***IBIMA Publishing***
*Journal of Administrative Sciences and Technology*
*https://ibimapublishing.com/articles/JAST/2026/558858/*
*Vol. 2026 (2026), Article ID 558858, 7 pages, ISSN: 2165-9435*
*https://doi.org/10.5171/2026.558858*

*Research Article*

# From Data Warehouses to the Lakehouse

**Felix ESPINOZA, Lea Nedomova and Milos MARYSKA**

Prague University of Economics and Business, Prague, Czech Republic

Correspondence should be addressed to: Milos MARYSKA; milos.maryska@vse.cz

Received date:20 October 2025; Accepted date:25 January 2026; published date: 27 April 2026

Academic Editor: Katarzyna Ragin-Skorecka



**Abstract**

This paper reviews the evolution of data architectures through the lens of enterprise architecture, focusing on the Data Warehouse (DWH), Data Lake (DL), and Data Lakehouse (DLH). It positions data architecture as a core element of enterprise architecture and outlines how frameworks such as Zachman, TOGAF, and the Gartner EA approach frame governance, integration, and strategic alignment of the data layer. Methodologically, the study follows a literature-based analysis and synthesis, aligned with the objective of describing the evolution of data architectures and providing a comparative view. The paper characterizes DWH as a structured, schema-on-write, multi-layer repository (stage/core/mart) with strong data quality but slower onboarding; DL as an object-storage approach emphasizing schema-on-read and ELT, high flexibility, and risks related to metadata/catalog management and transactional guarantees; and DLH as a hybrid that adds a metadata and transactional layer (ACID) while preserving DL flexibility, albeit with higher architectural complexity and skill demands. The paper also situates related concepts (modern cloud DWH, data fabric, data mesh) and presents a comparative table to summarize trade-offs. Overall, it offers a practitioner-oriented synthesis that clarifies where each paradigm fits and highlights the central role of governance and metadata stewardship in avoiding "data swamps" and sustaining value creation.

**Keywords**: data architecture; data warehouse; lakehouse; governance.

**Cite this Article as:** Felix ESPINOZA, Lea Nedomova and Milos MARYSKA (2026), " From Data Warehouses to the Lakehouse", Journal of Administrative Sciences and Technology, Vol. 2026 (2026),Article ID 558858, https://doi.org/10.5171/2026.558858

## Introduction

Architecture is often associated with building design, and the analogy is useful for information systems as well. In both domains, the goal is a functional and stable result; architecture additionally considers coherence and aesthetics across components (Zachman, 1987). The same principles apply to complex socio-technical systems: when many components interact, a systematic and well-structured approach is essential (Rashed & Drews, 2021).

John Zachman highlighted the growing complexity of information systems as early as 1987 and emphasized the need for logical structures and architectures (Zachman, 1987). As the number of applications and interconnections continues to grow, new trends and technologies emerge that challenge established paradigms (Gerber, le Roux, Kearney, 2020).

Phenomena such as the Internet of Things, robotic process automation, and cloud computing generate substantial data volumes and intensify the need to manage big data (Sá, Martins, Simões, 2015). Organizations therefore face the challenge of processing—and, more importantly, extracting value from—rapidly expanding data volumes (Mušić, Hribar, Fortuna, 2024). Data-driven initiatives have moved to the forefront and are expected to transform data into actionable outcomes (European Commission, 2020). Enterprise architecture must reflect the growing importance of the data layer and incorporate these trends appropriately.

Data, technology, and enterprise-architecture practices are evolving rapidly (Serra, 2024; Wang, Jiang, Cosenz, 2025). This dynamism affects organizations' ability to select a data-architecture approach that best supports their business objectives (Sebastian-Coleman, 2018).

The aim of this paper is to analyze how enterprise architecture influences data architecture, describe the evolution of data architectures, and compare the main approaches.

## Methods

Given the stated aim, the paper uses analysis and synthesis. The theoretical foundation is developed by analyzing studies and articles listed in the Literature section, followed by a synthesis that derives findings, conclusions, and links between the covered subject areas.

The comparison uses attributes defined in the following chapters. Key criteria include storage type, degree of centralization, data-security posture, total cost of ownership, technology maturity, and the organizational skill set required for implementation, continuous improvement, and operations.

## State of the Art

Data architecture is a component of enterprise architecture (EA). EA is often compared to architecture in construction: a poorly designed building reduces occupants' comfort, and even minor modifications become costly and labor-intensive. Similarly, information systems benefit from a holistic approach. Common methodologies that shape enterprise architecture—and thereby influence data architecture—include the Zachman Framework, The Open Group Architecture Framework (TOGAF), and the Gartner enterprise architecture approach. These are well covered in the literature and are therefore not described here in detail.

Data architecture importance increases with the growing volume and diversity of data. Many definitions of "data" exist; in this paper, we refer to data as individual facts and raw material from which information can be inferred (Brackett, 1994). Data can therefore be understood as a set of representations describing properties and states of objects and phenomena. In information systems, data are typically stored in digital form to enable further machine processing.

Global data volumes are growing exponentially. Estimates suggest that worldwide data creation and replication could reach 394 zettabytes by 2028, up from 33 zettabytes in 2018, indicating a dramatic expansion of data sources.

Digital transformation is increasingly data-driven (European Commission, 2020). Because acquiring, processing, and preserving data is costly, organizations seek to maximize and monetize this asset (Rashed & Drews, 2021). This motivation has fueled data-driven initiatives that place data at the center and emphasize their strategic importance for survival and growth in competitive environments (Rashed & Drews, 2021).

Across the reviewed sources, data-driven initiatives can be seen as a convergence of the phenomena described above and a response to the need to change established approaches to data processing. Progress toward the data-driven

domain can be assessed using enterprise data maturity models (Serra, 2024).

Managing data sources is one responsibility of data architecture, and the data dimension should be addressed in any EA methodology, potentially complemented by specialized frameworks such as the Data Management Association (DAMA). DAMA describes data architecture as a set of artifacts at varying levels of abstraction, including standards that define how data are acquired, stored, managed, used, and retired.

From this definition, it is clear that data architecture is not a narrowly bounded discipline. It includes many activities that intersect with other EA viewpoints. Without interlinking these viewpoints and maintaining a holistic perspective, enterprise-wide objectives are unlikely to be adequately supported. Key components of data architecture include:

- Data governance, which defines quality assurance and ensures compliance with legal standards and regulations.
- Data stores used by the organization, including on-premises platforms and cloud environments.
- Data security, including access control and protection against loss or damage.
- Data sources and their description through metadata, which is particularly critical for unstructured data.
- Data models that capture the organization's representation of data and the relationships between concepts.
- Data processes that govern how data enter the platform and what transformations are applied.

The following sections present modern data architectures. For completeness, the traditional DWH is included because it remains prevalent and has also been widely adopted in cloud environments.

## Evolution of Data Architectures

Data integration and analytics are not new concerns. Since the 1970s, organizations have tried to combine data from multiple source systems to improve decision-making. This effort gave rise to Decision Support Systems, from which Management Information Systems later emerged; today, these areas are often grouped under Business Intelligence. Their shared goal is to maximize value from data by integrating it into a central repository that supports analysis and reporting. They also share constraints: data must typically be structured, processing latency must be managed, and updates have traditionally occurred with a daily delay. Modern approaches seek to overcome these limitations by supporting diverse data sources and enabling real-time or near-real-time processing.

The following sections introduce three representative paradigms: the Data Warehouse, the Data Lake, and the Data Lakehouse.

### *Data Warehouse*

A Data Warehouse is a centralized repository of structured data, typically implemented on relational database technology, that integrates data from an organization's source systems. Historically, DWHs were deployed on-premises, but today they are commonly operated in the cloud as well. Data ingestion is usually implemented through an ETL (Extract, Transform, Load) layer.

A DWH is commonly organized into three layers - Stage, Core, and Mart - often referred to in modern terminology as Bronze, Silver, and Gold. The Stage (Bronze) layer preserves source-level granularity and stores data with minimal modification. The Core (Silver) layer integrates data across sources and may adjust granularity as needed. The Mart (Gold) layer is optimized for analytics and reporting and typically contains the most aggregated data.

Structured data bring clear advantages: well-defined types and constraints enable systematic data-quality controls. Indexes, primary keys, and foreign keys support performance and help prevent duplication while enforcing consistency. Schema-on-write validation ensures that records failing predefined formats are rejected at load time, reducing inconsistencies. High data quality, in turn, supports strong SQL performance for analytics and reporting and enables straightforward connectivity for end-user tools such as Microsoft Excel.

DWH limitations are mainly related to flexibility and speed of change. Onboarding new sources often requires careful modeling and transformation into a predefined structure. In complex warehouse implementations, recalculations and transformation chains can be time-consuming, which constrains how frequently data can be refreshed.

### *Data Lake*

The data lake concept was introduced by James Dixon in 2010 to describe a centralized repository—often cloud-based—that can store both structured and unstructured data. A data lake keeps data in their original format without

requiring upfront transformation. Removing strict schema requirements accelerates ingestion and increases flexibility for downstream processing. This approach is known as schema-on-read, where structure is applied at query or analysis time.

Data lakes typically follow the ELT (Extract, Load, Transform) pattern: data are extracted and loaded first, while transformations occur during processing. These characteristics support agile development and speed up change delivery. Users can perform ad-hoc analysis on raw data in many formats, such as spreadsheets, logs, and JSON-based social-media data.

A data lake is designed to scale by combining low-cost storage with elastic compute. This enables efficient management of petabyte-scale volumes and supports computationally intensive workloads. However, the absence of fixed structures increases the need for user expertise, making this architecture particularly suitable for advanced data practitioners.

Because it can store heterogeneous sources - both structured and unstructured - a data lake is a natural environment for developing machine-learning (ML) algorithms and artificial intelligence (AI) models.

A key operational challenge is long-term metadata and catalog management. The data catalog helps users discover datasets and supports indexing and governance. If catalog curation is neglected, visibility and performance degrade, leading to a "data swamp," where the repository becomes difficult to use and trust.

Another limitation is the lack of built-in transactional guarantees in many data-lake implementations. This increases the risk of inconsistent states after failures and can require additional engineering effort to restore integrity, especially in multi-writer scenarios.

In summary, a data lake is a flexible and scalable architecture for storing and analyzing large and diverse datasets. It is particularly attractive for agile development and advanced analytics, but effective use depends on strong metadata management and sufficient technical expertise.

### *Data Lakehouse*

The Data Lakehouse concept, introduced by Databricks, describes a cloud-based repository that aims to combine the strengths of the Data Lake and the Data Warehouse. It leverages low-cost object storage and supports both structured and unstructured data. Unlike a plain DL, it adds transactional capabilities to ensure integrity and consistency, while retaining schema-on-read flexibility and the ELT approach.

Like a data lake, a Lakehouse is well suited to machine-learning and artificial intelligence workloads. At the same time, it can offer warehouse-like conveniences for structured data by allowing schemas and constraints to be defined, improving usability for discovery and querying. Using one repository for both analytical and (in some cases) operational workloads can reduce data duplication and simplify integration across systems.

A key differentiator is the metadata and transactional layer, which enables ACID semantics, indexing, data versioning, and richer catalog descriptions. However, this layer also increases architectural complexity and requires careful design and ongoing maintenance. It plays a central role in access control, security, and auditability. Technologies such as Delta Lake exemplify how transactional table formats can provide ACID properties (Atomicity, Consistency, Isolation, Durability) on top of object storage.

The Lakehouse model also supports combining historical data with current inputs, which is valuable for predictive modeling and advanced analytics that rely on both trends and fresh signals. To enable near-real-time use cases, Lakehouses are often integrated with streaming technologies (e.g., Apache Kafka or Spark Structured Streaming).

Implementing a Lakehouse comes with challenges. It typically requires investment in platform engineering and in training teams to operate and govern the environment. Because the paradigm is relatively new, organizations may find fewer established reference architectures and operational playbooks than for mature DWH solutions. Migration from legacy DL or DWH platforms can also be complex due to data movement and process redesign.

Despite these challenges, the Lakehouse has the potential to become a dominant paradigm because it combines flexibility, scalability, and robustness in a single platform. For organizations seeking to unify BI and advanced analytics on governed data, it can be an attractive option - provided they invest in governance, metadata practices, and the required skills.

### *Comparison of Key and Other Architecture Models*

For completeness, this section briefly notes other data-architecture concepts that are relevant in

practice but are outside the scope of the detailed comparison.

Table 1 provides a high-level comparison of selected architectural models. Data Mesh is included as an organizational and governance approach rather than a storage technology, and therefore it is not directly comparable on purely technical dimensions.

The table also contextualizes the relational data warehouse, Data Lake, and Data Lakehouse approaches described above. The table was inspired by Serra (2024) and significantly extended for other measures.

**Table 1: Comparison of Selected Architecture Models, Author's**

| Characteristic | Relational DWH | Data Lake | Modern DWH | Data Fabric | Data Lakehouse | Data Mesh |
|---|---|---|---|---|---|---|
| **FOUNDATION** | | | | | | |
| **Year Introduced** | 1984 | 2010 | 2011 | 2016 | 2020 | 2019 |
| **Centralized / Decentralized** | C | C | C | C | C | DC |
| **DATA MODELING** | | | | | | |
| **Storage Type** | Relational | Object | Relational and Object | Relational and Object | Object | Domain-specific |
| **Schema Type** | On-Write | On-Read | On-Read and On-Write | On-Read | On-Read | Domain-specific |
| **OPERATIONS** | | | | | | |
| **Data Security** | High | Low to Medium | Medium to High | High | Medium | Domain-specific |
| **Data Latency** | Low | High | Low to High | Low to High | Medium to High | Domain-specific |
| **ADOPTION & COST** | | | | | | |
| **Time to Value** | Medium | Low | Low | Low | Low | High |
| **Total Cost of Solution** | High | Low | Medium to High | Medium to High | Low to Medium | High |
| **Company Skill Set Needed** | Low | Low to Medium | Medium | Medium to High | Medium to High | High |
| **CAPABILITY** | | | | | | |
| **Supported Use Cases** | Low | Low to Medium | Medium | Medium to High | High | High |
| **Difficulty of Development** | Low | Medium | Medium | Medium | Medium to High | High |
| **Maturity of Technology** | High | Medium | Medium to High | Medium to High | Medium to High | Low |

Legend:

- C: Centarlized
- DC: Decentralized

**Conclusion & Discussion**

This paper sets out to explain how data architecture has evolved and to compare three prominent paradigms—Data Warehouse, Data Lake, and Data Lakehouse—within an enterprise-architecture context. Across the sections, a consistent picture emerges: architectural choices are not merely technical; they are socio-technical decisions shaped by governance requirements, the nature of data sources, organizational skills, and the operating model promoted by EA frameworks.

The Data Warehouse remains a strong fit wherever structured data, regulatory reporting, repeatable analytics, and stable schemas dominate. Its layered design (stage/core/mart) and schema-on-write discipline enable data quality, lineage, and reliable SQL performance, which in turn support consistent reporting and decision-making. The trade-off is lower agility in onboarding novel or semi-structured sources and longer processing cycles when transformations are complex. These characteristics are well-recognized in the discussion of DWH benefits and limitations.

The Data Lake responds to the explosion in volume, variety, and velocity by decoupling storage and computation and embracing schema-on-read and ELT. This unlocks flexibility for data science and ML and offers attractive cost dynamics on object storage. However, the same flexibility increases the dependency on metadata management and catalog curation; without it, the risk of a "data swamp" is real. Moreover, the absence of built-in transactional guarantees complicates reliability for multi-writer analytics. These advantages and risks are highlighted in the part of DL.

The Data Lakehouse seeks to combine these worlds by overlaying the lake with a metadata layer that provides ACID transactions, indexing, versioning, and governance hooks. The result is a platform where BI and advanced analytics can converge without duplicating data across disparate stores. Yet, this unification introduces architectural complexity and a skill uplift for teams; reference to technologies that realize ACID semantics (e.g., transactional table formats) illustrates both the promise and the implementation burden. In short, DLH can reduce fragmentation and accelerate value delivery, provided the organization invests in platform engineering, catalog practices, and security.

A secondary contribution of the paper is to locate related concepts—modern cloud DWH, data fabric, and data mesh—relative to the three core paradigms. The text correctly frames modern cloud DWH as an elastic evolution of DWH, data fabric as an integration and delivery approach driven by active metadata, and data mesh as primarily an organizational and governance model (not a storage technology) that can sit atop any of the underlying stores. The comparative table usefully synthesizes trade-offs (e.g., latency, security, cost, maturity, skills), even if some entries reflect authorial assessment rather than benchmarked measures.

Taken together, the analysis supports pragmatic guidance. Organizations with strict governance and reporting needs can continue to rely on DWH. Those pursuing exploratory analytics and ML at scale may favor a DL, provided they invest early in cataloging and stewardship. When unifying BI and data science on a single, governed repository is paramount, DLH is attractive, with the caveat of operational complexity and team enablement.

In conclusion, no one-size-fits-all architecture exists. The comparison underscores that governance, metadata, organizational skills, and alignment to business outcomes should drive selection. In many organizations, a hybrid path that evolves existing DWH and DL capabilities toward more governed, interoperable platforms (including lakehouse patterns) can preserve current strengths while enabling new analytical and AI use cases with manageable risk.

In practice, the relational data warehouse still dominates as the most established and proven architecture for storing and processing structured data, particularly in industries like finance, pharmaceuticals, and retail. Its main advantages include high data quality, a well-defined data model, and smooth integration with BI tools. However, it is less flexible and often costly to scale or adapt to changing business needs. The Data Lake brought more freedom for storing unstructured data and offered lower storage costs, but many organizations struggled with the "data swamp" problem—loss of governance and declining data quality. As a result, both architectures have gradually evolved into more modern approaches that aim to combine structure and flexibility.

In recent years, Modern DWH, Data Lakehouse, Data Fabric, and Data Mesh have gained traction. The Modern DWH (e.g., Snowflake, Big Query)

merges cloud scalability with relational principles, offering rapid deployment and self-service analytics. The Data Lakehouse combines the advantages of DWH and Data Lake—cost-efficient raw data storage with ACID transactions and structured querying—appealing to both analysts and data scientists. Data Fabric remains more of a vision than a fully implemented reality, perceived as an ideal end state for integrating diverse data sources through metadata and automation. Data Mesh is increasingly popular in large, domain-oriented organizations, bringing agility and ownership of data within business domains, though it requires a mature data culture and strong governance. Overall, most companies today operate somewhere between traditional DWH and modern cloud-based models, while only the most advanced organizations are experimenting with fabric or mesh architectures.

In conclusion, there is no one-size-fits-all architecture. The comparative treatment underscores that governance, metadata, skills, and alignment to business outcomes should drive selection. A measured path—often hybrid—can sequence adoption, preserve existing strengths, and realize new capabilities without unnecessary disruption

**Acknowledgements**

This paper was supported by the institutional-support fund for the long-term conceptual development of science and research at the Faculty of Informatics and Statistics, Prague University of Economics and Business (IP400040).

**Literature**

- Brackett, M. H. (1994). *Data sharing: using a common data architecture*. John Wiley.
- European Commission. (2020). A European strategy for data. Communication from the Commission to the European Parliament, the Council, the European Economic and Social Committee and the Committee of the Regions, 2020(1), 1-34. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A52020DC0066
- Gerber, A., le Roux, P., Kearney, C., & van der Merwe, A. (2020). The Zachman Framework for Enterprise Architecture: An Explanatory IS Theory. *Responsible Design, Implementation and Use of Information and Communication Technology*, 383-396. https://doi.org/10.1007/978-3-030-44999-5_32.
- Harby, A. A., & Zulkernine, F. (2025). Data lakehouse: a survey and experimental study. Information Systems, 127. https://doi.org/10.1016/j.is.2024.102460
- Mušić, D., Hribar, J., & Fortuna, C. (2024). Digital transformation with a lightweight on-premise PaaS. *Future Generation Computer Systems*, *160*, 619-629. https://doi.org/10.1016/j.future.2024.06.026
- Noran, O., & Bernus, P. (2017). Business Cloudification - An Enterprise Architecture Perspective. *Proceedings of the 19th International Conference on Enterprise Information Systems*, 353-360. https://doi.org/10.5220/0006248603530360.
- Rashed, F., & Drews, P. (2021). How Does Enterprise Architecture Support the Design and Realization of Data-Driven Business Models? An Empirical Study. *Innovation Through Information Systems*, 662-677. https://doi.org/10.1007/978-3-030-86800-0_45.
- Sá, J. O. e, Martins, C., & Simões, P. (2015). Big Data in the Cloud: A Data Architecture. *New Contributions in Information Systems and Technologies*, 723-732. https://doi.org/10.1007/978-3-319-16486-1_71
- Sebastian-Coleman, L. (2018). *Navigating the Labyrinth: An Executive Guide to Data Management*. Technics Publications.
- Serra, J. (2024). *Deciphering data architectures: choosing between a modern data warehouse, data fabric, data lakehouse, and data mesh*. O'Reilly.
- Statista. (n.d.). Data growth worldwide, 2010-2028.
- TOGAF. (2025). The Open Group Architectural Framework. Retrieved January 23, 2025, from https://www.opengroup.org/togaf
- Wang, F., Jiang, J., & Cosenz, F. (2025). Understanding data-driven business model innovation in complexity: A system dynamics approach. *Journal of Business Research*, *186.* https://doi.org/10.1016/j.jbusres.2024.114967
- Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, *26*(3), 276-292. https://doi.org/10.1147/sj.263.0276