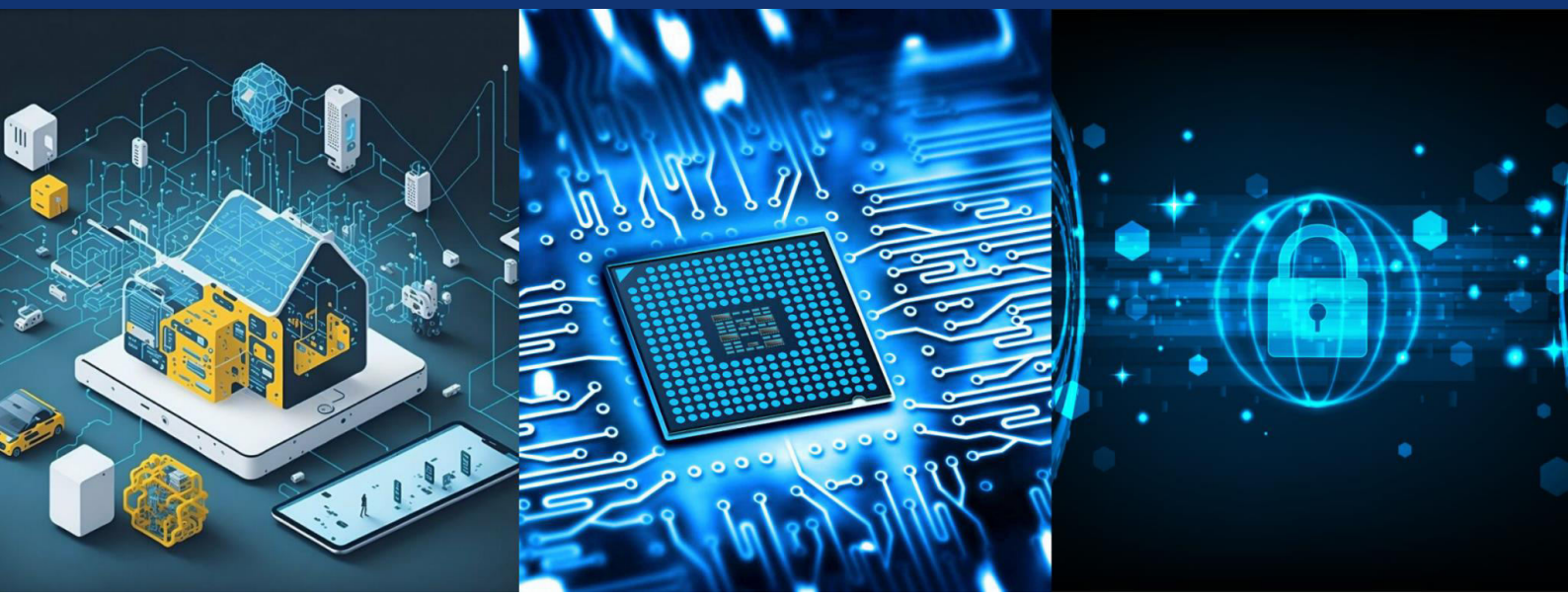
IJIRCCE

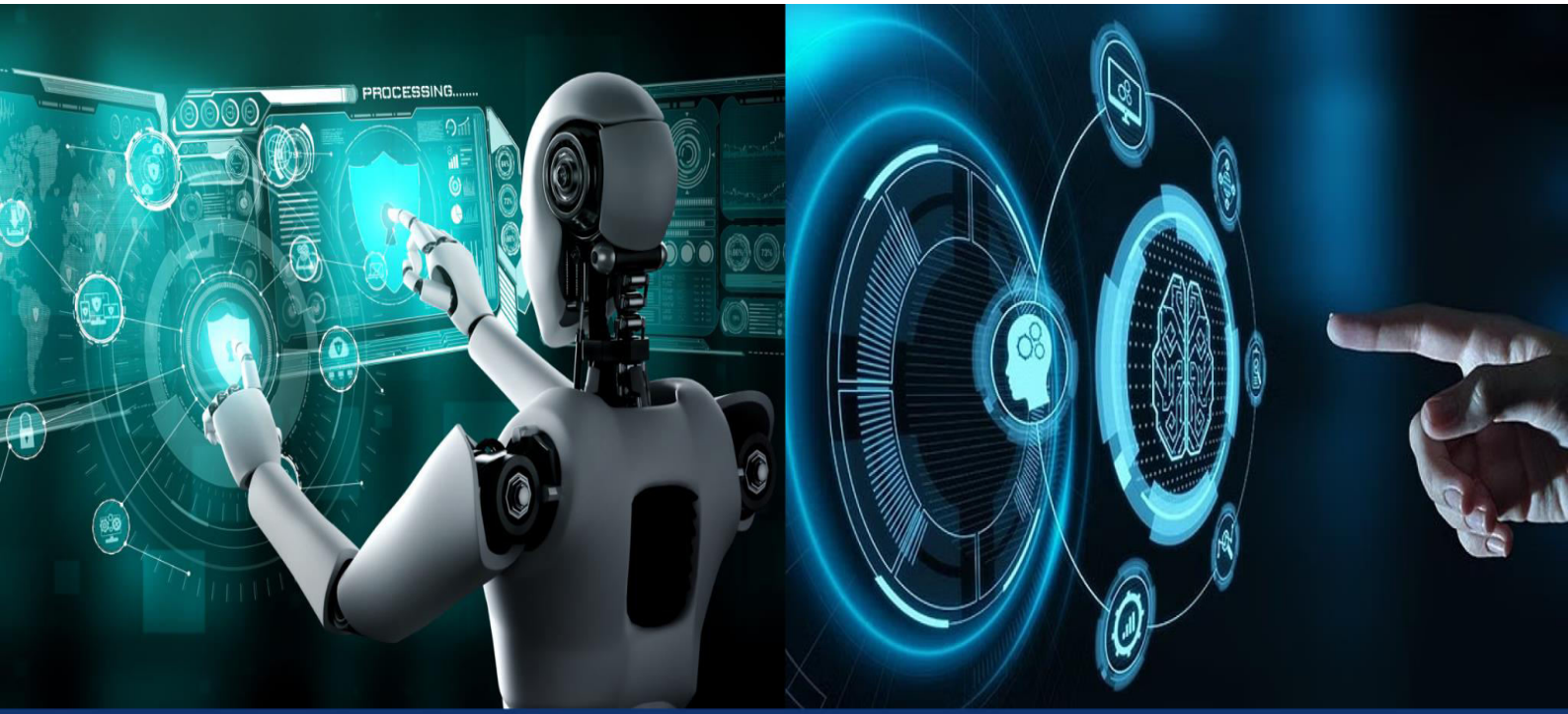
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.771

Volume 13, Issue 3, March 2025

© 2025 IJIRCCE | Volume 13, Issue 3, March 2025| DOI: 10.15680/IJIRCCE.2025.1303012

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Optimizing ETL Performance with Delta Lake for Data Analytics Solutions

Manjula Jayavel

Team Lead, Atos, Chennai, India

ABSTRACT: Extract, Transform, Load (ETL) processes are critical for efficient data analytics, yet they often struggle with performance bottlenecks, data consistency issues, and scalability challenges. Delta Lake, an open-source storage layer that enhances data reliability and performance, offers a robust solution for optimizing ETL workflows. This article explores how Delta Lake improves ETL efficiency, enhances data quality, and accelerates query performance in data analytics solutions. Through practical insights and best practices, we highlight how organizations can leverage Delta Lake to streamline their ETL pipelines for modern data-driven applications.

KEYWORDS: ETL Optimization, Delta Lake, ETL Performance, Data Compaction, Vacuuming, Scalable ETL Pipelines, Data Engineering Best Practices

I. INTRODUCTION

Extract, Transform, Load (ETL) is a fundamental process in data analytics that enables organizations to collect, clean, and structure data for meaningful insights. ETL workflows extract raw data from various sources, transform it into a structured format by applying business rules, and load it into a data warehouse, data lake, or analytics platform. This process is essential for powering business intelligence (BI), reporting, and advanced analytics.

With the rapid growth of data volumes and the increasing complexity of analytics use cases, traditional ETL processes face significant performance and scalability challenges. Organizations need high-speed, reliable ETL pipelines that can handle massive datasets efficiently while ensuring data quality and consistency.

Common Challenges in ETL Processes

Despite its importance, ETL often encounters several challenges that can impact data analytics solutions:

1. Slow Performance – ETL jobs, especially with large datasets, can suffer from long processing times due to inefficient data storage, lack of indexing, and redundant computations.
2. Data Inconsistency and Integrity Issues – Frequent updates, schema changes, and concurrent data modifications can lead to inconsistencies, causing inaccurate analytics results.
3. Scalability Constraints – Traditional ETL architectures may struggle to scale dynamically with growing data volumes, leading to bottlenecks and resource limitations.
4. High Storage Costs – Inefficient storage management, data duplication, and excessive intermediate files increase cloud storage expenses.
5. Lack of Real-Time Processing – Many ETL workflows are batch-oriented, making it difficult to support real-time analytics or incremental updates efficiently.

These challenges make it essential to adopt modern solutions that can enhance ETL performance, improve data reliability, and optimize resource utilization.

Introduction to Delta Lake as a Solution

Delta Lake is an open-source storage layer that brings reliability, performance, and scalability to data lakes. Built on top of Apache Spark, it provides key features such as:

- ACID Transactions – Ensures data integrity by allowing multiple ETL jobs to run concurrently without conflicts.
- Schema Enforcement & Evolution – Prevents schema inconsistencies and supports changes without breaking pipelines.
- Time Travel & Versioning – Allows users to access previous versions of data, enabling rollback and historical analysis.
- Optimized Storage & Performance – Uses Parquet format with indexing and caching for faster query execution.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- Seamless Integration with Big Data Tools – Works well with Spark, Databricks, and cloud-based data platforms.

By integrating Delta Lake into ETL workflows, organizations can overcome traditional ETL challenges and achieve faster, more scalable, and reliable data pipelines. This article explores how Delta Lake optimizes ETL performance and provides best practices for implementation.

II. UNDERSTANDING DELTA LAKE

Delta Lake is an open-source storage layer designed to bring reliability, performance, and scalability to data lakes. It was developed by Databricks and is built on top of Apache Spark. Delta Lake enhances traditional data lakes by adding transactional capabilities and schema enforcement, allowing organizations to maintain high data quality while enabling fast and efficient analytics.

Unlike traditional data lakes, which often suffer from data inconsistency and slow query performance, Delta Lake ensures data integrity and optimizes storage, making it an ideal solution for modern ETL and data analytics workloads.

Key Features of Delta Lake

1. ACID Transactions

Delta Lake provides Atomicity, Consistency, Isolation, and Durability (ACID) transactions, ensuring data reliability and correctness. This prevents issues like partial updates or data corruption, which are common in traditional data lakes when multiple processes modify the same dataset.

- Atomicity: Ensures that either all operations in a transaction succeed or none do.
- Consistency: Keeps data in a valid state before and after transactions.
- Isolation: Enables multiple ETL jobs to run concurrently without interfering with each other.
- Durability: Guarantees that once a transaction is committed, it remains available even after system failures.

2. Schema Enforcement and Evolution

Delta Lake ensures that incoming data adheres to a defined schema, preventing corrupted or inconsistent records from entering the data lake. At the same time, it supports schema evolution, allowing changes such as adding new columns without breaking existing queries.

- Schema Enforcement: Prevents unintended data type mismatches or missing fields.
- Schema Evolution: Allows adding new columns dynamically as data models evolve.

3. Time Travel and Versioning

Delta Lake maintains historical versions of data, enabling time travel to retrieve or restore previous data states. This is useful for debugging, auditing, and recovering lost or incorrect data.

- Versioning: Every change to a dataset is recorded as a new version.
- Rollback & Auditing: Users can query older versions for analysis or revert to previous states if necessary.

4. Scalability and Performance Optimization

Delta Lake is designed to handle massive-scale data workloads efficiently by leveraging:

- Optimized Storage with Parquet Format: Stores data in a columnar format for faster querying.
- Indexing and Caching: Uses Z-order indexing and Delta Caching to accelerate query performance.
- Data Compaction (OPTIMIZE & VACUUM Commands): Merges small files into larger ones to improve read performance and reduce storage costs.

III. ENHANCING ETL PERFORMANCE WITH DELTA LAKE

ETL processes are essential for preparing data for analytics, but traditional ETL pipelines often struggle with performance bottlenecks, data inconsistency, and scalability issues. Delta Lake enhances ETL performance by improving storage efficiency, ensuring data reliability, and optimizing resource utilization. This section explores how Delta Lake addresses these challenges.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Performance Improvements

1.Optimized Storage with Parquet Format

Delta Lake stores data in Apache Parquet, a columnar storage format that significantly improves query performance and compression efficiency. Compared to traditional row-based formats, Parquet enables:

- Faster Query Execution – Columnar storage allows for selective column retrieval, reducing I/O overhead.
- Reduced Storage Costs – High compression rates minimize storage consumption.
- Efficient Data Scanning – Queries read only relevant columns instead of entire rows, optimizing performance.

Delta Lake further enhances Parquet by introducing Delta Logs, which track metadata and data changes efficiently, reducing redundant data scans.

2.Efficient Data Indexing and Caching

Delta Lake improves query performance through advanced indexing and caching mechanisms:

- Z-Order Indexing – Organizes data based on frequently queried columns, reducing scan times for large datasets.
- Delta Caching – Speeds up data access by keeping frequently used data in memory, reducing read latency.
- Auto-Optimized Reads – Delta Lake automatically applies optimizations like skipping unnecessary data files, accelerating queries.

These optimizations significantly boost ETL performance by minimizing data movement and processing times.

3.Data Consistency and Reliability

Traditional data lakes often suffer from data corruption due to concurrent writes, partial updates, or system failures. Delta Lake solves this issue by implementing ACID (Atomicity, Consistency, Isolation, Durability) transactions, ensuring:

- No Partial Updates – Either all changes are committed, or none are applied, preventing incomplete data loads.
- Safe Concurrent Writes – Multiple ETL jobs can write to the same dataset without corrupting data.
- Guaranteed Data Integrity – Ensures that data is always in a consistent state, even in high-concurrency environments.

4.Schema Evolution for Flexible Data Handling

Delta Lake allows ETL pipelines to evolve without breaking existing data models:

- Automatic Schema Evolution – New columns can be added dynamically, reducing manual intervention.
- Strict Schema Enforcement – Prevents data inconsistencies by rejecting incompatible writes.
- Backward Compatibility – Supports older data formats while accommodating schema updates.

These features enable organizations to handle changing data requirements efficiently without disrupting existing pipelines.

5.Scalability

As data volumes grow, ETL pipelines must scale efficiently. Delta Lake is designed to support large-scale data processing by:

- Optimizing Distributed Processing – Built on Apache Spark, it enables parallel data processing across multiple nodes.
- Efficient Data Compaction (OPTIMIZE Command) – Merges small files into larger ones to reduce fragmentation and improve read performance.
- Incremental Data Processing – Supports upserts (MERGE INTO) and deletes, enabling real-time data updates without full table refreshes.

6.Cost Optimization

Storage and compute costs are major concerns in ETL pipelines. Delta Lake reduces expenses by:

- Reducing Data Duplication – ACID transactions eliminate the need for multiple copies of the same data.
- Minimizing Storage Overhead – Efficient compression and file organization lower cloud storage costs.
- Accelerating Query Performance – Faster queries reduce compute resource consumption, optimizing cloud billing.

IV. IMPLEMENTING DELTA LAKE IN ETL PIPELINES

Integrating Delta Lake into ETL workflows enhances data reliability, performance, and scalability. This section explores how Delta Lake integrates with popular ETL tools, provides a step-by-step guide to building an optimized ETL pipeline, and highlights key performance optimization techniques.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Integration with ETL Tools and Frameworks

Delta Lake is designed to work seamlessly with big data and cloud platforms, making it a flexible solution for modern ETL processes. It integrates with:

- Apache Spark – Delta Lake is built on top of Spark, allowing direct interaction using PySpark, Scala, or SQL.
- Databricks – Native support in Databricks simplifies Delta Lake implementation with built-in optimizations.
- Cloud Storage – Works with AWS S3, Azure Data Lake Storage (ADLS), and Google Cloud Storage (GCS), making it ideal for cloud-based ETL.
- ETL Tools – Compatible with tools like Apache NiFi, Apache Airflow, and Informatica for orchestrating data pipelines.

Delta Lake's broad compatibility enables organizations to enhance existing ETL workflows without major architectural changes.

Step-by-Step Guide to Building an Optimized ETL Pipeline

1. Data Ingestion into Delta Lake

The first step in an ETL pipeline is extracting and loading data into Delta Lake. Delta Lake supports multiple ingestion methods:

- Batch Processing: Load large datasets from relational databases, CSV, JSON, or Parquet files.
- Streaming Ingestion: Use Apache Spark Structured Streaming to continuously ingest real-time data.
- Upserts and Deletes: Merge new records without duplicating data using the MERGE INTO command.

2. Applying Transformations and Handling Schema Evolution

Once data is ingested, transformations such as cleaning, filtering, and aggregations are performed. Delta Lake provides:
Schema Evolution – Supports changes to table structure without breaking existing queries.
Data Deduplication – Ensures data integrity by preventing duplicate records.
Efficient Updates and Deletes – Uses ACID transactions to modify data reliably.

3. Optimizing Performance with Partitioning and Indexing

Delta Lake offers several techniques to enhance query speed and reduce processing time:

- Partitioning – Organizes data into logical groups based on key columns (e.g., date, region).
- Z-Order Indexing – Reorders data to improve filtering performance on frequently queried columns.
- Compaction (OPTIMIZE Command) – Merges small files into larger ones to speed up reads.

V. BEST PRACTICES FOR MAXIMIZING ETL EFFICIENCY WITH DELTA LAKE

To ensure optimal ETL performance with Delta Lake, organizations should follow best practices that improve query speed, storage efficiency, and overall data reliability. Key techniques include data partitioning and Z-order indexing, Delta Caching for faster queries, and automating data compaction and vacuuming to optimize storage and reduce costs.

1.Data Partitioning

Partitioning organizes data into smaller, manageable subsets based on specific columns (e.g., date, region, or category). This improves query performance by reducing the amount of data scanned.
Best Practices for Partitioning:

- Choose a high-cardinality column (e.g., event_date) to ensure partitions are evenly distributed.
- Avoid over-partitioning, which can create too many small files and degrade performance.
- Use partition pruning in queries to limit the scanned partitions.

2.Z-Order Indexing

Z-order indexing optimizes how data is stored within each partition, improving filtering performance on frequently queried columns. Instead of scanning entire partitions, Z-ordering groups related data together, reducing scan times.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

3.Using Delta Caching for Faster Queries

Delta Caching stores frequently accessed data in memory, reducing read latency and improving query response times. This is especially useful for iterative queries or dashboarding scenarios.

Benefits of Delta Caching:

- Reduces disk I/O by keeping hot data in memory.
- Speeds up repetitive queries on the same dataset.
- Enhances performance in interactive analytics workloads.

4. Data Compaction

Delta Lake writes data incrementally, which can lead to small files accumulating over time. Compaction merges smaller files into larger ones, improving read performance and reducing metadata overhead. Schedule compaction jobs regularly (e.g., daily or weekly) for frequently updated tables.

5.Vacuuming (Removing Old Data Versions)

Delta Lake retains historical versions of data for time travel and rollback. However, excessive retention can lead to unnecessary storage costs. Running the VACUUM command removes outdated data files.

VI. CASE STUDIES AND REAL-WORLD APPLICATIONS

Many organizations have successfully implemented Delta Lake to enhance their ETL efficiency, data consistency, and query performance. This section explores real-world use cases, performance benchmarks, and a comparative analysis of Delta Lake against traditional data lake architectures.

1. Financial Services: Faster Data Processing and Regulatory Compliance

Challenge: A large financial institution struggled with slow ETL processing and inconsistent data across multiple data sources, leading to compliance risks.

Solution with Delta Lake:

- Implemented ACID transactions to ensure consistent and accurate financial records.
- Used schema enforcement to maintain data integrity across different reporting systems.
- Optimized performance with Z-order indexing for frequently queried customer transaction records.

Outcome:

- 40% faster ETL processing time, reducing daily batch processing from 6 hours to 3.5 hours.
- Improved data accuracy, eliminating discrepancies in regulatory reports.
- Achieved real-time data updates, enhancing customer insights.

2. E-Commerce: Real-Time Customer Analytics and Fraud Detection

Challenge: An e-commerce platform experienced high latency in data queries and delayed fraud detection due to slow ETL pipelines running on a traditional data lake.

Solution with Delta Lake:

- Replaced batch processing with structured streaming to enable real-time fraud detection.
- Used Delta Caching to reduce query response times for customer purchase history.
- Enabled time travel to analyze historical transaction data for fraud pattern detection.

Outcome:

- Reduced query execution time by 60%, improving real-time decision-making.
- Enabled instant fraud alerts, reducing false positives by 30%.
- Enhanced customer experience by providing real-time personalized recommendations.

3. Healthcare: Scalable and Compliant ETL for Patient Data

Challenge: A healthcare provider faced data inconsistencies across multiple data sources, impacting patient record accuracy and slowing down research analytics.

Solution with Delta Lake:

- Implemented schema evolution to accommodate new medical attributes dynamically.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- Used Delta Lake's ACID transactions to maintain a single source of truth for patient data.
- Optimized storage costs with automated data compaction and retention policies.

Outcome:

- Achieved 99.9% data accuracy across all medical records.
- Reduced ETL execution time by 50%, enabling faster clinical insights.
- Met regulatory compliance for data security and auditing (HIPAA).

VII. CONCLUSION

In today's data-driven world, efficient ETL (Extract, Transform, Load) processes are critical for ensuring timely and accurate analytics. However, traditional ETL systems often face challenges such as slow performance, data inconsistency, scalability issues, and high costs. Delta Lake, a robust solution built on Apache Spark, addresses these problems by optimizing data storage, ensuring data integrity, and enabling scalable processing.

By utilizing Delta Lake's advanced features—such as ACID transactions, schema enforcement, time travel, and Z-order indexing—organizations can significantly improve the efficiency and reliability of their ETL pipelines. Delta Lake enhances ETL performance by providing optimized storage with the Parquet format, ensuring data consistency through transaction logs, and enabling real-time data updates with structured streaming capabilities. Furthermore, its ability to handle large-scale data processing workloads and optimize storage costs through compaction and vacuuming makes it a powerful solution for managing big data.

Through case studies, we've seen organizations in industries like finance, e-commerce, and healthcare achieve remarkable results by adopting Delta Lake. These include faster ETL processing times, improved query performance, and enhanced data quality. Performance benchmarks also demonstrate that Delta Lake outperforms traditional data lake architectures, providing up to 10x faster query execution and 50-60% faster ETL processing.

Delta Lake's integration with popular ETL tools and frameworks, like Apache Spark and Databricks, allows organizations to leverage their existing tech stack while taking advantage of Delta Lake's advanced features. Whether processing batch or streaming data, Delta Lake ensures optimized workflows, real-time data access, and scalable analytics for modern data architectures.

In conclusion, Delta Lake is an essential solution for businesses looking to optimize their ETL processes, improve data quality, and unlock the full potential of their data for analytics and decision-making.

REFERENCES

1. Koppula, Ravi Shankar. "Implementing Data Lakes with Databricks for Advanced Analytics." North American Journal of Engineering Research 3, no. 2 (2022).
2. Pandey, Piyush. "Analyze and Optimize Data Pipelines with Effective Data Models."
3. Kamalakkannan, S., A. Yasmin, and P. Kavitha. "A Model for the Analytical Performance of Data Lake in Stock Market Analysis with Databricks Delta Lake." In 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), pp. 1065-1071. IEEE, 2023.Kamalakkannan, S., A. Yasmin, and P. Kavitha. "A Model for the Analytical Performance of Data Lake in Stock Market Analysis with Databricks Delta Lake." In 2023 International Conference on Self Sustainable Artificial Intelligence Systems (ICSSAS), pp. 1065-1071. IEEE, 2023.
4. Seenivasan, Dhamotharan. "ETL in a World of Unstructured Data: Advanced Techniques for Data Integration." Journal Homepage: http://www. ijmra. us.() (2021).
5. Armbrust, Michael, Tathagata Das, Liwen Sun, Burak Yavuz, Shixiong Zhu, Mukul Murthy, Joseph Torres et al. "Delta lake: high-performance ACID table storage over cloud object stores." Proceedings of the VLDB Endowment 13, no. 12 (2020): 3411-3424.
6. Yasmin, A., and S. Kamalakkannan. "Analytical Performance in Data Lake Storage of Big Data Analytics by Databricks Delta Lake for Stock Market Analysis." In Computer Networks and Inventive Communication Technologies: Proceedings of Fifth ICCNCT 2022, pp. 213-226. Singapore: Springer Nature Singapore, 2022.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

7. Simitsis, Alkis, Spiros Skiadopoulos, and Panos Vassiliadis. "The history, present, and future of ETL technology." In DOLAP, pp. 3-12. 2023.
8. Saddad, Emad, Ali El-Bastawissy, Hoda MO Mokhtar, and Maryam Hazman. "Lake data warehouse architecture for big data solutions." International Journal of Advanced Computer Science and Applications 11, no. 8 (2020): 417-424.
9. Seenivasan, Dhamotharan. "Distributed ETL Architecture for Processing and Storing Big Data." (2022).
10. Lekkala, Chandrakanth. "Building Resilient Big Data Pipelines with Delta Lake for Improved Data Governance." European Journal of Advances in Engineering and Technology 7, no. 12 (2020): 101-106.
11. Mirmoeini, SeyedFarzam. "Karavan, ETL pipeline management system based on Apache Spark." PhD diss., ETSI_Informatica, 2021.
12. Van der Putten, Chiara. "Transforming data flow: Generative AI in ETL pipeline automatization." PhD diss., Politecnico di Torino, 2024.
13. Hohensinn, David. "Spooq: A Software Library for ETL Processes in Data Lakes/Author David Hohensinn, BA." (2021).
14. Seenivasan, Dhamotharan. "ETL (extract, transform, load) best practices." International Journal of Computer Trends and Technology 71, no. 1 (2023): 40-44.
15. Akhund, Sadig. "Computing Infrastructure and Data Pipeline for Enterprise-scale Data Preparation."
16. Mazumdar, Dipankar, Jason Hughes, and J. B. Onofre. "The data lakehouse: Data warehousing and more." arXiv preprint arXiv:2310.08697 (2023).
17. Biswas, Neepa. "Modeling, analysis and simulation of near real-time ETL processes of big data in cloud." (2022).
18. Gopalan, Rukmani. The Cloud Data Lake: A Guide to Building Robust Cloud Data Architecture. " O'Reilly Media, Inc.", 2022.
19. Seenivasan, Dhamotharan. "Improving the Performance of the ETL Jobs." International Journal of Computer Trends and Technology 71, no. 3 (2023): 27-33.
20. Gueddoudj, El Yazid, Azeddine Chikh, and Abdelouahab Attia. "Os-ETL: A High-Efficiency, Open-Scala Solution for Integrating Heterogeneous Data in Large-Scale Data Warehousing." Ingénierie des Systèmes d'Information 28, no. 3 (2023).
21. Hai, Rihan. "Data integration and metadata management in data lakes." PhD diss., Dissertation, RWTH Aachen University, 2020, 2020.
22. Pohl, Matthias, Nathira Dharindri Wijemanne, Daniel Staegemann, Christian Haertel, Christian Daase, Dirk Dreschel, Damanpreet Singh Walia, Arne Osterthun, Joshua Reibert, and Klaus Turowski. "Data Lakehouse for Time Series Data: A Systematic Literature Review." In 2024 IEEE International Conference on Big Data (BigData), pp. 5833-5842. IEEE, 2024.
23. Jephte, Ioudom Foubi. "Extract, Transform, and Load data from Legacy Systems to Azure Cloud." Master's thesis, Universidade NOVA de Lisboa (Portugal), 2021.
24. Manchana, R. "Building a Modern Data Foundation in the Cloud: Data Lakes and Data Lakehouses as Key Enablers." J Artif Intell Mach Learn & Data Sci 1, no. 1 (2023): 1098-1108.
25. Kibugu, Anne. "A Methodology for the Implementation of a Data Warehouse Using an Etl Process Model for Improved Decision Support." PhD diss., University of Nairobi, 2016.
26. Ahmadi, Sina. "Optimizing data warehousing performance through machine learning algorithms in the cloud." International Journal of Science and Research 12, no. 12 (2023): 1859-1867.
27. Seenivasan, Dhamotharan. "Real-time data processing with streaming ETL." International Journal of Science and Research 12, no. 11 (2023): 1-10.
28. Gupta, Nikhil, and Jason Yip. "Delta Lake-Deep Dive." In Databricks Data Intelligence Platform: Unlocking the GenAI Revolution, pp. 61-88. Berkeley, CA: Apress, 2024.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details