Modern Data Architectures

Evolution of Data Warehouse -> Data Lake -> Data Lakehouse -> Data Mesh

Data Warehouse:

A **centralized repository optimized for storing & processing structured data using SQL**, typically used for reporting and analysis

Data Lake:

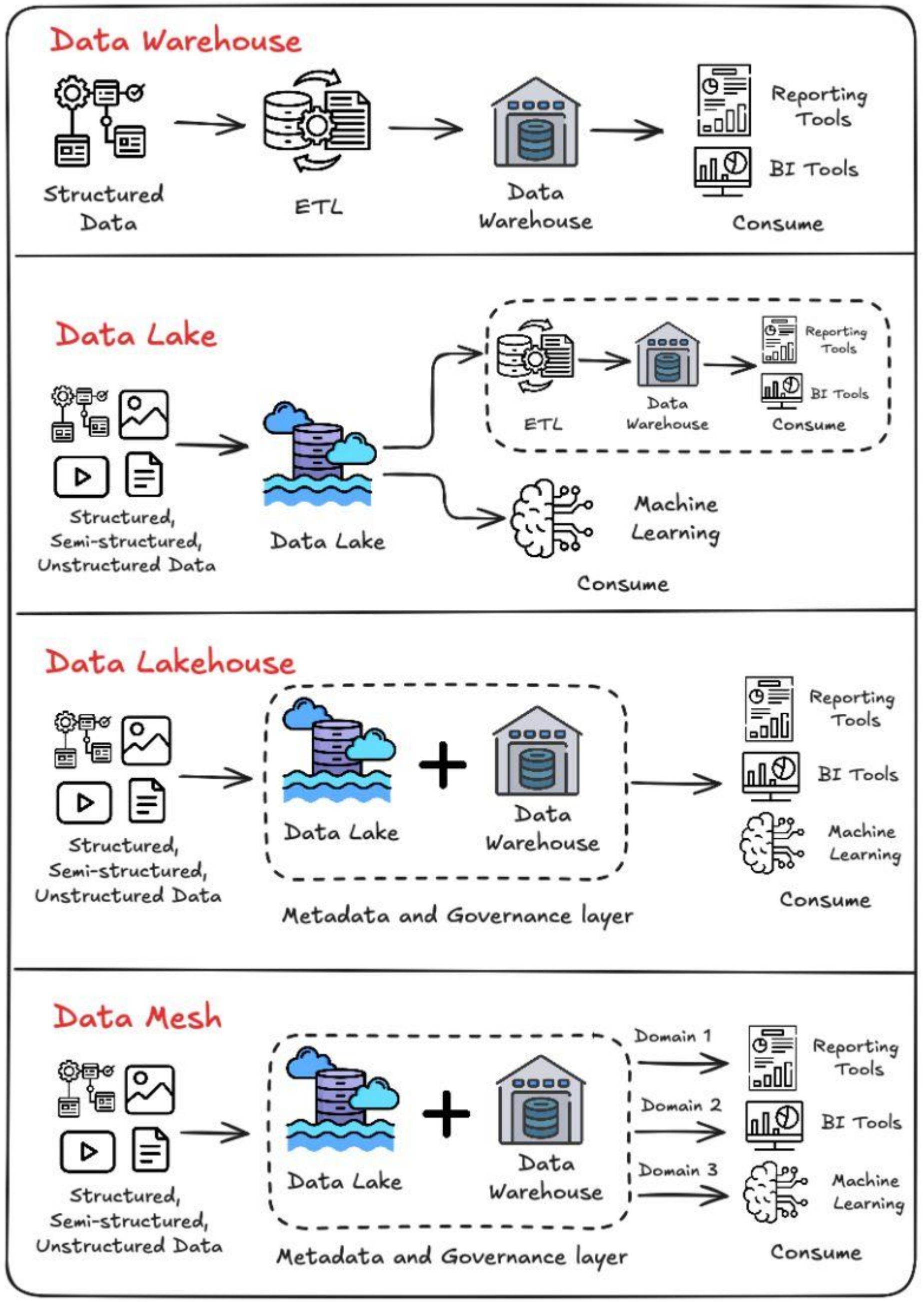
A vast **storage repository that can hold a large amount of raw data** in its native format, whether structured, semi-structured, or unstructured

Data Lakehouse:

A modern data architecture that combines the best features of data warehouses and data lakes

Data Mesh:

Data Mesh is a decentralized data architecture that treats data as a product, with domain-oriented teams owning, managing, and serving their data.



Data Warehouse (DW)

A **Data Warehouse (DW)** is a **centralized repository** designed to store, integrate, and manage large volumes of data from multiple sources for **business intelligence (BI), analytics, and decision-making**. It is **optimized for analytical queries (OLAP)**, not for day-to-day transactions (OLTP).

Key Characteristics

- **Subject-Oriented** – Organized around key business areas (sales, finance,HR).
- **Integrated** – Combines data from different sources.
- **Time-Variant** – Stores historical data for trend analysis.
- **Non-Volatile** – Data is stable once loaded (not frequently updated like in OLTP)
- **Supports ETL/ELT** – Data is extracted, transformed/cleaned, and loaded.

Benefits

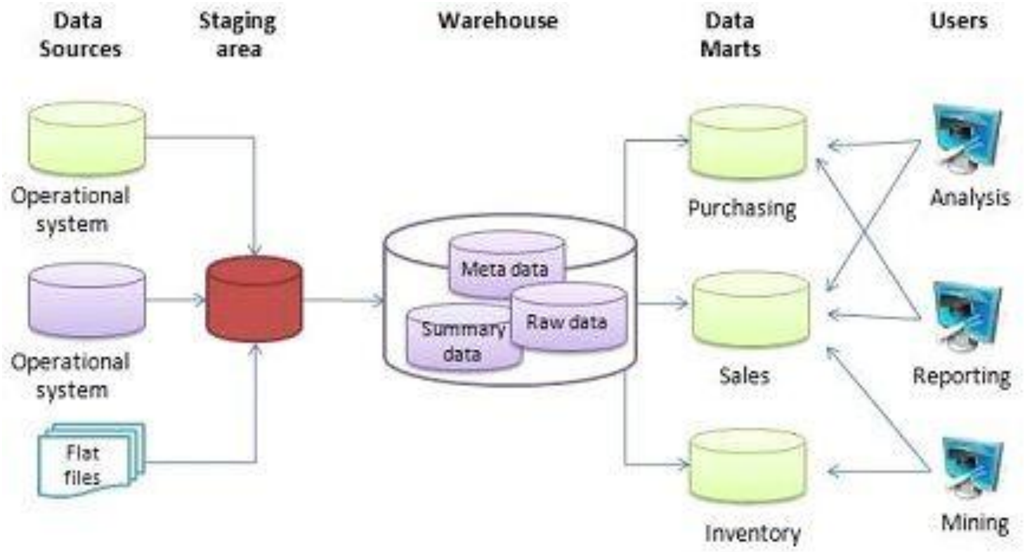
- Provides a single source of truth for analytics.
- Enables faster decision-making using dashboards & reports.
- Supports trend analysis, forecasting, and BI.
- Handles large-scale queries efficiently

On-Premise Data Warehouses (Legacy)

- Oracle Exadata
- Teradata
- IBM Db2 Warehouse
- Microsoft SQL Server Data Warehouse

Cloud Data Warehouse

- Google BigQuery (GCP)
- **Amazon Redshift** (AWS)
- **Azure Synapse Analytics** (Azure)
- **Snowflake** (multi-cloud)



Data Lake

A Data Lake is a **centralized repository that stores raw data in its native format** (structured, semi-structured, and unstructured) at any scale.

It allows organizations to store all their data without first needing to structure it.

Unlike a data warehouse (which requires cleaning and structuring before loading), a data lake follows the **ELT approach** – data is loaded first (raw) and structured later as needed.

Key Characteristics

- **Stores all data types:** Structured, Semi-Structured and Unstructured.
- **Schema-on-read:** Schema is applied when you read the data, not before storing.
- **Cost-effective & scalable:** Built on cheap storage systems (cloud object storage).
- **Real-time & batch ingestion supported.**

Benefits

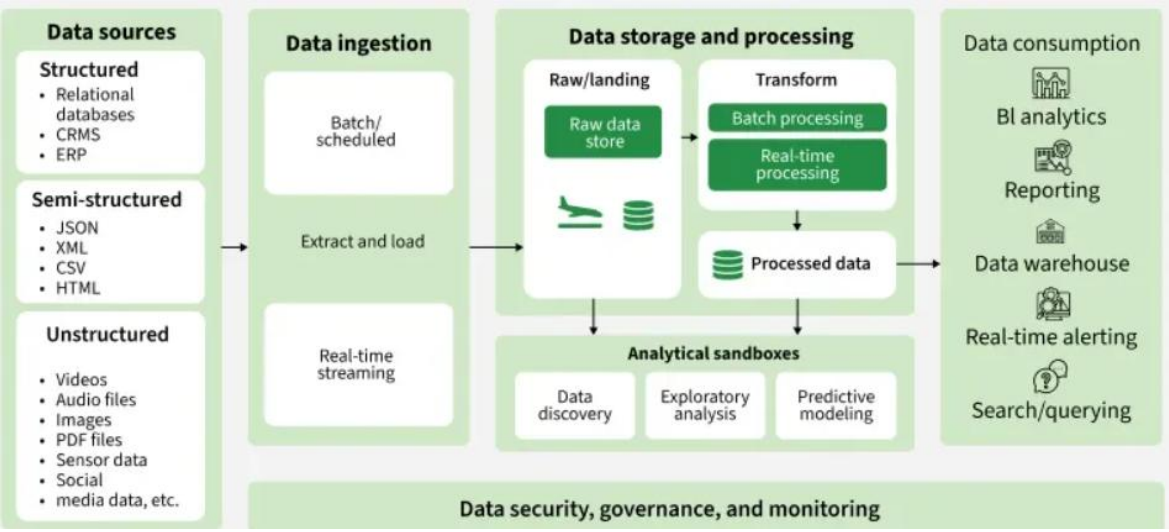
- Handles massive data volumes at low cost.
- Eliminates the need to decide data structure upfront.
- Enables data exploration and advanced analytics (AI/ML).
- Works as a single source of truth for raw data.

On-Prem Data Lake

- Hadoop Distributed File System(HDFS)
- Cloudera Data Platform (CDP)
- MapR Data Platform

Cloud Data Lake Platform

- Amazon S3 (AWS)
- Azure Data Lake Storage (ADLS)
- Google Cloud Storage (GCS)



Data Lakehouse

A Lakehouse is a modern data architecture that combines the low-cost, flexible storage of a Data Lake with the structured, performance-optimized features of a Data Warehouse.

- A Lakehouse merges both, allowing organizations to store all types of data and also run high-performance analytics, BI, and AI/ML on the same platform
- Data Lakes store raw, unstructured, and semi-structured data (cheap, scalable).
- Data Warehouses store clean, structured data (fast for analytics).

Key Features of Lakehouse

- **Unified storage** for raw and curated data.
- **ACID transactions** (ensures data consistency).
- **Schema enforcement and evolution.**
- **Time travel & versioning** (access past states of data).
- **Support for batch, streaming, BI, and ML workloads.**
- **Decoupled storage and compute** (scale independently).

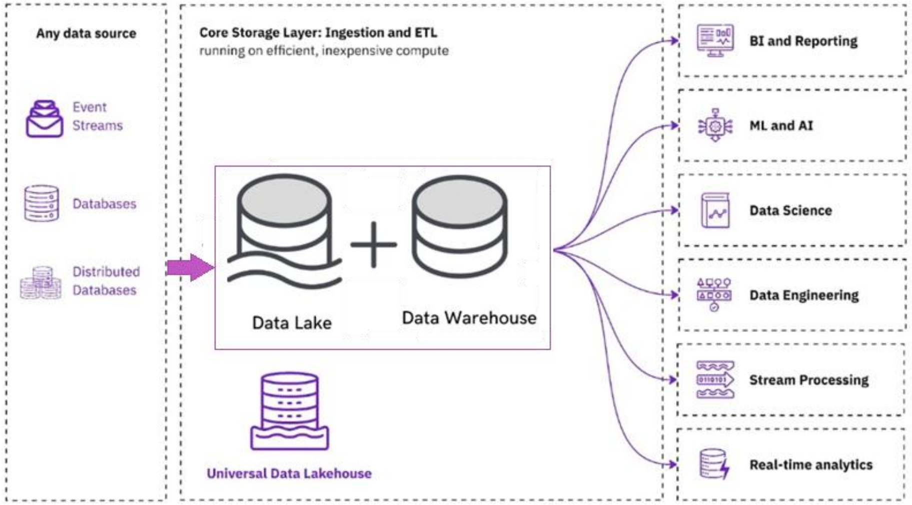
On-Prem Data Lake

- CDP - Supports Iceberg & Hudi.
- Apache Hadoop + Spark + Delta Lake (self-managed).
- MapR Data Platform

Cloud Data Lake Platform

- **Google BigLake (GCP)** – Unifies Data Lake + BQ
- **Databricks Lakehouse Platform (Delta Lake)**

  AWS, Azure, GCP.Azure Data Lake Storage
- **Amazon S3 + Athena + Redshift Spectrum** – Lakehouse-style setup.



Data Mesh

A Data Mesh is a modern decentralized data architecture that shifts data ownership from a central IT/data engineering team to individual business domains (e.g., sales, marketing, finance).

Instead of treating data as a by-product, Data Mesh treats data as a product—each domain manages, owns, and serves its data, making it available for others through standardized interfaces.

Key Principles of Data Mesh

1. Domain-Oriented Ownership

- Each domain (e.g., sales, HR) owns its data pipelines and governance.

2. Data as a Product

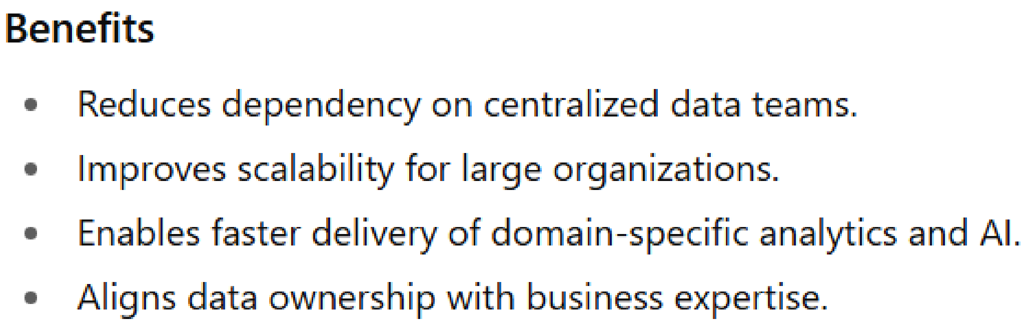
- Data must be discoverable, reliable, and usable (with clear SLAs).

3. Self-Serve Data Infrastructure

- Teams use standardized tools to ingest, transform, and share data without central bottlenecks.

4. Federated Computational Governance

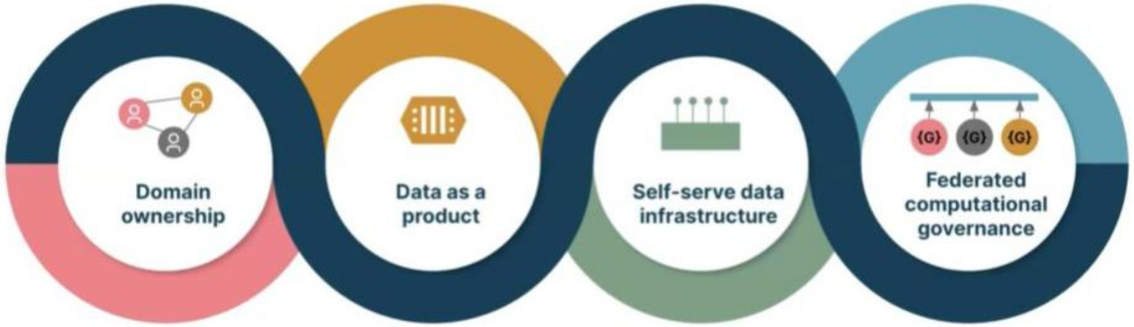
- Governance (security, privacy, compliance) is enforce in a distributed but standardized way.

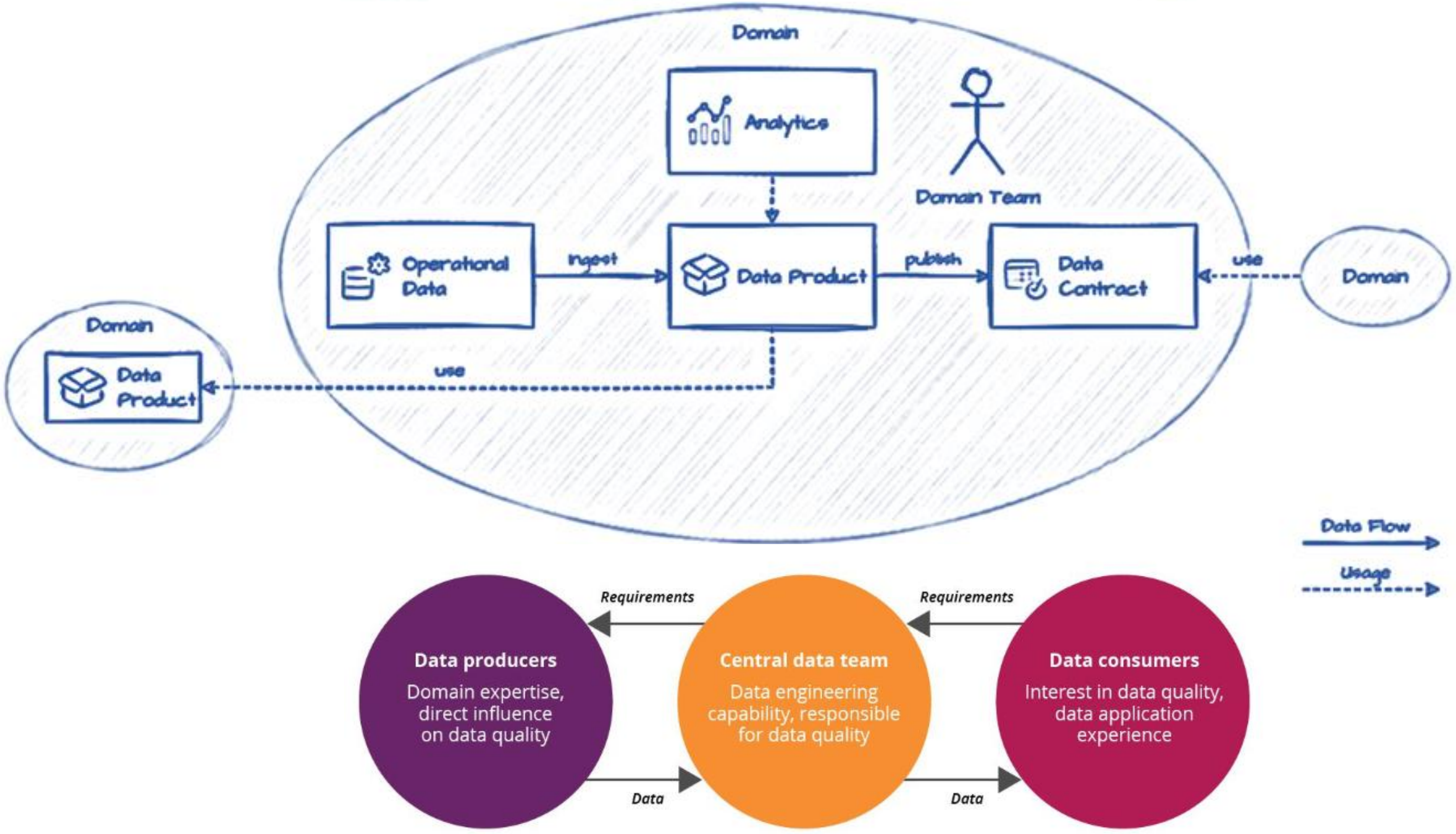


Benefits

- Reduces dependency on centralized data teams.
- Improves scalability for large organizations.
- Enables faster delivery of domain-specific analytics and AI.
- Aligns data ownership with business expertise.

Data Mesh


Domain ownership
Data as a product
Self-serve data infrastructure
{G} {G} {G}
Federated computational governance



Cloud Data Mesh

- Databricks Lakehouse (with Unity Catalog).
- Snowflake (with Snowgrid for multi-domain sharing).
- Google BigQuery with Dataplex.
- AWS S3 + Lake Formation + Glue.
- Azure Synapse + Microsoft Fabric + OneLake.