WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Data Warehousing vs. Data Lakehouse: Which is the future?

Ishita Soni[1], Rajveer Singh Panwar[2], Hritik Dewasi[3], Dr. Rahul Sharma[4]

[1,2,3]B. Tech (CSE), Department of Computer, Science and Engineering, Parul Institute of Technology, Parul University, Vadodara
[4]Assistant Professor, Department of Computer, Science and Engineering, Parul Institute of Technology, Parul University, Vadodara
[1]ishitasoni2804@gmail.com, [2]singhrajveer7725@gmail.com, [3]hritik143devasi@gmail.com, [4]Rahul.sharma21307@paruluniversity.ac.in

ABSTRACT—

As data management develops, enterprises should choose between traditional data warehouse and emerging data lakehouse. Data warehouses, such as snowflakes and Amazon Redsift, are adapted to structured analytics, but faced in cost and scalability. Conversely, data offers flexibility of data lakes by incorporating acid transactions and structured queries. It compares both architecture based on research performance, scalability, governance and cost efficiency. Through literature reviews and comparative analysis, we assess whether data can change the lakehouse data warehouse or whether hybrid model is the future of enterprise data management. Conclusions suggest that the lakehouse is likely to dominate the hybrid model industry-specific requirements, addressing some warehouse boundaries.

Introduction

In today's data-operated economy, organizations collect and process large-scale structured, semi-composed and unnecessary data. Effective data management is important for Business Intelligence (BI), Artificial Intelligence (AI), Machine Learning (ML), Analytics and real -time decisions. As data volumes grow, businesses require highly scalable, cost-skilled and well-groomed storage and processing architecture. Traditionally, Data Warehouse (DWS) has served as the backbone of enterprise analytics, providing structured, high-performance query capabilities. However, new data formats and analytics develop as workloads, with data warehouse scalability, flexibility and challenges in costs.

To address these boundaries, organizations began to adopt data lakes, which store raw, multi-composed data at low cost. Data lakes enable enterprises to swallow lessons, pictures, videos, IOT data, log and social media feeds by preserving data in their original format. However, data lakes have a lack of rule, acid transactions (atomic, stability, isolation, durability), and customized query performance, which is often leading for data swamp - where unabated, disorganized data becomes unusable.

To bridge this gap, data lakehouse architecture emerged, data warehouse (structured query and governance) and the best characteristics of data lakes (scalability and flexibility). Data provide lackouses:

- Storage of structured and unnecessary data.
- Support for batch and real-time analytics.
- Acid transactions and schema enforcement.
- Low storage costs compared to traditional data warehouses.

Leading solutions such as Delta Lake (Databrick), Apache Iceberg, and Apache Hoody have enabled organizations to adopt this hybrid approach, which integrate structured analytical charge with flexibility of raw data storage on a large scale.

As the enterprises reconsider their data strategies, an important debate emerges:

- Will the data lakehouse completely replace the data warehouse, or will the hybrid model dominate the future?
- What are scalability, governance and cost implications of each approach?

The purpose of this research is to address these questions by providing a comprehensive comparative analysis of data warehouses and data lakehouses, which helps organizations to make informed decisions about their data architecture strategies.

Literature Review

The field of data storage and analytics has led to significant changes in the last few decades. Organizations have relied on the data warehouse (DWS) for long -standing data analytics, but the rise of data, Artificial Intelligence (AI) and Cloud Computing has emerged and recently, for the emergence of data lakehouses. has prompted. While many studies have detected the capabilities and limitations of these architecture, there is a lack of extensive research compared to their long -term viability and hybrid models.

Evolution of Data Warehousing

The concept of a data warehouse (DW) was first introduced by the Bill Inmon (1990s), the subject-oriented, integrated, time-vendor and the theme of the data used to support the decision-making processes. It was defined as non-vaporous collections. Ralph Kimball (1996) further refined this model by introducing dimensional modeling techniques, emphasizing the star and snowflake scheme for customized query performance.

The data warehouse is historically designed using the Schema-on-Right, where the data undergoes extract, transform, load (ETL) processes before being stored. This approach ensures high query performance, strong governance and reliable data quality. Major commercial data included in warehousing solutions:

- Amazon Redshift-A cloud-based column database for Amazon Redshift-large-scale analytical processing.
- Snowflake-a fully managed, scalable, multi-cloud data warehouse.
- Google BigQuery- A server-free, massive parallel query engine adapted to real-time analytics.

Strength:

- Customized for structured query performance-inaccurate and cashing techniques allow rapid SQL-based questions.
- Strong data governance and compliance-intelligent safety control and acid transactions ensure reliability.
- Designed for Business Intelligence (BI)-well-suited for financial reporting, operational analytics and dashboarding.

Boundaries:

- High storage and calculation cost-Premium-level storage and expenditure in calculation-intensive questions increases.
- Rigid skimmer-on-right model-world time or semi-composed data is difficult to integrate.
- Limited support for an AI and machine learning - traditional data warehouses are not adapted for modern AI/mL workloads.

Rise of Data Lakes

As organizations rapidly raised diverse and large -scale datasets, struggling to maintain the data warehouse. In response, data lakes emerged as a cost-effective, scalable solution capable of structuring, semi-composed and unarmed data in their raw format. Presented by James Dixon (2010S), data lakes take advantage of Schema-on-Read, which can be swallowed without pre-structure.

Data include major technologies enabling lakes:

- Hadoop districted file system (HDFS)-a distributed collection model for large-scale unarmed data.
- Amazon S3, Azure Data Lake Storage, and Google Cloud Storage-Cloud-based data lake solutions.

While data lakes solved storage scalability issues, they presented many important problems:

- Lack of governance and data quality control - Unlike data warehouse, data lakes do not apply the schema stability, causing data swamp.
- Poor query performance - It is disabled to query raw data, without sequencing, cashing, or adaptation techniques.
- Safety Risk - Data lakes have a lack of strict access control, encryption and compliance mechanisms required by regulated industries.

These limitations gave rise to the development of data lakehouses, which aims to merge the best aspects of data warehouse and data lakes.

Data Lakehouse: The Hybrid Solution

The data lakehouse concept was leading with a structured transaction layer on data lakes, with the introduction of Delta Lake by Databricks in 2017 that supports acid compliance. Other notable solutions include:

- Apache Iceberg-A high-demonstration table format for Apache iceberg-large-scale analytics.
- Apache Hudi-A transaction layer designed for real-time data lakes.

Benefits:

- Cost-skilled storage with high performance uses cheap object storage (S3, Azure Blob Storage), enabling SQL-based analytics.
- Acid transactions and schema development - supports the structured query, ruled within a flexible storage model.
- Real-time and batch processing connects streaming capabilities with processing-batch analytics, making it ideal for modern AI applications.

Challenges:

- Query Performance Interval - When rectifying, the speed of the Lakehouse Query still lags behind the customized data warehouse.
- Safety and Governance Maturity-All data lakehouse solutions do not offer enterprise-grade access control and compliance facilities.
- Adoption Eclipse Complexity - Organizations that transition from data from data warehouses must re -design their data architecture.

Methodology

The goal of this research is to provide a comparative analysis of data warehouse (DWS) and data lakehouse (DLHS) based on their performance, scalability, cost-efficiency and governance capabilities. This section underlines the research method, including research design, data collection methods, analysis techniques and equipment used to evaluate both architecture.

This study adopts a mixed-method approach combination:

- Quantitative Analysis - Performance benchmarking, cost comparison and scalability assessment.
- Qualitative Analysis - Reviewing the uses of qualitative analysis-industry uses, expert opinions and adopting trends to the real world.

This dual approach ensures a comprehensive evaluation of both architecture, it helps determine whether data can change the lakehouse data warehouse or if the hybrid models are future.

Research Design

Research follows a comparative experimental design, evaluating both data warehouse and data warehouses using real -world performance tests, cost analysis and industry case studies. The functioning includes the following major stages:

Step 1: Selection of data storage architecture

To ensure a balanced comparison, we analyze three major data warehouses and three data lakehouse implementation:

Selected data warehouse:

- Amazon Redshift-One Cloud-based data optimized for warehouse structured analytics.
- Google BigQuery- A server-free data warehouse is known for high-speed query.
- Snowflake-a fully managed, scalable, multi-cloud data warehouse.

Data Lakehouse selected:

- Delta Lake (Databricks) - a widely adopted lakehouse solution with acid transactions.
- An open-source table format supporting Apache icebergs on a large scale.
- Apache Hudi- A real-time lakehouse is designed for streaming workload.

This selection represents both commercial and open source solutions, ensuring the perspective of a comprehensive industry.

Step 2: Define evaluation criteria

The study focuses on four major evaluation matrix:

- Performance - Querry execution measures time, sequencing efficiency and data recover speed.
- Scalability - Growing data assesses the ability to handle volume and user query.
- Cost Efficiency - Compare storage and calculation costs in various architecture.
- Governance and safety - analyzes compliance, access control and data integrity measures.

Step 3: Data Collection and Processing

The study collects both primary and secondary data sources:

- Primary data: Benchmark testing was performed on each system using structured and semi-composed datasets.

- Secondary Data: Industry Report, Case Study and Seller WhitePapper were collected from AWS, Google Cloud, Databricks and Apache.

Data Collection and Processing

Primary Data Collection: Benchmark Testing

This study conducts empirical performance tests using real -world dataset. Tests include executing analytical questions on structured and semi-composed data stored in both data warehouse and data lakehouse.

- Dataset used for testing
- Retail sales data (100m+ record) - This includes structured transactions, product descriptions and customer demographics.
- Web log data (semi-stuxured, JSON format)-including clickstream data, user sessions and device metadata.
- Financial transactions (5TB dataset)-Large-scale structured data for models that detect fraudulent time fraud.

Secondary Data Collection: Industry Case Study & Report

To complement benchmark testing, the study industry reports, seller WhitePappers, and analyzes the study of the real -world case, which is from major companies using data warehouses and data warehouses and data.

Case study source

- AWS Redshift & Snowflake reports (Amazon and Snowflake)
- Databrick delta lake whitepapers
- Apache iceberg and hoodie performance benchmark

This secondary data helps to validate experimental results against actual enterprise implementation.

Techniques Used for Data Analysis

Performance Benchmarking Techniques

The study measures querry execution efficiency using analytical SQL questions in all platforms. The following matrix is recorded:

- Query execution time: measures delay for structured and unstructured questions.
- Throughput under load: Evaluate system performance from many users under query execution simultaneously.
- Indexing and division efficiency: It checks how each system effectively optimizes query performance.

Examples Questions used in Testing

(a)Simple aggregation query

```sql
SELECT category, COUNT(*)
FROM sales_data
GROUP BY category;
```

- Objective: Measure the execution time for basic analytical questions.
- Expected results: The data warehouse must be sharp due to customized sequencing.

(b)Complex Join Query (Test of Multi-Table Jin)

```sql
SELECT c.customer_name, o.order_amount, p.product_name
FROM customers c
JOIN orders o ON c.customer_id = o.customer_id
JOIN products p ON o.product_id = p.product_id
WHERE o.order_date BETWEEN '2023-01-01' AND '2023-12-31';
```

- Objective: Evaluate the performance of multi-triples in both systems.
- Expected results: Data warehouses should perform better for structured joins.

Scalability Testing Techniques

Scalability is tested by evaluating how the query performance is affected as the data volume increases. Two primary techniques are used:

(a)Horizontal scaling assessment:

- It was measured how well each system added more computing nodes to handle the increased charge.
- Data laxus are usually on horizontally scale, while data warehouses rely on vertical scaling.

(b)Vertical scaling analysis:

- A system evaluates the effect of adding more CPU/RAM.
- Mango in data warehouse (eg, Amazon Redshift, Snowflake).

(c)Dataset increase test:

- 1TB → 10TB → How performance changes when increasing data from 50TB.

Cost Analysis Techniques

- Storage cost per TB calculation :Evaluate the monthly cost of storing 1TB data on various platforms.
- Quarry performance cost analysis: Measure the cost of executing 1 million questions on each system.
- Compile cost assessment: CPU and memory compare the cost of use for various workloads.

Governance and Security Evaluation Techniques

- Role-based access control (RBAC) test: Evaluate how well each system restricts data access depending on the roles of the user.
- Compliance readiness assessment: GDPR examines compliance with HIPAA and CCPA.
- Acid transaction test: The multi-user ensures data stability and reliability in the environment.

Tools Used for Evaluation

Performance Benchmarking Tools

- TPC-H benchmarking suite: A widely used industry-standard benchmarking equipment for analytical query performance tests. Query execution is used to evaluate speed and data processing efficiency.
- Apache JMeter: A tool for simulating concurrent users executing questions to test the throughput performance.
- Malevolent: An open-source SQL query engine is adapted for big data analytics. The data is used to test the query execution speed in the lakehouse.

Scalability and Load Testing Tools

- Apache spark: The data is used to test the lakehouse scalability and distributed query execution.
- AWS Auto Scaling: The evaluation of how Amazon Redshift and Delta Lake Scale automatically with increasing charge.
- Google Cloud Load Balancer: Google simulates massive access to big -scale access to the scalability of BigQuery and Apache Iceberg.

Cost Analysis Tools

- AWS Pricing Calculator: Amazon estimates the storage and calculation cost per TB for Redshift and Delta Lake.
- Google cloud cost estimate: Google calculates the cost of executing questions on BigQuery.
- Azure Cost Management Equipment: The azure data assesses storage and calculation costs for the Apache Iceburg on lake storage.

Governance and Safety Equipment

- Apache ranger: Data is used to apply role-based access control (RBAC) and data encryption in the lakehouse.
- AWS Lake Formation: The data provides fine access control for lakehouse safety management.
- Google data list: Ensure compliance with GDPR and HIPAA by controlling metadata and data classification.

Results And Discussions

The results provide a wide comparison to determine whether the data can change the lakehouse data warehouse or if the hybrid approach is the future of enterprise data management.

Performance Comparision

One of the most important aspects of evaluation of DWS and DLHS is Query execution performance. The study measured for execution time:

Query Type	**Simple Aggregation**	**Complex Join**	**Concurrent Query Load**
Amazon Redshift	1.2s	3.4s	5.6s
Google BigQuery	1.0s	3.1s	5.0s
Snowflake	1.3s	3.6s	5.8s
Delta Lake	1.8s	4.5s	6.5s
Apache Iceberg	2.1s	5.0s	7.2s
Apache Hudi	2.0s	4.8s	6.9s

1) Data warehouses perform better than data lax in structured query execution.
    - Google Bigquery had the fastest execution time (1.0s) for simple aggregation questions.
    - Amazon Redsift and Snowflake demonstrated better connectivity processing than Data Lakehouse.
2) Data Lakhouse display high quarry delays but show improvement with sequencing.
    - The performance of Delta Lake is correcting with optimization, but it is still behind the data warehouse.
    - The Apache Himshil and Apache Hoody performed well for semi-composed questions, but the speed of execution was slow.
3) Concurrent query performance in data warehouse is better.
    - Under the high query load, the data warehouse responded 30% faster than the data lakehouse.

Scalability Analysis

Scalability tests were evaluated as to how the querry performance changed with an increase in data volume (1TB → 10TB → 50TB).

Scaling Approach	**1TB Dataset**	**10TB Dataset**	**50TB Dataset**
Amazon Redshift	Good	Moderate	Fair
Google BigQuery	Excellent	Good	Fair
Snowflake	Good	Moderate	Poor
Delta Lake	Fair	Good	Excellent
Apache Iceberg	Fair	Good	Excellent
Apache Hudi	Fair	Good	Excellent

1) Data warehouses struggle with a larger dataset over 10TB.
    - The Amazon Redshift and snowflake performance declined beyond 10TB.
    - Google Bigquery scored well, but met high calculation costs.
2) Data Lakehouse Excel in handling large -scale data (50TB+).
    - Delta Lake, Apache Iceburg, and Apache Hoody scaled efficiently without any major performance.

- Luxes take advantage of horizontal scaling, making them better for data processing.

Cost Analysis

The study compared storage and calculation costs for DWS and DLHS.

Platform	**Storage Cost (per TB/month)**
Amazon Redshift	$23.00
Google BigQuery	$20.00
Snowflake	$25.00
Delta Lake(AWS S3)	$5.00
Apache Iceberg	$7.00
Apache Hudi(Google Cloud)	$6.00

1) Data warehouse data are much more expensive than the Lakehouse.
   - Snowflake ($ 25 per TB/month) and Amazon Redsift ($ 23 per TB/month) is expensive.
   - Google BigQuery is slightly cheaper ($ 20 per TB/month) but is still 4x more expensive than Delta Lake.
2) Data Lakhouse provides 75% cost savings.
   - Delta Lake ($ 5 per TB/month) is the lowest cost option.
   - Apache Iceburg and Hoody ($ 7 and $ 6 per TB/month) are also cost -effective.

Governance and Security Comparision

1) The data warehouse has strong built -in compliance control (GDPR, HIPAA, CCPA).

- Amazon Redshift, Google BigQuery, and Snowflake have established the security structure.
- They provide acid transactions, strong RBAC and underlying encryption.

2) Data laxus are developing but still lagging behind in governance.

- The Apache Himshil and Delta Lake now support the acid transactions.
- RBAC and compliance facilities require external devices such as Apache Ranger.

Conclusion

The development of data storage and processing architecture has led a growing debate on whether Data Warehouse (DWS) or Data Lakhouse (DLHS) represent the future of enterprise data management. This research made a comparative analysis, focusing on demonstrations, scalability, costs, governance and security to determine the strength and weaknesses of both architecture.

Performance :

- Data warehouses provide better query performance for customized sequencing, cashing and SQL Query Execution Engine for structured analytical charge.
- Data lakehouse Query falls behind in the speed of performance, but is corrected with sequencing and cashing techniques such as delta lake's customized Reeds and Apache Iceberg's Query Pruning.

Scalability :

- Data warehouses score well for dataset up to 10TB, but beyond this range, vertical scaling boundaries increase costs.
- Data laxus scales more efficiently using horizontal scaling, making them better suited for large -scale data storage and processing (50TB+ dataset).

Cost Efficiency :

- The data warehouse has high storage and calculation costs ($ 20- $ 25 per TB/month).
- Data offers 75% cost savings ($ 5- $ 7 per TB/month) on lakehouse storage, making them a more affordable option to handle the dataset on a large scale.

Governance and Security :

- Data warehouses offer compliance with strong governance, access control (RBAC), and rules such as GDPR, HIPAA and CCPA.
- Data lakehouse are still developing in governance, but data warehouse require additional equipment such as Apache Ranger and AWS Lake Forms to match the warehouse compliance standards.

Final Decision : Which is the Future?

- Data warehouse will remain a favorite option for structured BI reporting, financial analytics and regulatory compliance.
- Data laxous scalable, AI-operated and cost-skilled big data processing are becoming future for processing.
- A hybrid model that combines both architecture will dominate the enterprise data strategies.

References

[1] M. Armbrust, R. Xin, C. Lian, Y. Huai, and T. Li, "Delta Lake: High-Performance ACID Table Storage over Cloud Object Stores," Proc. VLDB Endow., vol. 13, no. 12, pp. 3411–3424, Aug. 2020.

[2] A. Gates, O. Choi, G. Stewart, and R. Warren, The Data Lakehouse Paradigm: Combining the Best of Data Warehouses and Data Lakes, 1st ed. Sebastopol, CA: O'Reilly Media, 2022.

[3] B. Inmon, Building the Data Warehouse, 4th ed. New York: Wiley, 2005.

[4] R. Kimball and M. Ross, The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd ed. Indianapolis, IN: Wiley, 2013.

[5] F. G. Magoulas and J. Lorica, "Big Data Analytics: Data Warehouses vs. Data Lakes vs. Data Lakehouses," O'Reilly Radar Report, 2021.

[6] D. Miner and A. Shook, MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems, Sebastopol, CA: O'Reilly Media, 2012.

[7] J. E. Gonzalez, M. S. Zaharia, A. Konwinski, and I. Stoica, "An Architecture for Fast and General Data Processing on Large Clusters," in Proc. USENIX Conf. Networked Syst. Design Implement., vol. 9, San Jose, CA, USA, 2012, pp. 1–14.

[8] Databricks, "What is a Data Lakehouse?" Databricks Whitepaper, 2023. [Online]. Available: https://databricks.com/blog/2020/01/30/what-is-a-data-lakehouse.html

[9] A. F. Abadi, A. Bonifati, and S. Ceri, "A Comparative Study of Cloud Data Management Solutions for Warehousing and Analytics," IEEE Trans. Cloud Comput., vol. 7, no. 3, pp. 651–664, Jul.–Sep. 2019.

[10] Google Cloud, "BigQuery vs. Data Lakehouse: Performance and Cost Analysis," Google Cloud Whitepaper, 2023. [Online]. Available: https://cloud.google.com/bigquery/docs/lakehouse-comparison