

Systematic Literature Review: Modern Data Stack Architecture Layers and Technologies for Data Warehousing

Nanang Fajar Untoro^{1*}, Farisa Aulia Safira², Natia Seanesya Kencana³, Rizky Muflih⁴, Vitri Tundjungsari⁵

^{1*,2,3,4,5}Magister Ilmu Komputer, Universitas Esa Unggul, DKI Jakarta, Indonesia

e-mail: nanangfajaru@gmail.com^{1*}, ichafa@gmail.com², natsea0815@gmail.com³,

rizkymuflih10@gmail.com⁴, vitri.tundjungsari@esaunggul.ac.id⁵

Article Information

Article History:

Received : 16 June 2026
Revised : 20 October 2026
Accepted : 22 October 2026
Published : 29 October 2026

*Correspondence:

nanangfajaru@gmail.com

Keywords:

Modern Data Stack, Data Warehouse, Cloud Architecture Layers, Systematic Literature Review

Copyright © 2026 by Author.

Published by Universitas Dinamika.



This is an open access article under the [CC BY-SA](#) license.



10.37802/joti.v8i2.1460

Journal of Technology and Informatics (JoTI)

P-ISSN 2721-4842

E-ISSN 2686-6102

[https://e-](https://e-journals.dinamika.ac.id/index.php/joti)

[journals.dinamika.ac.id/index.php/joti](https://e-journals.dinamika.ac.id/index.php/joti)

Abstract:

This systematic literature review (SLR) examines modern data stack architectures for data warehousing, focusing on their architectural layers and the tools used within each layer. Following SLR procedures, scholarly articles published between 2020 and 2025 were retrieved from IEEE Xplore and Google Scholar and screened using predefined inclusion and exclusion criteria, resulting in 35 primary studies. The review addresses three research questions: (1) which architectural layers predominate in modern data stack frameworks, (2) which tools and technologies are commonly applied within each layer, and (3) which trends shape the evolution of these architectures. The findings show a clear convergence toward modular, cloud-native architectures consisting of six layers: data ingestion, storage, transformation, orchestration, serving, and governance. Four major trends were identified: the shift from Extract, Transform, Load (ETL) to Extract, Load, Transform (ELT), the emergence of lakehouse architectures, the growing adoption of managed and serverless services, and the expansion of real-time analytics. This study contributes a layer-by-layer technology mapping that offers a structured reference for practitioners, researchers, and organizations in evaluating and designing cloud-based data warehouse platforms.

INTRODUCTION

1. Background

Data warehousing has changed substantially in recent years, driven by the adoption of cloud computing, advances in big data technologies, and the growing demand for real-time analytics that support operational efficiency and strategic decision-making [1]. Traditional data warehouses were monolithic, relied on tightly coupled Extract, Transform, Load (ETL) processes,

and ran on on-premises infrastructure; they are increasingly being replaced by modular data stack architectures that prioritize flexibility, scalability, and cloud-native design [2].

Modern data stack architectures represent a paradigm shift in how organizations design, build, and maintain their data infrastructure. They combine cloud platforms, managed services, and open-source tools into composable pipelines that can integrate diverse data sources, serve multiple analytical workloads, and scale elastically with business demand [3]. The decoupling of storage and compute, a foundational principle of these architectures, allows organizations to optimize operational costs while maintaining high performance [4].

Several prior studies have examined related topics. A systematic review of distributed data warehouse architectures synthesized 29 studies under the PRISMA 2020 guidelines and organized its findings around four governance-oriented themes, namely security, federation and data mesh, lakehouse adoption, and real-time streaming, underscoring organizational autonomy and compliance as the central concerns of distributed warehousing research [5]. Database design studies, including work on physical and conceptual data modeling for institutional information systems [6], show that structuring data persistence remains an active area of inquiry even as the surrounding architecture shifts toward cloud-native paradigms. Studies on big data adoption in specific domains, such as Islamic (sharia) accounting [7], likewise rely on structured literature searches to map fragmented bodies of knowledge into coherent themes, reinforcing the value of systematic reviews for fast-moving technology domains. However, none of these studies provides a layer-by-layer technology mapping from ingestion to governance specifically for data warehousing; most focus on a single concern, such as governance, security, or a specific industry, or remain at a conceptual level without consolidating the tool ecosystem. This gap motivates the present review.

2. Problem Statement

Despite the widespread adoption of modern data stack architectures, there is still a lack of literature that systematically examines their architectural layers and the tools associated with each layer. Organizations migrating from legacy systems to modern architectures need evidence-based guidance on both architectural patterns and tool selection [8]. Furthermore, the rapid pace of innovation in the data engineering ecosystem calls for periodic reviews that capture emerging trends, best practices, and lessons learned from implementation case studies.

3. Research Questions

This systematic literature review addresses the following research questions (RQ):

- a) RQ1: Which architectural layers are predominantly employed in modern data stack frameworks for data warehousing?
- b) RQ2: Which tools and technologies are commonly utilized within each architectural layer?
- c) RQ3: Which primary trends characterize the evolution of present-day data stack architectures?

METHOD

1. Search Strategy

A systematic search was conducted in academic databases and digital libraries, namely IEEE Xplore and Google Scholar, to identify peer-reviewed articles and relevant publications on modern data stack architectures for cloud-based data warehousing.

2. Search Terms

The following search terms and Boolean combinations were used:

- a) "modern data stack" AND "cloud architecture"
- b) "data warehouse" AND "cloud" AND "architecture layers"
- c) "data lakehouse" AND "data platform"
- d) "tools" AND "platform"

3. Criteria for Inclusion and Exclusion

The inclusion and exclusion criteria applied during study selection are presented in Table 1.

Table 1. Criteria for Inclusion and Exclusion

Inclusion	Exclusion
1) Published from 2020 to 2025	1) Published before 2020
2) Focus on data warehouse architectures or modern data stack components	2) Purely theoretical without architectural context
3) Discusses architectural layers, design patterns, or tool ecosystems	3) Focus solely on algorithms without architectural discussion
	4) Duplicate publications

4. Data Extraction

The following information was extracted from each selected article:

- 1) Architectural layers: the layers of the data stack examined in the study.
- 2) Tools and technologies: the tools and technologies used or recommended in each layer.
- 3) Trends and future directions: emerging patterns and potential avenues for future research.

5. Selection Process

The study selection followed a multi-stage process, as illustrated in Figure 1:

1. Initial search: 240 papers were identified across all databases.
2. Title and abstract screening: 96 papers were retained after removing duplicates and irrelevant studies.
3. Full-text assessment: 70 papers met the publication year criterion (2020–2025).
4. Final selection: the 35 most relevant papers were selected based on relevance ranking and alignment with the research questions.

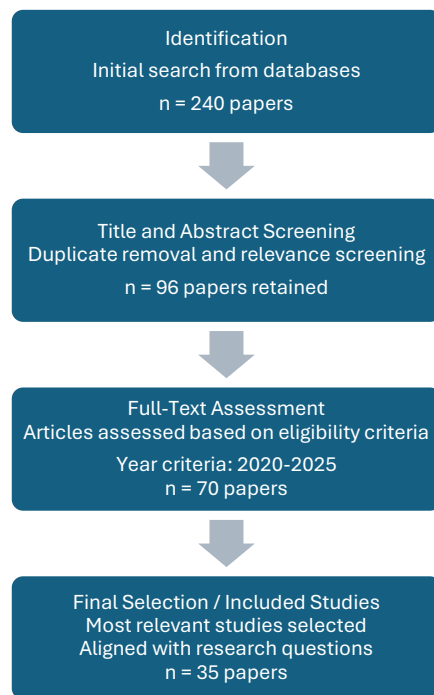


Figure 1. Study Selection Process

6. Quality Assessment

Each of the 35 selected studies was appraised using a simplified quality checklist adapted from common SLR practice, covering five criteria:

- 1) Methodological rigor: strength of the research design and methods used.
- 2) Clarity of architecture: how clearly the system or data stack is described.
- 3) Practical implementation: whether the study includes real-world use or validation.
- 4) Citation and relevance: level of citations and relevance to the topic.
- 5) Relevance to research questions: how well the study supports the research objectives.

The distribution of the selected studies by publication year is shown in Figure 2. Most studies were published in 2024 and 2025, reflecting the growing research interest in modern data stack architectures.

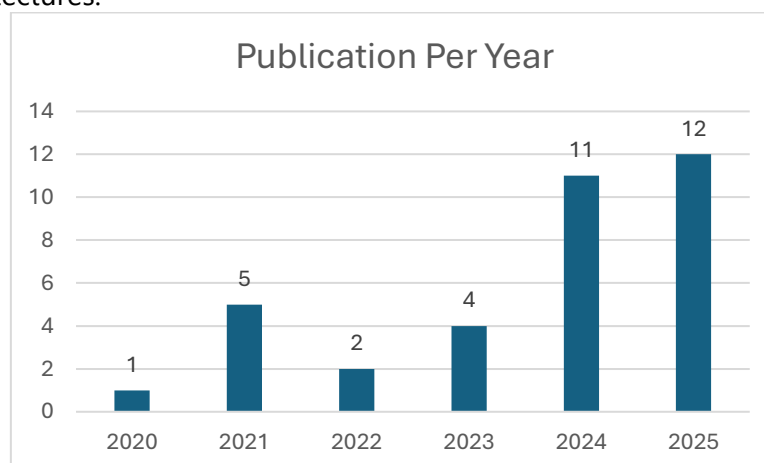


Figure 2. Publication Count per Year

RESULTS AND DISCUSSION

1. Overview of Modern Data Stack Architecture

Modern data stack architectures converge on a modular, layered framework that separates concerns and allows each component to scale independently. The literature consistently identifies six fundamental layers that form the foundation of modern data warehousing implementations [1], [2], [3].

The core design principle of these architectures is the separation of storage and compute, which allows organizations to scale resources independently and improve cost efficiency [4]. Enabled by cloud object storage and managed compute services, this separation departs from conventional monolithic data warehouses, in which storage and processing were tightly coupled.

2. Architectural Layers in Modern Data Stacks

Based on the systematic analysis, six layers emerged as standard components of modern data stack architectures, as shown in Figure 3.

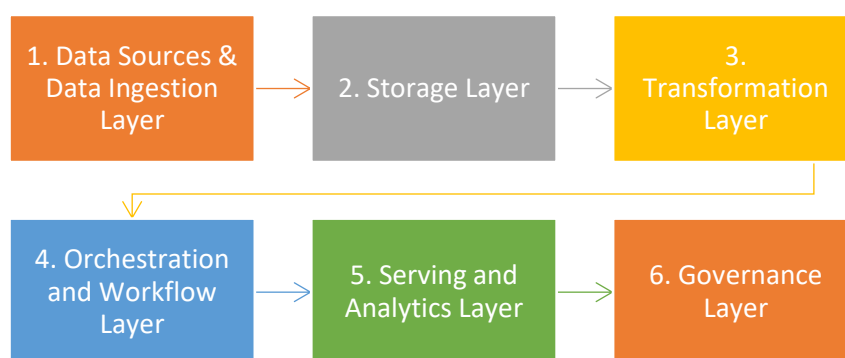


Figure 3. Architectural Layers of the Modern Data Stack

A. Data Sources and Ingestion Layer

The ingestion layer captures data from operational systems, databases, Software as a Service (SaaS) applications, and streaming sources. Modern frameworks support both batch and real-time ingestion, enabling more efficient handling of diverse data streams [9]. Its key characteristics include support for diverse source types (databases, APIs, files, and streams), Change Data Capture (CDC) for incremental updates, schema detection and evolution management, and a connector-based architecture that improves extensibility.

B. Storage Layer

The storage layer has evolved significantly, with three primary patterns emerging from the literature:

- 1) Data lake pattern: raw data is stored in cloud object storage (S3, Azure Blob, GCS) using open formats (Parquet, Avro, ORC) [10]. Data lakes provide a cost-effective way to accumulate large volumes of structured, semi-structured, and unstructured data.
- 2) Cloud data warehouse pattern: managed services (Snowflake, BigQuery, Redshift) provide SQL interfaces, ACID transactions, and optimized query performance [11], offering the reliability required for complex analytical workloads.
- 3) Lakehouse pattern: a hybrid architecture that combines the flexibility of data lakes with the reliability of warehouses through open table formats (Delta Lake, Apache Iceberg, Apache

Hudi) [12], [13]. It has gained attention as enterprises seek to consolidate analytics and machine learning workloads on a single platform.

The literature further indicates that lakehouse architectures address the limitations of operating separate lake and warehouse systems [12], that open table formats enable ACID transactions on data lakes [13], and that decoupled storage allows multiple compute engines to access the same data [14].

C. Transformation Layer

The transformation layer implements business logic, data cleansing, and analytical model creation. Modern architectures increasingly prefer the Extract, Load, Transform (ELT) approach over conventional ETL [15]. ELT leverages warehouse compute power for transformations, retains raw data for reprocessing and auditing, simplifies pipelines by eliminating intermediate transformation servers, and enables version control and testing of transformation logic. Common transformation patterns include SQL-based transformations executed within the warehouse [16], modular and dependency-aware transformation workflows [17], and incremental processing for efficiency [18].

D. Orchestration and Workflow Layer

Orchestration coordinates pipeline execution, manages dependencies, and handles scheduling across the data stack [19]. Its key requirements are dependency management between pipeline stages, retry and error-handling mechanisms, monitoring and alerting, and support for complex, branching workflows. Common orchestration patterns include Directed Acyclic Graphs (DAGs) for workflow definition [19], event-driven orchestration for real-time pipelines [20], and metadata-driven pipeline generation [21].

E. Serving and Analytics Layer

The serving layer provides interfaces for data consumers, including business intelligence (BI) tools, data science platforms, and operational applications [22]. Its components include BI and visualization platforms, SQL query interfaces and semantic layers, machine learning feature stores, and reverse ETL for operational analytics [23]. Key trends in this layer are the emphasis on self-service analytics [24], embedded analytics in operational applications [25], and real-time dashboards for operational intelligence [26].

F. Governance, Metadata, and Observability Layer

This cross-cutting layer addresses data quality, lineage, cataloging, and compliance requirements [27], [28]. Its key components are metadata catalogs for data discovery, lineage tracking across pipelines, quality monitoring and anomaly detection, and access control and compliance frameworks. Emerging priorities include data observability platforms for proactive monitoring [29], automated data quality checks [30], and privacy and compliance automation [28].

3. Tools and Technologies by Layer

Table 2 summarizes the tools and technologies most frequently discussed in the reviewed literature, organized by architectural layer.

Table 2. Tools and Technologies by Architectural Layer

Layer	Category	Tools/Technologies	Paper References
Ingestion	Managed Connectors	Fivetran, Stitch, Airbyte	[8], [31], [32]
	Custom Integration	API ingestion, CDC tools	[9], [33]

Layer	Category	Tools/Technologies	Paper References
Storage	Streaming	Kafka, Kinesis, Pub/Sub	[20], [26]
	Cloud Warehouses	Snowflake, BigQuery, Redshift, Synapse	[2], [11], [34]
	Data Lakes	S3, Azure Data Lake, GCS	[10], [14]
Transformation	Lakehouse Formats	Delta Lake, Iceberg, Hudi	[12], [13]
	ELT Frameworks	dbt (data build tool)	[15], [16], [17]
	Processing Engines	Spark, Databricks	[4], [35]
	In-Warehouse SQL	Native SQL transformations	[16], [36]
Orchestration	Workflow Engines	Apache Airflow, Prefect, Dagster	[19]
	Managed Services	Azure Data Factory, AWS Glue	[8], [37]
Serving	Business Intelligence	Tableau, Power BI, Looker	[22], [24]
	Query Engines	Presto, Trino, Athena	[14], [38]
Governance	Catalogs	Alation, Collibra, DataHub	[27]
	Observability	Monte Carlo, Datadog	[29]

Table 2 shows that the storage and transformation layers have the most mature tool ecosystems, dominated by cloud data warehouses, open table formats, and dbt-based ELT, whereas the governance layer is supported by comparatively few studies, indicating an area that requires further research.

4. Key Trends in Modern Data Stack Architectures

A. Shift from ETL to ELT Paradigm

The literature documents a clear move from traditional ETL toward ELT, in which raw data is loaded directly into the warehouse before transformation [15]. Enabled by the computational power of cloud warehouses, this shift simplifies the architecture by eliminating intermediate transformation servers, preserves raw data for reprocessing, and allows transformations to be modified without re-ingesting data.

B. Lakehouse Architecture Emergence

The lakehouse paradigm, which combines the flexibility of data lakes with the reliability of data warehouses, is a notable architectural advancement [12], [13]. It addresses the “two-platform dilemma” in which organizations maintain separate systems for structured analytics (data warehouses) and unstructured data processing (data lakes). Its characteristics include ACID transactions on data lakes via open table formats, support for diverse data types and workloads, a unified governance and security model, and cost optimization through tiered storage. Armbrust et al. [12] and Park and Kim [4] provide detailed analyses of lakehouse architectures and their benefits for specific use cases.

C. Managed and Serverless Services Adoption

The literature shows increasing adoption of managed and serverless data services that reduce the operational burden [8]. Organizations are shifting from self-managed infrastructure to cloud-native services that automatically handle provisioning, scaling, and maintenance. The main advantages are reduced operational complexity, automated scaling and performance

tuning, usage-based pricing, and faster time to value. Bukhari et al. [2] document migration strategies from legacy systems to cloud-native managed services and highlight both their benefits and challenges.

D. Real-Time and Streaming Analytics

Growing demand for operational intelligence drives the adoption of real-time data pipelines and streaming architectures [20], [26]. Modern stacks increasingly support both batch and streaming workloads on unified platforms, and big data analytics and visualization platforms play a key role in improving decision- and policy-making [39]. Common real-time patterns include Lambda and Kappa architectures for combined batch and streaming processing, CDC for near-real-time updates, and stream processing frameworks integrated with warehouses.

5. Theoretical and Practical Implications

Theoretically, this review contributes a layer-based framework that complements prior theme-based studies on distributed data warehousing [5]. Whereas [5] organizes its findings around governance concerns (security, federation, lakehouse adoption, and streaming), this study structures the same domain according to functional pipeline layers (ingestion, storage, transformation, orchestration, serving, and governance). The two perspectives are complementary, suggesting that future research may benefit from a two-dimensional model that combines architectural layers with governance concerns.

Practically, the layer-by-layer tool mapping provides a reference for organizations evaluating build-versus-buy decisions across the data stack. It also shows that architectural choices such as ELT, lakehouse adoption, managed services, and streaming should not be adopted merely because they are trends; they must be weighed against cost, governance, and vendor lock-in. For researchers, this study provides a more granular, layer-based technology mapping that extends prior governance-oriented reviews and offers an updated synthesis of modern data stack architectures.

CONCLUSIONS AND SUGGESTIONS

This systematic literature review examined modern data stack architectures for data warehousing based on 35 primary studies published between 2020 and 2025. Regarding RQ1, modern data stack architectures converge on six core layers: data sources and ingestion, storage (lake, warehouse, or lakehouse), transformation, orchestration and workflow, serving and analytics, and governance and observability. Together, these layers form a modular paradigm that supports independent scaling, technology substitution, and a clear separation of responsibilities. Regarding RQ2, each layer is supported by a distinct tool ecosystem: managed connectors (Fivetran, Stitch, Airbyte) and streaming platforms (Kafka, Kinesis) for ingestion; cloud data warehouses (Snowflake, BigQuery, Redshift), data lakes, and open table formats (Delta Lake, Iceberg, Hudi) for storage; dbt, Spark, and Databricks for transformation; Airflow, Prefect, Dagster, Azure Data Factory, and AWS Glue for orchestration; BI platforms (Tableau, Power BI, Looker) and query engines (Presto, Trino) for serving; and data catalogs (Alation, Collibra, DataHub) and observability tools (Monte Carlo, Datadog) for governance. Regarding RQ3, four major trends shape these architectures: the shift from ETL to ELT, the emergence of lakehouse architectures, the growing adoption of managed and serverless services, and the integration of real-time streaming analytics through CDC pipelines and Lambda or Kappa patterns.

This study has several limitations. The search was limited to IEEE Xplore and Google

Scholar, and the final selection relied on relevance ranking, so that some relevant studies may have been missed. In addition, several of the reviewed sources are practitioner-oriented rather than empirical evaluations. Future research should extend the search to other databases such as Scopus, ACM Digital Library, and ScienceDirect, empirically compare tool combinations in each layer in terms of cost, performance, and governance, and develop a two-dimensional framework that integrates architectural layers with governance concerns.

REFERENCES

- [1] A. Eweje and F. Ohaegbu, "Advances in Modern Data Stack Architectures for Scalable Data Integration and Business Intelligence," *International Journal of Multidisciplinary Research and Growth Evaluation*, vol. 2, no. 5, pp. 538–550, 2021. DOI: 10.54660/ijmrge.2021.2.5.538-550
- [2] T. T. Bukhari, O. Oladimeji, E. D. Etim, and J. O. Ajayi, "Cloud-Native Business Intelligence Transformation: Migrating Legacy Systems to Modern Analytics Stacks for Scalable Decision-Making," *International Journal of Scientific Research in Humanities and Social Sciences*, vol. 1, no. 2, pp. 744–762, 2024. DOI: 10.32628/ijrsssh242763
- [3] K. M. Reddy, "Cloud Data Warehousing: A Transformative Technology Across Industries - A Comprehensive Analysis of Implementation and Impact," *European Modern Studies Journal*, vol. 9, no. 4, pp. 508–514, 2025. DOI: 10.59573/emsj.9(4).2025.49
- [4] S. K. Park and J. Kim, "Design of Vessel Data Lakehouse with Big Data and AI Analysis Technology for Vessel Monitoring System," *Electronics*, vol. 12, no. 8, p. 1943, 2023. DOI: 10.3390/electronics12081943
- [5] L. AlOud and O. Alharbi, "A Systematic Literature Review of Distributed Data Warehouse Architectures," *International Journal of Technology Innovation and Management (IJTIM)*, vol. 5, no. 1, 2025. DOI: 10.54489/ijtim.v5i2.567
- [6] I. R. Mukhlis and R. Santoso, "Perancangan Basis Data Perpustakaan Universitas Menggunakan MySQL dengan Physical Data Model dan Entity Relationship Diagram," *Journal of Technology and Informatics (JoTI)*, vol. 4, no. 2, pp. 81–87, 2023. DOI: 10.37802/joti.v4i2.330
- [7] A. R. Aulia, S. Saputra, A. D. Titandy, M. Zacky, and A. Arwani, "Pengaruh Big Data dalam Akuntansi Syariah: Analisis Prediktif dan Pengambilan Keputusan Strategis," *Journal of Technology and Informatics (JoTI)*, vol. 6, no. 1, pp. 17–21, 2024. DOI: 10.37802/joti.v6i1.704
- [8] D. Anoshin, D. Foshin, R. Storchak, and X. Ireton, *Azure Data Factory Cookbook: Build and Manage ETL and ELT Pipelines with Microsoft Azure's Serverless Data Integration Service*. Birmingham, UK: Packt Publishing, 2020.
- [9] B. Guntupalli, "The Evolution of ETL: From Informatica to Modern Cloud Tools," *International Journal of AI, BigData, Computational and Management Studies*, vol. 2, no. 2, pp. 66–75, 2021. DOI: 10.63282/3050-9416.ijaibdcms-v2i2p108
- [10] P. Chauhan and M. Sood, "Big Data: Present and Future," *IEEE Computer*, vol. 54, no. 4, pp. 59–65, Apr. 2021. DOI: 10.1109/MC.2021.3057442
- [11] S. K. Somayajula, "Demystifying Modern Data Warehousing: From Traditional to Cloud-Native Solutions," *Journal of Cloud Data Management*, vol. 11, no. 1, pp. 348–362, 2025. DOI: 10.32628/CSEIT25111235
- [12] M. Armbrust, A. Ghodsi, R. Xin, and M. Zaharia, "Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics," in *Proc. 11th Conference on Innovative Data Systems Research (CIDR)*, 2021. [Online]. Available:

- http://cidrdb.org/cidr2021/papers/cidr2021_paper17.pdf
- [13] R. Thalpati, *Practical Lakehouse Architecture: Designing and Implementing Modern Data Platforms at Scale*. Shelter Island, NY, USA: Manning Publications, 2024.
 - [14] A. V. Chaudhari and P. A. Charate, "Optimizing Data Lakehouse Architectures for Scalable Real-Time Analytics," *International Journal of Scientific Research in Science, Engineering and Technology*, vol. 12, no. 2, pp. 809–822, 2025. DOI: 10.32628/IJSRSET25122198
 - [15] D. Naphade, "The Evolution and Modernization of Data Pipeline Architectures," *European Journal of Computer Science and Information Technology*, vol. 13, no. 6, pp. 42–53, 2025. DOI: 10.37745/ejcsit.2013/vol13n64253
 - [16] D. Foshin, T. Chernyshova, D. Anoshin, and X. Ireton, *Azure Data Factory Cookbook: Build ETL, Hybrid ETL, and ELT Pipelines using ADF, Synapse Analytics, Fabric and Databricks*. Birmingham, UK: Packt Publishing, 2024.
 - [17] B. Mohammed, "A Systematic Taxonomy of ETL Activities for Modern Data Pipelines," *International Journal of Advanced Multidisciplinary Research and Studies*, vol. 5, no. 2, pp. 106–111, 2025. DOI: 10.62225/2583049X.2025.5.2.3816
 - [18] M. V. Talakh, T. O. Ushenko, O. V. Kaduk, and M. Y. Maksimovich, "Balancing Efficiency and Accuracy: Incremental Learning as a Key to Big Data Processing," *Optoelectronic Information-Power Technologies*, vol. 48, no. 2, pp. 45–57, 2024. DOI: 10.31649/1681-7893-2024-48-2-45-57
 - [19] M. Nora, A. Shaikh, and R. Pradhan, "Managing Data Pipeline with Apache Airflow," *International Journal of Advanced Research in Science, Communication and Technology*, vol. 3, no. 2, pp. 244–250, 2023. DOI: 10.48175/IJARST-12134
 - [20] V. N. K. Kundavaram, "Event-Driven Data Pipelines: A Cloud-Based Approach to Real-Time Data Processing," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 10, no. 6, pp. 364–369, 2024. DOI: 10.32628/CSEIT24106183
 - [21] P. K. Vattumilli, "Metadata-Driven ETL Pipelines: A Framework for Scalable Data Integration Architecture," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 10, no. 6, pp. 1799–1807, 2024. DOI: 10.32628/CSEIT241061224
 - [22] D. Bansal, "Enterprise Data Engineering: Architecting Modern Data Warehouses for Business Success," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 11, no. 2, pp. 3266–3277, 2025. DOI: 10.32628/CSEIT251112348
 - [23] B. Dash and S. Swayamsiddha, "Reverse ETL for Improved Scalability, Observability, and Performance of Modern Operational Analytics - A Comparative Review," in *Proc. 2022 OITS International Conference on Information Technology (OCIT)*, 2022, pp. 491–494. DOI: 10.1109/OCIT56763.2022.00097
 - [24] A. Dibouliya and V. Jotwani, "The Transformational Impact of Modern Data Warehousing on the Banking System," *International Journal for Research in Applied Science and Engineering Technology*, vol. 11, no. 6, pp. 1862–1869, 2023. DOI: 10.22214/ijraset.2023.53968
 - [25] P. S. Dhoni, "From Data to Decisions: Enhancing Retail with AI and Machine Learning," *IRE Journals*, vol. 5, no. 1, pp. 38–51, 2024. DOI: 10.47941/ijce.1660
 - [26] J. Qu and J. Wang, "Real-time Data Warehousing in the Big Data Environment: A Comprehensive Review of Implementation in the Internet Industry," *Applied and Computational Engineering*, vol. 88, no. 1, pp. 105–114, 2024. DOI: 10.54254/2755-

- 2721/88/20241643
- [27] J. C. Ogeawuchi, A. C. Uzoka, A. A. Abayomi, and O. A. Agboola, "Innovations in Data Modeling and Transformation for Scalable Business Intelligence on Modern Cloud Platforms," *IRE Journals*, vol. 5, no. 5, pp. 406–431, 2021.
 - [28] V. Garg, "Data Cataloging as a Catalyst for Effective Data Governance and Improved Decision-Making," *International Scientific Journal of Engineering and Management*, vol. 3, no. 12, pp. 1–6, 2024. DOI: 10.55041/ISJEM0387
 - [29] Z. Ali, "AI-Ready Data Infrastructure: A Review of Zero-ETL, Declarative Pipelines, and Data Observability," *Cognizance Journal of Multidisciplinary Studies*, vol. 5, no. 8, pp. 486–507, 2025. DOI: 10.47760/cognizance.2025.v05i08.019
 - [30] G. Govindarajan, "Data Quality and Automation in Modern Data Ecosystems," *International Journal of Advances in Engineering and Management*, vol. 7, no. 4, pp. 228–239, 2025. DOI: 10.35629/5252-0704228239
 - [31] J. Jayabalan and D. Ramachandran, "Optimizing Enterprise Intelligence: A Strategic Framework for Integrating Salesforce with Modern Cloud-Based Data Warehouses for Real-Time Unified Analytics," *International Research Journal on Advanced Engineering Hub*, vol. 3, no. 6, pp. 3010–3016, 2025. DOI: 10.47392/IRJAEH.2025.0443
 - [32] O. F. Enow, A. T. Ofoedu, E. Y. Gbabo, and P. E. Chima, "Advances in Real-Time Data Ingestion Strategies Using Fivetran, Rudderstack, and Open-Source ELT Tools," *Journal of Frontiers in Multidisciplinary Research*, vol. 3, no. 1, pp. 500–516, 2022. DOI: 10.54660/JFMR.2022.3.1.500-516
 - [33] A. Cahyadi and I. G. S. Rahayuda, "Data Processing Using Change Data Capture for Real-Time Data Synchronization," *Jurnal Elektronik Ilmu Komputer Udayana*, vol. 12, no. 4, pp. 743–754, 2024.
 - [34] P. Dubey, "The Data Lakehouse: An Evolving Paradigm in Data Architecture," *International Journal of Computing and Engineering*, vol. 7, no. 10, pp. 30–47, 2025. DOI: 10.47941/ijce.2958
 - [35] L. Alva, "AI-Augmented Real-Time Retail Analytics with Spark and Databricks," *World Journal of Advanced Engineering Technology and Sciences*, vol. 15, no. 2, pp. 1024–1033, 2025. DOI: 10.30574/wjaets.2025.15.2.0631
 - [36] A. M. Kashimkar and Sudepta, "The Role of Data Warehousing in Revolutionizing Banking and User Experience," *International Journal for Multidisciplinary Research*, 2023. DOI: 10.36948/ijfmr.2023.v05i06.8890
 - [37] M. S. Dahibhate, "Enhancing Enterprise Data Orchestration Using Azure Data Factory," *International Scientific Journal of Engineering and Management*, vol. 4, no. 5, pp. 1–9, 2025. DOI: 10.55041/ISJEM03578
 - [38] S. Gourishetti, "Performance Optimization in Distributed SQL Environments: A Comprehensive Analysis of Presto Query Engine," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 10, no. 6, pp. 241–253, 2024. DOI: 10.32628/CSEIT24106173
 - [39] V. Tundjungsari, K. Ernawati, and N. Mutia, "Public Health Surveillance to Promote Clean and Healthy Life Behaviours Using Big Data Approach (An Indonesian Case Study)," in *Cyber-physical Systems and Digital Twins (REV 2019)*, M. Auer and K. Ram B., Eds., *Lecture Notes in Networks and Systems*, vol. 80. Cham, Switzerland: Springer, 2020. DOI: 10.1007/978-3-030-23162-0_69