The Data Lakehouse:
A Modern Data Architecture

Karthick Selvam

May 10, 2025

1 Introduction: The Data Dilemma and the Lakehouse Concept

Today's enterprise faces a significant data dilemma: despite generating vast amounts of data, a large percentage remains unused. According to IDC in 2023, **73% of enterprise data goes unused** for decision-making. This paradox stems from the limitations of traditional data architectures, which forced organizations to make difficult trade-offs.

Historically, organizations had to choose between:

- **Data Warehouses**: Designed for performance on structured data, excellent for traditional Business Intelligence (BI), but rigid and lacked support for modern workloads.
- **Data Lakes**: Offering flexibility and cost-efficiency for handling diverse data types, but notorious for poor data governance and reliability.

This forced choice led to fragmented architectures, data silos, and operational complexity. The Data Lakehouse architecture emerged as a synthesis, designed to **merge the best capabilities of both data lakes and data warehouses** onto a single, unified platform.

While the term ”Data Lakehouse” was originally coined by Databricks, the architecture it represents is now widely adopted. According to Databricks, it's a new open data architecture combining the flexibility, cost-efficiency, and scalability of data lakes with the data management, transaction, and governance capabilities of data warehouses. This fusion enables support for both BI and Machine Learning (ML) workloads on a unified platform, eliminating the need for complex, multi-system architectures.

2 Evolution of Data Architectures: Warehouses and Lakes

To understand the Lakehouse, let's briefly review the architectural evolution that led to its necessity.

2.1 Data Warehouses: The Era of Structured Analytics

Data warehouses first emerged in the early 1980s to provide a centralized repository for organizational data. This allowed businesses to move away from departmental silos and consolidate information for comprehensive decision-making.

Key characteristics of data warehouses included:

- Stored primarily **structured operational data**, with some large warehouses including external data.
- Could also handle **semi-structured formats** like JSON or XML, but **not unstructured data** such as images and videos.

- Data went through a rigorous **ETL (Extract, Transform, Load)** process before loading, ensuring data quality and consistency upfront.
- Often included **data marts** for specific subject areas, containing cleaned, validated, and aggregated data for KPIs.
- Accessed mainly through **Business Intelligence reports**.

By the early 2000s, most large companies relied on data warehouses, which were crucial for business decision-making based on available structured data.

2.1.1 Challenges of Data Warehouses

The exponential growth of data volumes and the emergence of new data types like videos and images due to the internet posed significant challenges for traditional data warehouses:

- **Limited Data Type Support:** Unable to process unstructured data, missing out on insights from a vast and growing data source.
- **Long Development Times:** The ETL process required thorough quality checks and transformations, leading to long delays (often 24-48 hours of latency) to add new data sources or make changes, resulting in stale insights.
- **Proprietary Technology Vendor Lock-in:** Built on traditional relational databases or MPP engines using proprietary file formats, leading to vendor lock-in and limited tool flexibility.
- **Scalability Issues:** Traditional on-premises warehouses were hard to scale.
- **High and Inflexible Storage Costs:** Storage was expensive ($23/TB vs. pennies on the cloud) and coupled with compute, preventing independent scaling and cost optimization.
- **Insufficient AI/ML Support:** Not designed for the iterative, compute-intensive workloads required for modern data science and machine learning.

These limitations highlighted the need for a more flexible and scalable approach.

2.2 Data Lakes: The Promise of Flexibility

Data lakes were introduced around 2011, driven by the need to address the challenges of data warehouses, particularly the inability to handle diverse data types and the high costs.

Key characteristics of data lakes include:

- Designed to handle **all data types**: structured, semi-structured, and crucially, **unstructured data** (which is 90% of modern data).
- Raw data ingested directly with minimal initial processing (**schema-on-read**), allowing quicker ingestion and faster initial solution development.
- Built on **cheap, scalable storage** solutions like HDFS and cloud object stores (e.g., Amazon S3, Azure Data Lake Storage Gen2), with storage costs around $2.30/TB.
- Utilized **open source file formats** like Parquet, ORC, and Avro, promoting flexibility and tool interoperability.
- Provided better access and support for **data science and machine learning** workloads.

Data lakes offered the necessary flexibility and cost benefits for the big data era. However, they came with their own significant drawbacks.

2.2.1 Challenges of Data Lakes: The ”Data Swamp”

While solving some problems, data lakes introduced new complexities and challenges, earning them the moniker ”data swamps”:

- **Lack of ACID Transactions:** This was a major problem. Data lakes did not have built-in support for ACID (Atomicity, Consistency, Isolation, Durability), essential for reliable data management. This led to many issues:

  - Partial loads could leave behind partially written files from failed jobs, requiring manual cleanup.
  - Lack of consistent reads meant users might access inconsistent or partially updated data, compromising reliability for analytics.
  - No direct support for updates or deletes. Correcting data or handling ”right to be forgotten” requests (like GDPR) required complex processes involving partitioning files and rewriting entire large files, which was time-consuming, error-prone, and expensive.
  - No easy way to roll back changes, making recovery from errors difficult.
- **Poor Performance for BI:** Data lakes struggled to provide the fast, interactive query performance required for traditional BI reports, often taking 12-15 seconds per query compared to sub-second in warehouses.
- **Lack of Proper Data Governance:** Essential features like security, access control, data discovery, and audit trails were difficult to implement consistently across the lake. Lack of version control made tracking changes and ensuring governance harder.
- **Operational Complexity:** Setting up and managing reliable data lakes required significant expertise.
- **Complex Lambda Architecture:** Handling both streaming and batch data required separate processing layers, leading to the complex and difficult-to-manage Lambda architecture.

To mitigate the BI performance issue, companies often copied data from the lake back into a warehouse, resulting in fragmented architectures with data silos and duplication.

3 The Data Lakehouse Architecture: The Unified Solution

The Data Lakehouse architecture is the synthesis, designed to effectively support **both BI and Data Science/ML/AI workloads** on a single platform, eliminating the complexity and limitations of previous approaches.

A Lakehouse is essentially a data lake (built on cheap cloud object storage with open formats) enhanced with data management and governance capabilities borrowed from data warehouses. This is achieved by adding a metadata layer providing transactional guarantees and a unified catalog for governance on top of the lake storage.

3.1 Lakehouse Core Components

The Databricks platform, which implements the Lakehouse vision, relies on key components:

Delta Lake Logo
(Please download)

Unity Catalog
(Please download)

3.1.1 Delta Lake: The Transactional Foundation

Delta Lake is an open-source storage layer that brings reliability to the data lake. Its key features include:

- **ACID Transactions:** Ensures data integrity, prevents corruption from failed writes, and provides consistent views for readers.
- **Time Travel (Data Versioning):** Allows access to historical versions of data. This is crucial for audits, rollbacks, and reproducing ML experiments.
- **Schema Enforcement & Evolution:** Enforces schema on writes to prevent bad data entry while allowing schemas to evolve over time without disruptive rewrites.
- **Upserts/Deletes:** Enables efficient updates and deletes directly on the data lake, simplifying compliance and data correction.
- **Optimization Features:** Includes techniques like Z-ordering to optimize data layout, leading to significantly faster query performance (e.g., 94% faster scans for certain queries).

Delta Lake transforms the unreliable data lake into a reliable, transactional database layer.

3.1.2 Unity Catalog: Unified Governance

Unity Catalog is a centralized metadata, governance, and security layer for the Lakehouse. It provides:

- **Single Source of Truth:** A unified view and management plane for data, analytics, and AI assets across clouds and data types.
- **Fine-grained Security:** Offers robust Role-Based Access Control (RBAC), including row-level and column-level security, essential for meeting strict data privacy and compliance requirements (e.g., HIPAA).
- **Automated Data Lineage:** Automatically tracks how data flows and is transformed, providing essential audit trails for compliance (e.g., SOX).
- **Global Data Search & Discovery:** Makes it easy for users to find relevant data assets quickly (e.g., $\sim$200ms metadata retrieval across millions of assets).
- **Audit Trails:** Logs access to data for monitoring and compliance.

Unity Catalog addresses the governance failures of raw data lakes, enabling secure and compliant data sharing and access.

Together, Delta Lake provides the reliable storage layer, and Unity Catalog provides the unified governance layer on top of the cost-effective cloud object storage. This combination eliminates the need for complex Lambda architectures by allowing streaming and batch data to be processed seamlessly on the same tables. Data from the Lakehouse can then be directly accessed by BI tools (such as Power BI, Tableau) via optimized query engines and used for data science and ML tasks, all while adhering to centralized security and governance policies enforced by Unity Catalog.

4 Benefits and Performance of the Lakehouse

The Lakehouse architecture delivers significant advantages over traditional fragmented approaches:

4.1 Key Benefits

- **Handles All Data Types:** Processes structured, semi-structured, and unstructured data on a single platform.
- **Cost-Effective Storage:** Leverages cheap cloud object storage (around $2.30/TB), offering roughly a 90% reduction in storage costs compared to data warehouses.

- **Unified Workloads:** Supports BI, SQL analytics, Data Science, Machine Learning, AI, and streaming on a single, consistent copy of the data.
- **Direct BI Integration:** BI tools connect directly to the Lakehouse, ensuring analysts have access to the most up-to-date data without duplication.
- **Reliability and Data Management:** ACID transactions, versioning (time travel), schema management, and efficient upserts/deletes prevent data swamps and ensure data quality.
- **Improved Performance:** Query engines optimized for the Lakehouse provide competitive or superior performance for many workloads compared to both data lakes and traditional warehouses.
- **Simplified Architecture:** Eliminates the need for maintaining separate systems for different workloads (warehouses, lakes, streaming layers), reducing operational overhead and complexity.

4.2 Performance Benchmarks

Benchmarks highlight the tangible performance and cost benefits:

Metric	Improvement	Context
BI Query Speed (TPC-DS)	1.2x faster vs Warehouse	Lakehouse query engines are highly optimized for SQL.
Storage Cost	90% Reduction	Lakehouse utilizes inexpensive cloud object storage.
Data Pipeline Failures	83% Reduction	Achieved by customers due to ACID reliability and schema enforcement.
ML Model Deployment	6x Faster	Enabled by unifying data prep and ML on a single, reliable platform.

Table 1: Lakehouse Performance Gains and Cost Reductions (Source: Databricks internal benchmarks and customer reports)

These improvements directly translate to faster time-to-insight, reduced operational costs, and increased agility for data teams.

5 Real-World Adoption and Getting Started

The Lakehouse architecture is being adopted by leading organizations across industries to drive significant business outcomes.

5.1 Customer Success Stories

Examples of Lakehouse impact include:

- **Goldman Sachs**: Processing 45TB/day of transaction data for sub-second fraud detection.
- **Moderna**: Accelerating vaccine research data cycles from weeks to hours.
- **Starbucks**: Unifying and leveraging real-time store and mobile app data for personalized recommendations.
- **Unilever**: Consolidating 17PB of previously siloed data, leading to 83% fewer pipeline failures and 6x faster ML deployment.

5.2 Getting Started with Lakehouse

Migrating to or implementing a Lakehouse typically follows a phased approach:

1. **Assess Current Architecture:** Understand existing data sources, workloads, and identify key pain points the Lakehouse can solve.
2. **Pilot with Delta Lake:** Start by landing new data or converting a subset of existing data to Delta Lake format to demonstrate reliability and performance improvements.
3. **Implement Unity Catalog:** Establish centralized governance, security, and discovery for the pilot data and plan for broader implementation.
4. **Migrate High-Value Workloads:** Move critical BI and AI workloads onto the Lakehouse to realize immediate business value and consolidate data.

6 Conclusion: The Future is Lakehouse

In summary, data warehouses excelled at traditional BI but were ill-equipped for modern data diversity and AI. Data lakes offered flexibility and cost but suffered from reliability and governance issues. The Data Lakehouse architecture combines the best of both worlds, providing a unified, reliable, cost-effective, and high-performance platform for all data, analytics, and AI workloads. Enabled by foundational technologies like Delta Lake and Unity Catalog, the Lakehouse simplifies the data stack, accelerates innovation, and allows organizations to finally leverage the vast majority of their data that currently remains unused. The Lakehouse is not just another architecture; it is becoming the de facto standard and the foundation for the next decade of data-driven innovation.