



Review Article

Building a Modern Cloud Data Platform with Databricks


Dr. Manish Kumar ^{1*}, Dr. Ashish Kumar Saha ²

¹ Assistant Professor, University Department of Computer Applications,
Vinoba Bhave University, Hazaribag, Jharkhand, India

² Associate Professor, Department of Chemistry
Vinoba Bhave University, Hazaribag, Jharkhand, India

Corresponding Author: *Dr. Manish Kumar

DOI: <https://doi.org/10.5281/zenodo.15585840>

Abstract	Manuscript Information
<p>Databricks offers a comprehensive analytics platform designed for managing extensive data sets in the cloud. It is based on Apache Spark, an open-source cluster computing framework that is optimized for rapid processing of large-scale data workloads. The company was established by the same engineers from the University of California, Berkeley, who initially developed Spark, which subsequently became an Apache project. A significant innovation of Databricks is its 'lakehouse' architecture, which merges the advantages of data lakes (for the storage of vast quantities of raw data) and data warehouses (for structured analytics). Databricks utilizes cloud object storage as a cohesive interface for data engineering, data science, and analytics. This article demonstrates how Databricks implements the Lakehouse architecture to provide a contemporary cloud data platform.</p>	<ul style="list-style-type: none"> ▪ ISSN No: 2583-7397 ▪ Received: 23-04-2025 ▪ Accepted: 29-05-2025 ▪ Published: 03-06-2025 ▪ IJCRM:4(3); 2025: 266-269 ▪ ©2025, All Rights Reserved ▪ Plagiarism Checked: Yes ▪ Peer Review Process: Yes <p>How to Cite this Article Kumar M, Saha AK. Building a Modern Cloud Data Platform with Databricks. Int J Contemp Res Multidiscip. 2025;4(3):266-269.</p>
	<p>Access this Article Online</p>  <p>www.multiarticlesjournal.com</p>

KEYWORDS: Data bricks, lackhouse, cloud storage, artificial intelligence, machine learning

INTRODUCTION

Given the current landscape of analytics, which encompasses everything from basic SQL reports to sophisticated machine learning predictions, there is a pressing need to establish a centralized, open-source data lake that integrates data from various sources and facilitates access for diverse use cases. Presently, most organizations find it challenging to achieve this

objective on a large scale due to the associated complexity and costs. However, with Databricks' open and unified data platform, which streamlines data management and artificial intelligence for extensive data engineering, collaborative data science, comprehensive machine learning lifecycles, and business analytics, the possibilities are limitless. A lakehouse employs a

cost-effective cloud object storage as its data storage layer, allowing for virtually unlimited scalability at minimal expense. For instance, one can easily set this up by creating an AWS S3 Bucket or a Microsoft Azure Data Lake Storage Gen2 repository. To move over your data from current applications, databases, data warehouses, and other data stores, you can use Databricks Ingest, a service that quickly and easily loads data into your lakehouse [1].

Utilizing Delta Lake to Add Reliability to Your Lakehouse

Delta Lake addresses the data reliability problems that have plagued data lakes, making them data swamps. The open-source storage layer that Delta Lake provides brings improved reliability to data lakes. Delta Lake on Databricks allows you to configure data lakes based on your workload patterns and provides optimized layouts and indexes for fast, interactive queries and sits on top of object storage. The format and the compute layer help simplify building big data pipelines and increase the overall efficiency of your pipelines [2].

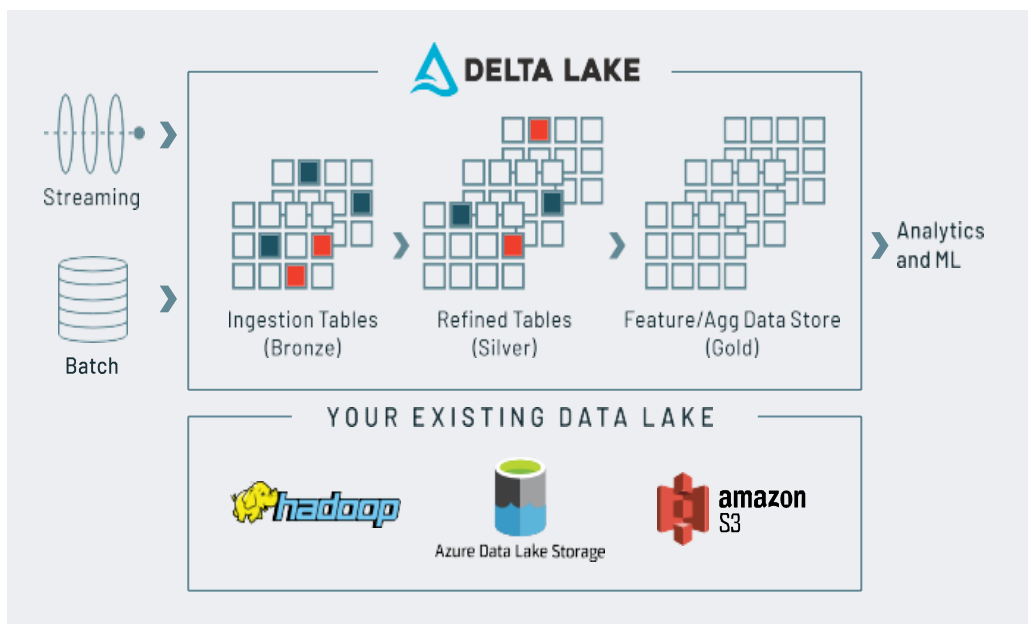


Figure 1: Delta Lake is an open-source storage layer that brings improved reliability to the lakehouse

Since Databricks made Delta Lake open-source in 2019, numerous organizations have begun utilizing the open-source Lakehouse, which has become significantly more reliable and efficient. The atomicity, consistency, isolation, and durability (ACID) transactions of Delta Lake, along with its effective indexing, are crucial for accommodating various data access patterns, ranging from ad hoc SQL queries in business intelligence (BI) tools to machine learning models developed in Python and R. This approach of establishing a single, central, reliable, and efficient source of truth for open-format data caters to use cases that span from BI to machine learning, while allowing for decoupled compute and storage, forms the core of the lakehouse methodology. Nevertheless, it is vital to guarantee the reliability of data within a lakehouse from the beginning to prevent potential data corruption issues in the future. Generally, two data ingestion scenarios must be considered [3]:

Data acquisition from external sources: You generally possess important user information across multiple internal data sources. The Databricks Data Ingestion Network provides an automated

method to fill your lakehouse from numerous data sources into Delta Lake.

Data acquisition from cloud storage: You already have a mechanism to pull data from your source into cloud storage. As new data arrives in cloud storage, you can load this new data by using the Delta Lake Auto-Loader capability in Databricks.

Incorporating Delta Engine to Enhance Performance in Your Lakehouse: Delta Engine provides exceptional performance across all workloads on Delta Lake, encompassing ETL data pipelines, SQL analytics, real-time analytics, data science, and machine learning. Delta Engine is entirely compatible with Spark APIs. It comprises three fundamental components:

Vectorized query engine: This encompasses a newly developed massively parallel processing (MPP) engine, built from the ground up using C++, along with a fully vectorized engine designed for contemporary SIMD (single instruction, multiple

data) hardware. It has been optimized for current workloads, removing null checks and enhancing string processing.

Superior query optimizer: A cost-based optimizer is incorporated to enhance physical plans, along with adaptive query execution that facilitates dynamic rescheduling during runtime. Additionally, this encompasses dynamic partition pruning and runtime filters to exclude irrelevant data.

Intellectual caching: The Delta Engine autonomously caches input data and distributes workloads evenly across a cluster. Additionally, it utilizes advancements in Non-Volatile Memory Express (NVMe) SSDs, employing industry-leading columnar compression methods, which can yield performance enhancements of up to ten times for both interactive and reporting tasks.

Leveraging Databricks Unified Data Analytics Platform as Your Lakehouse

The Databricks Unified Analytics platform embodies the architectural features of a lake house. Organizations aiming to construct and implement their own systems can utilize open-

source file formats, such as Delta Lake, which are particularly suitable for developing a lake house. Figure 4-2 demonstrates the straightforward nature of the lake house methodology employed by the Databricks Unified Analytics platform. Users of the lake house on the Databricks Unified Analytics platform also benefit from a range of standard tools (Spark, Python, R, ML libraries) tailored for non-BI tasks like data science and machine learning. Data exploration and refinement are fundamental components of numerous data science and analytics applications. Delta Lake is engineered to empower users to progressively enhance data quality from their data center to operational environments. To facilitate machine learning management and maintain oversight, your organization requires a solution that enables the orchestration and management of models to expedite deployment while upholding governance standards. Databricks provides this assistance, offering a business process management solution that aids in the operationalization of machine learning models. This encompasses support for the creation, storage, testing, comparison, approval, publication, monitoring, and, when necessary, retraining of these models in an automated and regulated fashion.

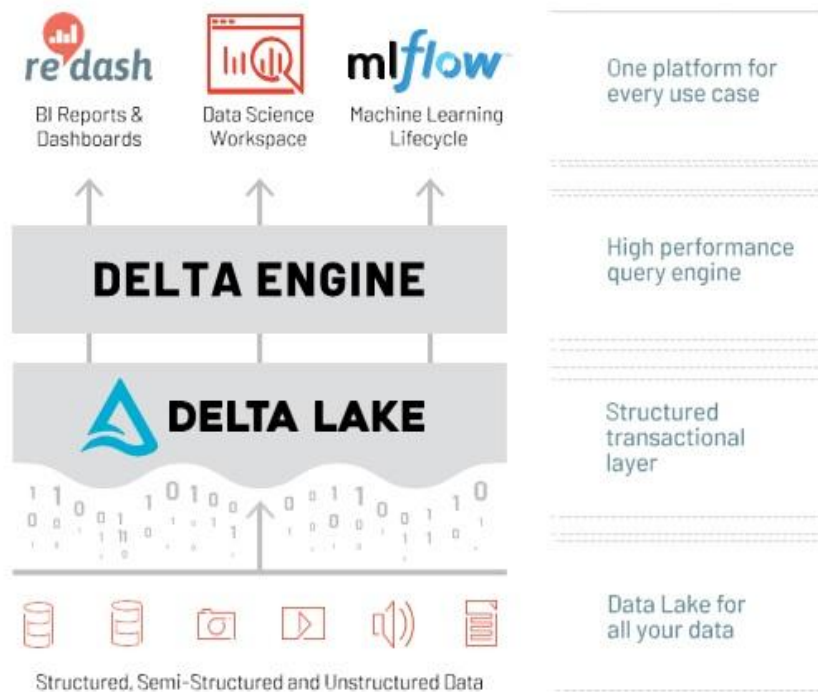


Figure 2: The Databricks Unified Data Analytics Platform

Simultaneously, this automated architecture represents a "build one, use many" solution that minimizes the need for manual human intervention and enhances customers' capacity to operationalize machine learning models. This capability is facilitated by MLflow, an open-source platform created by Databricks to assist in managing the complete machine learning lifecycle with enterprise-level reliability, security, and

scalability. Evidence from multiple Databricks implementations indicates that the productivity of data teams increases by 20% when utilizing the Lakehouse solution [4].

Sharing a Customer Case Study

A leading global entity in technology and entertainment required a cloud-based data platform. The organization served millions of

residential broadband customers utilizing video, high-speed Internet, and telephony services. Additionally, it managed vast amounts of data derived from billions of events generated by users' entertainment systems and voice remote controls, resulting in petabytes of data that required preparation for analysis. The company also faced challenges with a fragile and complex data pipeline that frequently experienced downtime and was difficult to restore. Furthermore, there was collaboration among data scientists within the organization, who were distributed globally and utilized various programming languages, leading to difficulties in sharing and reusing code. Tensions existed between development and deployment roles: development teams aimed to leverage the latest tools and models, while operations personnel preferred to deploy on established infrastructure. The client sought a lakehouse solution, and Databricks assisted in designing the new cloud data platform for the company [5].

CONCLUSION

By utilizing Databricks' unified analytics platform, the technology and entertainment firm has developed extensive data sets on a large scale. This solution has also allowed the organization to enhance machine learning capabilities at scale, optimize workflows among teams, promote collaboration, minimize infrastructure complexity, and provide exceptional customer experiences. The simplification of infrastructure management has further led to a decrease in operational costs through automated cluster management and economical features like auto-scaling and spot instances. The establishment of collaborative workspaces throughout the organization has enabled interactive notebooks to enhance team collaboration and creativity in data science, significantly speeding up model prototyping for quicker iterations. Reliable ETL at scale with Delta Lake has facilitated efficient analytics pipelines that can consistently link historical and streaming data for a more thorough analysis.

REFERENCES

1. Alagiannis I, Borovica-Gajic R, Branco M, Idreos S, Ailamaki A. NoDB: Efficient query execution on raw data files. *Commun ACM*. 2015 Nov;58(12):112–21.
2. Apache Hadoop [Internet]. Available from: <https://hadoop.apache.org>
3. Armbrust M, Das T, Sun L, Yavuz B, Zhu S, Murthy M, Torres J, van Hovell H, Ionescu A, Muszczak A, Switakowski M, Szafranski M, Li X, Ueshin T, Mokhtar M, Boncz P, Ghodsi A, Paranjpye S, Senster P, Xin R, Zaharia M. Delta Lake: High-performance ACID table storage over cloud object stores. *Proc VLDB Endow*. 2020.
4. Armbrust M, Xin RS, Lian C, Huai Y, Liu D, Bradley JK, Meng X, Kaftan T, Franklin MJ, Ghodsi A, Zaharia M. Spark SQL: Relational data processing in Spark. In: *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*; 2015. p. 1383–94.
5. Stonebraker M. Why the 'data lake' is really a 'data swamp'. *BLOG@CACM* [Internet]. 2014. Available from:

<https://cacm.acm.org/blogs/blog-cacm/176450-why-the-data-lake-is-really-a-data-swamp/>

Creative Commons (CC) License

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY 4.0) license. This license permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

About the Corresponding Author



Dr. Manish Kumar is an Assistant Professor in the University Department of Computer Applications at Vinoba Bhave University, Hazaribag, Jharkhand, India. With a strong academic and research background in computer science, he is dedicated to teaching and advancing knowledge in areas related to software development, data science, and emerging technologies.