data Lake
- Unstructured Data may lead to Ungoverned Chaos, Unusable Data, Disparate & Complex Tools, negatively impacting Enterprise-Wide Collaboration, Unified analysis, Data Consistency, and Commonality of usage
  - **Data Swamps** - Preventing a data lake from turning into a data swamp is a major challenge. If it isn't set up and managed properly, it can become a messy dumping ground for data. Users may not find what they need and data managers may lose track of data, even as more pours in.
- Increased storage & compute costs
- There is no way to get insights from others who have worked with the data because there is no account of the lineage of findings by previous analysts.
- Security and access control. Sometimes data can be placed into a lake without any oversight, as some of the data may have privacy and regulatory needs.

Lakes vs Warehouses

Parameters	Data Lakes	Data Warehouse
Data	Store everything.	Only on Business Processes.
Processing	Data are mainly unprocessed	Highly processed data.
Type of Data	Un, semi-, and fully -structured	Mostly in tabular form
Task	Share data stewardship	Optimized for data retrieval
Agility	Highly agile, reconfigurable	Less agile and fixed configuration
Users	Mostly Data Scientist	Business professionals
Storage	Designed for low-cost storage	Expensive for fast response times
Security	Offers lesser control.	Allows better control of the data.
EDW Replacement	Can be EDW source	Complementary to EDW
Schema	On read (no predefined)	On write (predefined)
Data Processing	Fast ingestion of new data.	Time-consuming for new content
Data Granularity	Low level	Summary or Aggregated
Tools	Open source/tools like Hadoop	Mostly commercial tools.

Lakes vs Warehouses

The biggest distinctions between data lakes and data warehouses are their support for data types and their approach to schema. In a data warehouse that primarily stores structured data, the schema for data sets is predetermined, and there's a plan for processing, transforming and using the data when it's loaded into the warehouse. That's not necessarily the case in a data lake. It can house different types of data and doesn't need to have a defined schema for them or a specific plan for how the data will be used.

To illustrate the differences between the two platforms, think of an actual warehouse versus a lake. A lake is liquid, shifting, amorphous and fed by rivers, streams and other unfiltered water sources. Conversely, a warehouse is a structure with shelves, aisles and designated places to store the items it contains, which are purposefully sourced for specific uses.

Because of their differences, many organizations use both a data warehouse and a data lake, often in a hybrid deployment that integrates the two platforms. Frequently, data lakes are an addition to an organization's data architecture and enterprise data management strategy instead of replacing a data warehouse.

Lakes vs Warehouses

This conceptual difference shows up in several ways, including:

Technology platforms: A data warehouse architecture usually includes a relational database running on a conventional server, whereas a data lake is typically deployed in a Hadoop cluster or other big data environment.

Data sources: The data stored in a warehouse is primarily extracted from internal transaction processing applications to support basic business intelligence (BI) and reporting queries, which are often run in associated data marts created for specific departments and business units. Data lakes typically store a combination of data from business applications and other internal and external sources, such as websites, IoT devices, social media and mobile apps.

Users: Data warehouses are useful for analyzing curated data from operational systems through queries written by a BI team or business analysts and other self-service BI users. Because the data in a data lake is often uncurated and can originate from various sources, it's generally not a good fit for the average BI user. Instead, data lakes are better suited for use by data scientists who have the skills to sort through the data and extract meaning from it.

Lakes vs Warehouses

This conceptual difference shows up in several ways, including:

Data quality: The data in a data warehouse is generally trusted as a single source of truth because it has been consolidated, preprocessed and cleansed to find and fix errors. The data in a data lake is less reliable because it's often pulled in from different sources as is and left in its raw state without first being checked for accuracy and consistency.

Agility and scalability: Data lakes are highly agile platforms: Because they use commodity hardware, most can be reconfigured and expanded as needed to meet changing data requirements and business needs. Data warehouses are less flexible because of their rigid schema and prepared data sets.

Security: Data warehouses have more mature security protections because they have existed for longer and are usually based on mainstream technologies that likewise have been around for decades. But data lake security methods are improving, and various security frameworks and tools are now available for big data environments.

Lakehouses

Data lakes are hard to properly secure and govern due to the lack of visibility and ability to delete or update data. These limitations make it very difficult to meet the requirements of many regulatory bodies.

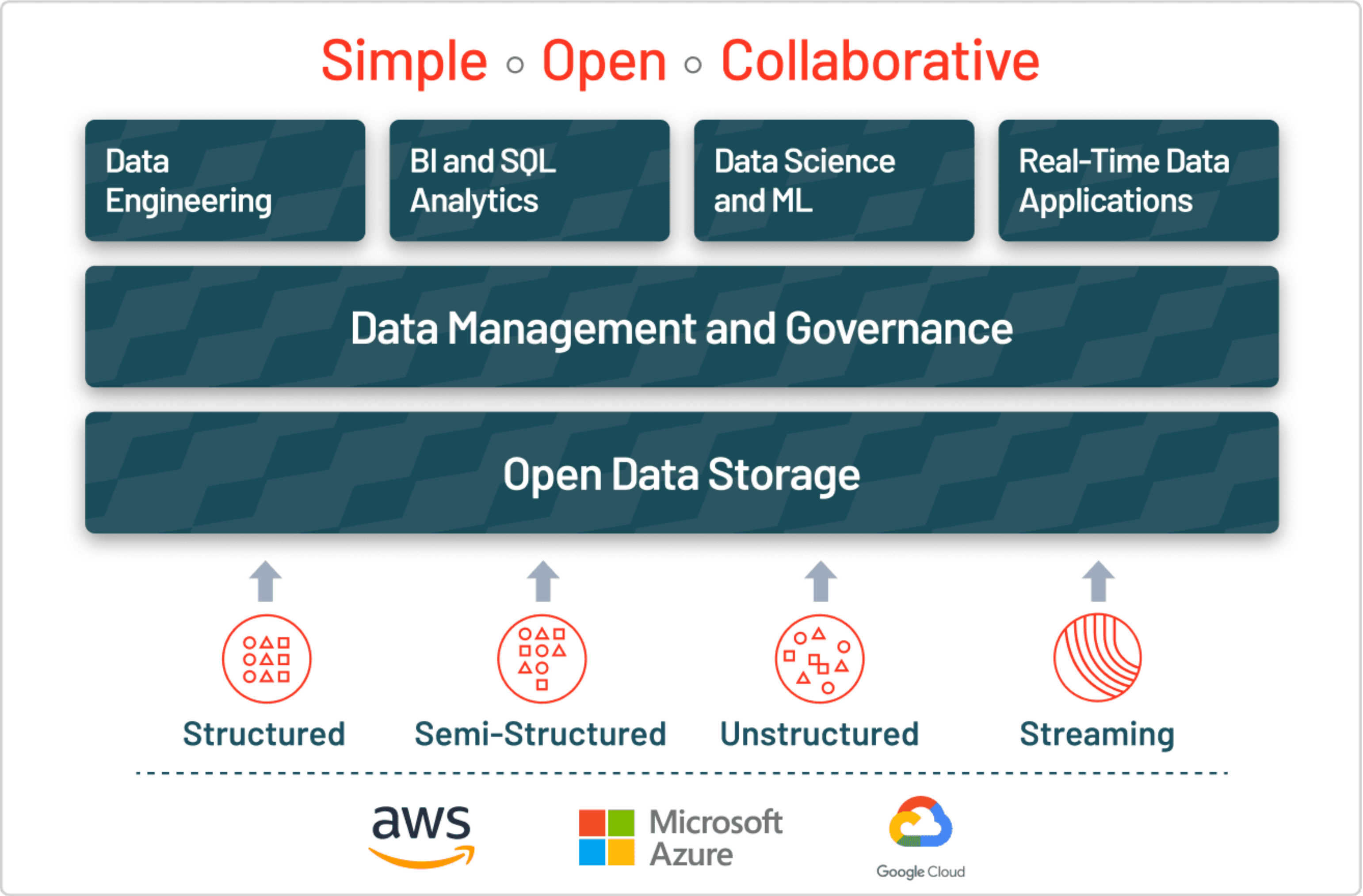
For these reasons, a traditional data lake on its own is not sufficient to meet the needs of businesses looking to innovate, which is why businesses often operate in complex architectures, with data siloed away in different storage systems: data warehouses, databases and other storage systems across the enterprise. Simplifying that architecture by unifying all your data in a data lake is the first step for companies that aspire to harness the power of machine learning and data analytics to win in the next decade.

Lakehouses

The answer to the challenges of data lakes is the lakehouse, which solves the challenges of a data lake by adding a transactional storage layer on top. A lakehouse that uses similar data structures and data management features as those in a data warehouse but instead runs them directly on cloud data lakes. Ultimately, a lakehouse allows traditional analytics, data science, and machine learning to coexist in the same system, all in an open format.

A lakehouse enables a wide range of new use cases for cross-functional enterprise-scale analytics, BI, and machine learning projects that can unlock massive business value. Data analysts can harvest rich insights by querying the data lake using SQL, data scientists can join and enrich data sets to generate ML models with ever greater accuracy, data engineers can build automated ETL pipelines, and business intelligence analysts can create visual dashboards and reporting tools faster and easier than before. These use cases can all be performed on the data lake simultaneously, without lifting and shifting the data, even while new data is streaming in.

Lakehouses



Simple ◦ Open ◦ Collaborative

Data Engineering

BI and SQL Analytics

Data Science and ML

Real-Time Data Applications

Data Management and Governance

Open Data Storage

Structured

Semi-Structured

Unstructured

Streaming

aws

Microsoft Azure

Google Cloud

Acknowledgement & Sources

Sources:

- https://www.guru99.com/data-lake-architecture.html
- https://en.wikipedia.org/wiki/Data_lake
- https://searchdatamanagement.techtarget.com/definition/data-lake
- https://databricks.com/discover/data-lakes/introduction