Journal of Data, Information and Management (2024) 6:309–334
https://doi.org/10.1007/s42488-024-00132-1

ORIGINAL ARTICLE

The evolution of data storage architectures: examining the secure value of the Data Lakehouse

Nathalie Janssen[1,2] · Tharaka Ilayperuma[2] · Jeewanie Jayasinghe[1] · Faiza Bukhsh[1] · Maya Daneva[1]

Received: 16 May 2024 / Accepted: 26 July 2024 / Published online: 15 August 2024

Abstract

The digital shift in society is making continuous growth of data. However, choosing a suitable storage architecture to efficiently store, process, and manage data from numerous sources remains a challenge. Currently, there are three storage architecture generations in practice, and the most recent one is Data Lakehouse. Given its novelty, limited research has been done into the rationale behind its introduction, strengths, and weaknesses. In order to fill this gap, this study aims to investigate the secure value (comparative strengths) of the data lakehouse architecture compared to data warehouse and data lake architectures. After conducting a comprehensive systematic literature review, we propose a data storage evolution model showing the comparative strengths and weaknesses of data warehouse, lake, and lakehouse architectures. With the use of the proposed model and expert interviews, this study demonstrates the secure value of the data lakehouse compared to the preceding architectures. In addition, the study presents a high-level view of the overlapping strengths of data Lakehouse with both data warehouse and data lake. In essence, the artifact produced by this study can be used to explain the rationale behind the evolution of data storage architectures. Further, the proposed model will help the practitioners in studying the trade-off between different architectures to offer recommendations. Finally, authors acknowledge that this study has several limitations, such as the limited sample size for the interviews and the bias due to the use of qualitative research approach. However, all the available measures were taken to minimize the effects of these limitations.

Keywords Data storage architecture · Data warehouse · Data lake · Data lakehouse · Data storage · Evolution model

1 Introduction

In today's world, information has become one of the most important assets a company can have. Since society has been making a digital shift, more and more data sources have become available that serve as "oil" to the company. As the years progressed, the rate at which data was generated accelerated, giving rise to the term 'big data'. Doug Laney (Laney et al. 2001) is the first to describe big data with the 3 V's: Volume, Variety, and Velocity. Volume is related to the size of the data set, variety is all about the different types of data formats, and velocity is the speed at which the data comes in and goes out (Chen and Zhang 2014). These V's are seen as the original V's that define big data. However, throughout the years, more V's were introduced to touch upon different aspects of big data. From a practical point of view, once the data does not fit on a machine or the processing times are really high, then you are dealing with big data.

For each V that defines big data, different challenges occur. For instance, challenges related to the volume of the

Tharaka Ilayperuma, Jeewanie Jayasinghe, Faiza Bukhsh and Maya Daneva contributed equally to this work.

✉ Jeewanie Jayasinghe
j.a.jayasinghearachchige@utwente.nl

Nathalie Janssen
nathalie.esmee@hotmail.com

Tharaka Ilayperuma
tharaka.ilayperuma@staffs.ac.uk

Faiza Bukhsh
f.a.bukhsh@utwente.nl

Maya Daneva
m.daneva@utwente.nl

[1] EEMCS, University of Twente, Drienerlolaan 5, Enschede 7522NB, The Netherlands

[2] Department of Computing, Staffordshire University, College Rd, Stoke-On-Trent ST4 2DE, UK

data can be: 1) the way data should be stored, 2) how the stored data is still easily accessible when needed, and 3) how analyses can be performed in an efficient way (Chen and Zhang 2014). With regards to Velocity, typical challenges are how to maintain a scalable storage solution in order to deal with the fast-incoming data and allow real-time processing and immediately make this available for usage. According to Lu et al. (2018), dealing with the variety of data in the current database ecosystems is the most challenging issue. The data can now be presented in a structured, semi-structured, or unstructured format. As a result, the need to deal with all these different types of data grew, so additional data management techniques started to emerge in order to deal with big data. One of the most robust techniques is the use of cloud computing.

There are many possibilities that can be offered by cloud computing, one of which is a Data Management Platform (DMP) solution. A data management platform is a central hub where data from multiple sources are stored and managed. The data is formatted through a data pipeline to be used for any analytical purpose such as making predictions, detecting trends, gaining a deeper understanding of the customers, performing analyses and creating reports, or publishing critical insights on a dashboard.

One of the core parts of a data management platform is the storage module. Given the complexity that comes with dealing with big data, it is crucial to implement the right data storage architecture. Over the last few decades, it can be observed that three generations of storage architectures have evolved. The first is the Data Warehouse, which is now seen as a traditional storage solution where structured data from multiple sources are stored together in a unified data repository for analytical and reporting purposes. Despite its strengths, there are also some disadvantages, such as lack of flexibility not handling semi-structured and unstructured data well. Moreover, the implementation, maintenance, and scaling costs of data warehouses are very high. Consequently, a new architecture was developed to provide storage solutions to deal with these disadvantages.

The second generation is the data lake. This can be defined as a methodology enabled by a massive data repository based on low-cost technologies that improve the capture, refinement, archival, and exploration of raw data within an enterprise. A data lake contains the mess of raw unstructured or multi-structured data that, for the most part, has unrecognized value for a firm (Fang 2015)." Next to being able to support any format, data lakes are flexible, durable, and cost-effective. Nevertheless, this storage solution also has its downsides. For instance, it does not support data management functionalities, lacks support for ACID (atomicity, consistency, isolation, and durability) transactions, has the risk of keeping corrupt data in a data lake since there is no quality control, and there is no possibility for versioning and time travel. As a result, a third architecture was introduced.

The third generation is the data lakehouse. This architecture combines the best practices of a data warehouse and a data lake. It incorporates the low-cost and flexible architecture of the data lake and data warehouse capabilities such as traditional database management features such as metadata management, caching, indexing, schema enforcement, data layout optimizations, and ACID transactions. In doing so, the data lakehouse also tackles the limitations of the previous generations of storage architectures. Finally, the data lakehouse is capable of serving use cases from traditional reporting and business intelligence to modern workloads such as data science and machine learning use cases.

Hence, it is clear that the evolution of storage architectures consists of data warehouses, data lakes, and data lakehouses. However, deeper analysis and comparison are required to assess the strengths and weaknesses of each one of these data storage architectures. Due to the novelty of data lakehouse, limited research has been conducted on the rationale behind its introduction, strengths, and weaknesses. Therefore the research gap identified in this study is the lack of comparative analysis of the strengths and weaknesses of data lakehouse architecture concerning the other two architectures. Thus, we focus on investigating the secure value of data lakehouse meaning that we plan to explore the comparative strengths of data lakehouse architecture that make it a resilient alternative over the data warehouse and data lake architecture in this study. To explore the secure value that the data lakehouse provides over the data warehouse and data lake, we need to deeply analyse the evolution process and the worth of each. In this direction, we identify that the following problems need to be investigated. Firstly, in literature and practice, no model explains the rationale behind the evolution process. Without this knowledge, it is very challenging to understand the reasons behind the introduction of the data lakehouse architecture. Secondly, given the novelty of the data lakehouse architecture, a limited amount of research is done on this topic, especially related to the analysis of the secure value of the data lakehouse architecture.

All the problems mentioned above contribute to a knowledge gap in literature and practice in understanding the added value of the data lakehouse in relation to the evolution of storage architectures. This research aims to close this knowledge gap by developing a data storage architecture evolution model that shows how different data storage architectures are connected with each other through their respective strengths and weaknesses. This will be achieved by performing an in-depth literature review, producing an evolution model, and performing validation interviews. As such, this research will contribute to understanding the secure value of the data

lakehouse architecture as a data management platform in relation to the evolution process.

Understanding secure value enables us to explore why data lakehouse may be a safer option for dealing with emerging data technology requirements. This we try to achieve by answering the following research question:

> RQ: "How to model the secure value of the data lakehouse architecture by studying its comparative strengths and weaknesses with respect to other data storage architectures?"

By answering the above questions, this study will contribute both to literature and practice through the following contributions: we will 1) provide a comparative study between data warehouses, data lakes, and data lakehouses, 2) present a validated model that explains the evolution of storage architectures. These contributions aim to examine the value of the data lakehouse by analysing the evolution of the data storage architectures and closing the current knowledge gap in literature and practice.

The remainder of this paper is as follows. Firstly, the three data storage architectures used in this research are discussed in the background section. Secondly, the research methodology used for this research study is explained. This is followed by the findings of the literature study that was performed to design a Data Storage Evolution Model. Next, a discussion on the validation and improvement of the designed model is provided, and a newly developed architecture is presented. Finally, a conclusion is given, along with a discussion on limitations.

2 Background

The three data storage architectures, namely data warehouse, data lake, and data lakehouse, are used as the foundation of this research. Therefore, these three architectures are discussed in detail in this section. In the first sub-section, a high-level comparison of these data storage architectures is presented. Then each of these models is described in detail in the subsequent sections.

2.1 Data storage architectures

While the big data wave emerged over the years, the data storage solutions also started to develop and adjust themselves accordingly. Nowadays, data warehouses are seen as traditional data storage solutions. For a long time, this has been considered to be the unique solution that was able to deliver accurate and reliable information to an organization. With the big data wave around year 2010, the second data storage architecture evolution emerged. First of all, there was a need for a storage solution that could deal with the challenges of big data. Secondly, data governance became a crucial part of the new information architecture since privacy and compliancy became more important. As a result, the data lake was developed, which can be seen as an "incubator" environment in which data of any type can be stored to generate insights (Madera and Laurent 2016). This storage architecture challenged the whole rationale of a data warehouse and can be seen as the exact opposite. However, it turned out that new problems occurred with the data lake technology and the governance and security aspects were still not fully solved. Besides, the emergence of the big data wave has not yet stopped as society is still shifting towards digitalization. Consequently, this led to the rise of the third data storage architecture: data lakehouse. This can be defined as "a data management system based on low-cost and directly accessible storage that also provides traditional analytical DBMS (Database Management System) management and performance features such as ACID transactions, data versioning, auditing, indexing, caching, and query optimization (Armbrust et al. 2021)."

The progression of the storage architectures in relation to the big data waves is graphed in Fig. 1. As previously mentioned, the big data wave is expected to continue exponentially. On the other hand, the development of data storage architectures started with a very steep line. The reason is that the data warehouse and data lake were disruptive technologies. Afterward, the line tends to slow down, indicating the newer technologies not to be as disruptive as the previous ones. This claim is based on the fact that a data lakehouse incorporates best practices from the data warehouses and data lakes. As for the data meshes that are expected to be the next generation, the rationale will be built on the ideas of the data lakehouse. As a result, the trend line showing the evolution of storage architectures is flattening.

2.1.1 Data warehouse architecture

Understanding the definition of a data warehouse is essential to grasp how the architecture is constructed. The concept of data warehouses originated in 1980s when two IBM researchers developed the business data warehouse. There were numerous visualizations of how the data warehouse is constructed. To obtain a general understanding of the main components of a data warehouse, we have chosen to look into three different visualizations of the data warehouse architecture. The first is taken from an article that was published in a Business Journal (Al-Okaily et al. 2022). The second is taken from Inmon et al. (2021). Lastly, we evaluated an architecture published on a blog dedicated to technologists by Lavrentyeva and Sherstnev in (2022).

The architecture presented by Al-Okaily et al. in (2022), consists of four horizontal layers and one vertical layer. The bottom-horizontal layer indicates the data sources, such as operational systems, ERP (enterprise resource planning) systems, and external data sources. Before this data is stored

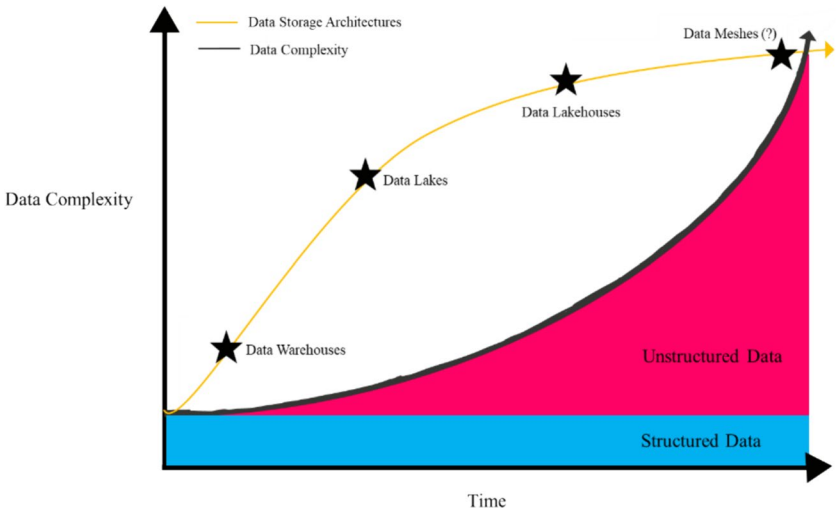

Fig. 1 Evolution of storage architectures in relation with data complexity

in the data warehouse, the data lands in the second layer which is a transformation component. This component is dedicated to cleaning and transforming the data to a specific structure since a data warehouse applies a schema to all stored data. That is how they manage to store only structured data. When the data is ready, it will be stored in the data warehouse database. From here, there are two possibilities, either the end-user directly obtains the data stored in the data warehouse, or it is moved and stored in a data mart. A data mart is a small and simple form of a data warehouse that stores data related to a specific department or subject. A data warehouse can consist of multiple data marts given that there is a well-structured way of storing the data. The risk here is that data becomes isolated because data marts are stand-alone entities. Finally, the layer on top is dedicated to the possible end-users who desire to utilize the data that is stored in the data warehouse. Then there is one vertical layer, the metadata management layer, and this is a layer that is a specific trait of the data warehouse. Due to the metadata management layer, the data warehouses can efficiently incorporate data governance practices.

The second visualisation, presented in Inmon et al. (2021) by Inmon et al., projects a minimalistic representation of the data warehouse architecture. It only consists of three layers: data source, data warehouse, and use cases. The data source layer indicates that only the storage of structured data is supported. Then the data is stored in a data warehouse that has a data management and governance layer. This layer contains several features that are implemented in this layer, such as metadata, taxonomies, data lineage, and ETL (extract, transform and load) processes. Finally, the third layer presents the use cases for which a data warehouse is typically used which are BI(business intelligence) and SQL (structured query language) analytics.

The third, presented in Lavrentyeva and Sherstnev (2022) is a very elaborate architecture that explains different processes related to the data warehouse. First of all, the architecture is split up into three tiers, the bottom tier, middle tier, and top tier. The bottom tier starts with a layer that indicates data from different sources is loaded and pushed into the data warehouse via ETL processes. When the data enters the data warehouse, it is stored in a central data store. Then, a metadata and summary data database is created to store information about the data. From here, the data warehouse is split up into multiple data marts that represent subsets of data and serve specific business stakeholders. These all are part of the bottom tier. Then the middle tier is a service layer where online analytical processing (OLAP) is carried out. OLAP is a computing method reorganizing data into a multidimensional format, enabling users to easily and selectively extract and query data. This can then be analyzed from different points of view. Finally, the top tier is the layer where all the tools are connected to utilize the data for particular use-cases like data mining, analysis, or reporting.

In summary, the first architecture consists of 5 layers or components: 1) data source, 2) transformation component, 3) data storage including both a data warehouse database and data marts, 4) end-user tools/use-cases, and 5) metadata management layer. The second architecture is a simplistic representation of the data warehouse, starting with a data source layer, then the data warehouse layer, and the use-case layer. Finally, the third architecture is very sophisticated because it is built out of three tiers, and it describes the way data flows through the data warehouse in a detailed way.

2.1.2 Data lake architecture

Given the scope of this study, we follow the architectural view explanation "data lake uses a flat architecture to store data in its raw format and also support the storage of cleansed and transformed data. Each data entity in the lake is associated with a unique identifier and a set of extended metadata, and consumers

can use purpose-built schemas to query relevant data, which will result in a smaller set of data that can be analysed to help answer a consumer's question (Walker and Alrehamy 2015)." The data lake challenges data warehouses for storing heterogeneous complex data (Khine and Wang 2018). All the data that an organization wants to ingest, will be stored in a data lake in their original format. As a result, "complex pre-processing and transformation of loading data into data warehouses are eliminated, and the upfront data ingestion costs are reduced (Khine and Wang 2018)." Once the data is stored, the data is made available to anyone from the organization who is authorized to perform analyses on the data. This definition contributes to enhancing the understanding of what and how the data lake architecture is constructed. Again, we have evaluated three types of data lake architectures found in the literature by Lavrentyeva and Sherstnev in (2022), Inmon et al. in (2021), and (Ravat and Zhao 2019a).

Firstly, it is important to note that all three architectures represent a data lake in three layers. The first architecture, presented in Lavrentyeva and Sherstnev (2022), starts with a data source layer. It depicts data sources and types like OLTP (online transaction processing), flat files, ERP/CRM (customer relationship management), Cloud, social media, and logs. Then, the data is stored in the storage component that is split up into four zones. It starts with a bronze zone where raw data is stored. Then there is a silver zone where intermediate data is stored. This entails data that was cleaned and processed to some extent. The third is the gold zone where trusted data is stored, and this data is seen as the source of truth and can be immediately used for data consumption. Next to these zones, there is also a governing zone that implies that on top of this storage object features such as security measures, ETL processes, and lookups through metadata are supported. Finally, the third layer is the data consumption zone which implies that data from any zone can be used for any use-case.

The second architecture, presented by Inmon et al., (2021), is again a very minimalistic representation of the data lake architecture having three layers that show structured, textual, and other unstructured data types are supported. The data is stored in an 'open data lake', after which the data can be used at any time for machine learning use-cases. The third architecture designed by Ravat and Zhao (2019a) also has three layers in which the first represents the data sources, the second the data lake as a storage object, and finally data consumption. This architecture also presents different types of data sources, such as weblogs, social media, OLTP, ERP, CRM systems, documents, emails, machine-generated, and cloud services. Secondly, the data lake consists of three data zones and one overarching governing zone, (1)there is a raw data zone where data in its original format is stored. (2) is called a process zone where processed data is stored, and (3) there is an access zone where fully processed data for certain business demands is stored. At last, there is a layer indicating that the stored data can be used for consumption.

In summary, a data lake architecture typically consists of 3 layers: the data source layer, the data lake layer, and the data consumption layer. The first layer represents all the different data formats a data lake can ingest. The second layer is the storage object and consists of different data zones where data is stored in different states. And finally, the third layer is dedicated to all the different use cases for which a data lake is suitable.

2.1.3 Data lakehouse architecture

In the literature, only a few definitions were found for the data lakehouse. Armbrust et al. (2021) defines the data lakehouse as "a data management system based on low-cost and directly accessible storage that also provides traditional analytical DBMS management and performance features such as ACID transactions, data versioning, auditing, indexing, caching, and query optimization". Another definition by Schneider et al. (2024) defines it as "lakehouse is an integrated data platform that leverages the same storage type and data format for reporting and OLAP, data mining and machine learning, as well as streaming workloads." Lakehouses thus combine the key benefits of data lakes and data warehouses: low-cost storage in an open format accessible by a variety of systems from the former, and powerful management and optimization features from the latter. In order to understand the data lakehouse better, again three architectures were evaluated.

We found articles by Armbrust et al. (2021), Inmon et al. (2021), and Lavrentyeva and Sherstnev (2022), and Schneider et al. (2023) related to the data lakehouse architecture.

The architecture designed by Armbrust et al. (2021) starts with a data source layer indicating that storing structured, semi-structured, and unstructured data is supported. Then, the data enters the data lake which is the storage object. On top of this storage object, there is a metadata, caching, and indexing layer. This layer enables the implementation of governance and data management features. In order for the data to be used for any use-case, a connection is established with APIs. Different APIs can be implemented to allow different use-cases to be performed. The data lakehouse can be used for any type of use-cases such as BI, reports, data science, and machine learning.

The second architecture is presented by Inmon et al. in (2021). The first layer in this architecture shows all the different data formats that are supported. These are stored in its raw form in a data lake.

On top of this storage object a layer is presented that is called 'curated data with governance'. The goal of this layer is to indicate that different database management techniques

are supported. Finally, through open API's the stored data can be directly accessed using SQL, R, Python, and other languages for BI, SQL analytics, real-time data applications, data science, and machine learning use cases.

The third architecture is developed by Lavrentyeva and Sherstnev (2022). This architecture also starts with a layer indicating different data sources and formats. However, then, the architecture shows that through ETL processes data is stored in a database. Then there is a compute layer indicating compute, ACID transactions, data filtering, and security can be implemented with and on the data. Then there is an API layer to connect a use case with the stored data. No specific use cases are mentioned, only that the data can be consumed through an API layer. This architecture is remarkable given that it mentions ETL processes are needed to ingest the data, and the storage object is a generic database and not specifically a data lake. Still, the architecture shows in general which components are needed to construct a data lakehouse architecture.

Finally, the articles by Schneider et al. (2023, 2024) analyse prevalent definitions for data lakehouses and derives eight technical requirements. Though these studies do not propose a new architecture as such, the requirements they present can be considered as providing a foundation for constructing data lakehouses and hence can be considered as having more or less similar scope to the above studies. The derived requirements cover the spectrum of activities related to ETL processes (e.g. requirements 1 to 4 in Schneider et al. (2023)), and ACID compliance (e.g. requirements 5 to 8 in Schneider et al. (2023)). The authors evaluate these requirements by applying them to popular data management tools to check if these tools allow creation of data lakeshouses based on these new requirements.

To summarize, all three architectures contain the same components, although it has been presented slightly differently for each architecture. Nonetheless, it became very clear that the architecture of a data lakehouse consists of 5 components: 1) data sources, 2) data storage, 3) compute layer, 4) APIs and 5) use-cases for data consumption. Moreover, the way in which the data lakehouse overlaps with the data warehouse and the data lake is shown by the fact that the storage layer is a data lake. And the data warehouse is represented by the layer that enables data management and governance features.

2.1.4 Summary of key features of the data warehouse, data lake and data lakehouse architectures

Summarizing the three main storage architectures presented in previous sub-sections, a high-level comparison is presented in this section. In the literature, only one architecture was presented for the data lakehouse. However, it is not as fine-grained like the architectures that explain the data warehouse or data lake. Therefore, this research aims to understand the evolution of architectures and how they have been constructed so that a a fine-grained and sophisticated architecture for the data lakehouse can be developed. Table 1 presents a high-level comparison between the three storage architectures (Lavrentyeva and Sherstnev 2022; Kutay 2021a, b), and (Oreščanin and Hlupić 2021). This comparison already contributes to building foundational knowledge on how the three storage solutions differ according to several aspects.

3 Methodology

The focus of this research is to discover the secure value (special capabilities) of data lakehouse architecture with respect to the data warehouse and data lakes. Based on the above focus and the nature of this research, it is important to select the most suitable research methodology when building the reference model. Among several design science research methodologies, the engineering cycle by Wieringa (Wieringa 2014) was selected as it has more logical interrelations among phases with respect to our research. The Peffers's method (2007) also has more similar steps as in Wieringa (2014), however, it doesn't provide an opportunity to validate the artefact before implementing. We believe it is important to validate the artefact before implementing it as it opens an opportunity to refine the artefact based on the validation.

Figure 2 shows the phases of the engineering cycle proposed in Wieringa (2014) which should be read clock-wise. The first phase is the problem investigation that explores the problem/phenomena, stakeholders, and goals. The second phase is the treatment design, where an artefact is developed to treat the problem that was identified in the first phase. The treatment validation phase, which is the third phase of the engineering cycle, concerns the validation of the designed artefact. The goal in this phase is to examine whether the designed artefact will solve the identified problem or not. When the results of the validation process are positive, the artefact moves to the treatment implementation phase. In this phase, the problem is treated with the artefact. Finally, the implementation evaluation takes place, where the success of the implementation is evaluated. This may lead to another iteration through the engineering cycle.

According to the above approach, firstly, the necessary knowledge to investigate the problem is collected through a literature review. Secondly, the knowledge collected through the literature review is used to design the artefact in the design phase. Finally, experts are interviewed to validate the artefact in the treatment validation phase. Due to time constraints, this research does not implement the designed artefact. Hence, the first three phases of the engineering cycle are covered.

Table 1 High-level comparison between the three storage solutions

Attribute	Data Warehouse	Data Lake	Data Lakehouse
Data Types	Structured data and processed data	Structured, semi-structured, and unstructured raw data	Structured, semi-structured, and unstructured raw data
Data Format	Closed, proprietary format	Open format	Open format
Purpose	Optimal for data analytics and business intelligence(BI) use-cases	Suitable for machine learning(ML) and artificial intelligence(AI) workloads	Suitable for all use-cases (data analytics, BI, ML and AI workloads)
Cost	Storage is costly and time consuming	Storage is cost-effective, fast, and flexible	Storage is cost-effective, fast, and flexible
Users	Business professionals	Business analysts, data scientists, data engineers, and data architects	Everyone in the business environment
Scalability	Scaling might be difficult because of tightly coupled storage and compute	Scaling is easy and cost-effective because of the separation of storage and compute	Scaling is easy and cost-effective
Agility	Less agile, fixed configuration	Highly agile, adjustable configuration	Highly agile, adjustable configuration
Analytics	Reporting, BI, dashboards	Advanced analytics	Suitable for all types of analytics workflows, both advanced analytics and BI
Ease of use	The fixed schema makes data easy to locate, access, and query	Time and effort are required to organize and prepare data for use. Extensive coding is involved	Simple, interfaces are provided that are similar to traditional data warehouses together
Processing	Schema-on-write	Schema-on-red	Schema-on-write and Schema-on-read
ACID compliance	Records data in an ACID-compliant manner to ensure the highest level of integrity	Non-ACID compliance: updates and deletes are complex operations	ACID-compliant to ensure consistency as multiple parties concurrently read or write data

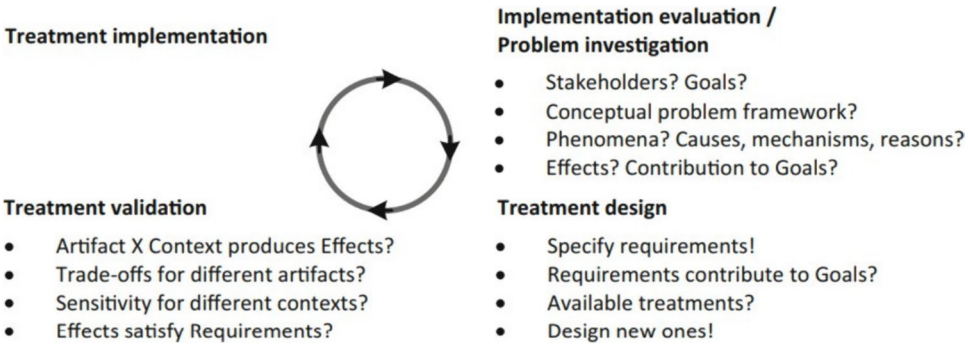

Fig. 2 Engineering cycle (Wieringa 2014)

3.1 Application of design cycle research methodology

Figure 3 demonstrates how this research adopted the engineering cycle methodology to solve the research question. To explore the secure value provided by the data lakehouse architecture, during the problem investigation phase, we use the results of the narrow-focused literature review performed to gather in Janssen (2022). In (Janssen 2022) we used a set of search queries to guide the search and seek pertinent sources that would aid in obtaining articles for analysis, and a set of inclusion and exclusion criteria to decide on which materials were to be included.

In the treatment design phase, we focus on developing a conceptual model to present the evolution of data storage architectures based on the literature review in Janssen (2022), while showing the value of each architecture. The focus of the treatment validation phase is to validate the conceptual model through, interviews with experts to obtain in-depth information from the practice and refine the conceptual model.

3.2 Limitations of methodological choices

According to Janssen (2022), one of the limitations of the literature analysis is including the Google Scholar and the Google Search Engine as a database. This is seen as a risk due to the inclusivity and zero-boundary principle that is enforced by these databases. As a result, sources may be incorrect, unreliable, or of poor quality. These risks are however mitigated through the use of inclusion and exclusion criteria in Janssen (2022). The criteria focused for instance on the credibility of authors, relevancy of the sources, whether they were cited, and what sources were used to substantiate claims.

Secondly, the analysis of data generated from interviews is labour intensive and challenging because of the variety of answers. As a result, the analysis could have some inconsistencies. However, this is dealt with by having a structured set of interview questions and following a table and specific structure to organize the results.

Thirdly, a limitation of this study is related to the sample size of 5 experts. However, while deciding on who to include in this study, we looked at the saturation point at which we believed no new information would be obtained anymore (Dworkin 2012; Mason, et al. 2010). This saturation concept is an essential factor for qualitative research and is defined as "when gathering fresh data no longer sparks new theoretical insights, nor reveals new properties of your core theoretical categories (Charmaz 2006)".

Finally, with conducting interviews, there is always a human-related risk that the experts' viewpoints can be biased, or the interviewer interprets the answers differently than it was intended. These risks were however reduced by having structured factual interview questions, avoid vendor-specific sources, carry out validation checks during the interviews, and recording and transcribing the interviews. More specifically, as a means of handling individual bias of

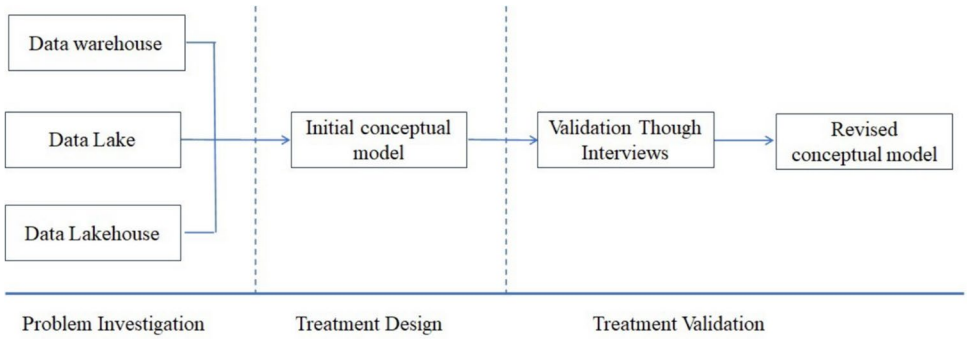

Fig. 3 Adopting engineering cycle

experts to a certain extent, they were asked to base and elaborate their responses upon the literature sources that were shared with them. Additionally, when the experts were identified as biased towards a specific vendor, they were asked if they could focus on other vendors in providing their reviews and when this wasn't possible, additional research were done to validate their inputs before they were considered in the evaluation process.

4 Developing data storage evolution model

Over the years, the requirements related to the capacity and performance of data storage architectures have increased mainly due to the volume and complexity of both the data and the applications that process these data. To cater these evolving requirements, as discussed in the previous sections, different data storage architectures were developed. To understand the value provided by each of the data storage architectures, we first need to analyze how the evolution of data storage architectures has happened over the time. One way to do this analysis is to create artefacts that support capturing the essence and value provided by each generation of the data storage architecture.

Thereby, in Sect. 4.1, we first analyse data warehouse, data lake, and data lakehouse architectures to enlist their respective strengths and weaknesses and then present the data storage evolution model developed by highlighting how a weakness of one architecture is addressed by the respective strengths of another during the evolution in Sect. 4.3.

4.1 Strengths and weaknesses of datawarehouse, data lake and data lakehouse architectures

4.1.1 Strengths and weaknesses of data warehouse architecture

Strengths Since it's introduction in 1980s, a number of research studies have been done to investigate and improve the design concerns such as simplifying the integration of multiple data sources, access to the database, data enrichment, and automated procedures related to data warehouse architecture. Through them, it is possible to identify numerous benefits of the data warehouse architecture.

As data warehouse stores structured data, one of its main advantages is its ability to conveniently mine and analyse data. As a result, business intelligence is enhanced, and the overall decision-making processes are improved due to the support of analysis reports generated from the data stored in a data warehouse. This means that the data warehouse architecture provides improved quality and increased quantity of information (Roelofs et al. 2013; Watson et al. 2002) and the high-quality data warehouses lead to a high level of user satisfaction and increased productivity in decision-making (Al-Okaily et al. 2022; Shiyal 2021). Thus, *enhancing the creation of business intelligence, and improving the decision-making processes* can be regarded as strengths of data warehouses.

One of the technical strengths of a data warehouse is its ability to perform transactions *supporting atomicity, consistency, isolation, and durability (ACID)* (Chen et al. 2002). The implementation of ACID transactions contributes to the guarantee of keeping the data reliable, consistent, and integral which is one of the key aspects in information security. Thus, we identify the *support to ACID transactions* as a strength of data warehouses. Additionally, data warehouse has metadata management that allows obtaining information about the data stored in a data warehouse. These *metadata management controls* allow for efficient access to the data. Further, metadata makes it possible to have *version control and access to historical data* (Jarke et al. 2002). Therefore, we identify that *metadata management mechanisms*, and *version control and access to historical data* as strengths of the data warehouses.

Gosain and Arora (2015) argue that since a data warehouse could store a lot of data, which could be sensitive as well as with a multiple years life span, securing data is crucial for the sustainability and reliability of the data warehouse. Over the years, security controls have become more and more enhanced and sophisticated in data warehouses (Gosain and Arora 2015; Rosenthal and Sciore 2000), and (Vishnu et al. 2014). Therefore, the *granular level of security* is an important strength of the data warehouse.

Weaknesses Whilst data warehouse is a robust and stable storage solution, it still holds a few weaknesses. Firstly, it *lacks flexibility* in the sense that it is incapable of storing semi-structured and unstructured data (Janssen 2022). Before the data is stored in a data warehouse, it is required to go through a process to extract, transform and load (ETL) data. However, with the rise of Big Data, it is hardly possible to develop ETL processes that fit any data format. Traditionally, data warehouses are implemented on-premises and in return, the company would have more control over how and where data is stored and there is no reliance on high-speed internet and connectivity to ensure low latency. However, this requires *high implementation costs* including the need to have on-site IT staff, location/space for machines, and operational costs like electricity, etc. Moreover, there will be high regular maintenance costs to keep the data warehouse up to date which may even exceed the initial implementation cost (Adelman 2021).

Thus, *lack of flexibility, high implementation cost*, and *high maintenance cost* can be identified as weaknesses in data warehouses.

4.1.2 Strengths and weaknesses of data lake architecture

Strengths One of the key strengths of data lake architecture is its support for storage of heterogeneous data (Begoli et al. 2021; Hassan 2024) which solves one of the apparent weaknesses of data warehouses. Data lakes support accumulating data from heterogeneous sources are and mostly associated with Hadoop ecosystem (Mehmood et al. 2019). Therefore, we identify that data lakes support for storage of heterogeneous data as one of the advantages of a data lake.

Given the rise of big data and thus a wider variety of data formats and a higher volume of data, the data lake is designed in such a way that data can be stored in its raw form. As the data lake focuses on storing a wide variety of data formats in a flat architecture, advanced analytics and data science techniques are better supported. As an example, real-time analytics is possible since storing and accessing real-time data is enabled. Moreover, data lakes are capable of handling batch processing, and various machine learning techniques can also be used to analyse data (Fang 2015; Madera and Laurent 2016), and (Mehmood et al. 2019). Therefore, we identify the support for advanced analytics and data science techniques as a strength of the data lake architecture.

Furthermore, it is a cost-effective storage solution given the immense growth in volume of data and since storing costs have become more important than before (Fang 2015; Sawadogo and Darmont 2021), and (Hassan 2024). Data lake is designed to be an object-based storage that stores vast amounts of unstructured data (Sawadogo and Darmont 2021). This is usually optimized for a lower cost per Giga Byte (GB) stored there. Further to this, a data lake is always designed in the cloud and therefore there are no up-front implementation costs. Additionally, less CRUD (create, read, update and delete) operations are necessary because all the data is stored in its raw format and the data lake architecture is easy to scale, since it is an object storage in the cloud, and these can be identified as strengths. Finally, data lakes facilitate easy access to the data since everything is stored in one central repository where multiple users can access the data simultaneously for monitoring, exploration and analysis (Mehmood et al. 2019).

Weaknesses One of the key weaknesses of data lakes is that it is very challenging to address general requirements for metadata management over raw data (Mehmood et al. 2019). Though several attempts have been made, due to the varying data formats, it is difficult to have proper data governance and perform metadata management in a uniform way (Madera and Laurent 2016). Another weakness of data lake is that applying appropriate security controls appears to be very challenging. Again, this is caused by the varying data formats because security controls have to consider the specific data format. Though there is some security control in place for access control, the study by Mehmood et al. (2019) suggests that this is still insufficient to protect the data against different types of attacks and that the only way of securing a data lake is by providing access only to white-listed IPs.

The lack of metadata management and low-security assurance in data lakes leads to a high risk of turning into a data swamp (i.e., messy data) (Nargesian et al. 2019). Since, any type of data can be dumped into the data lake, there is no control of what is being stored in there. There is a possibility that the data is corrupt or that the data is never being used but still stored in the data lake. This leads to the weakness of holding poor quality and unreliable data. Finally, the performance level for all the different workloads is inconsistent in the data lakes. Though flexibility is a strong point of the data lake, the performance for all the different workloads is inconsistent because of all the different types of data formats that are being used. Moreover, a data lake is not organized, it is just a central repository where data is dumped. Hence, finding the necessary data for a specific workload can be inconvenient and in certain cases, it will be more convenient. Due to this, the performance level of data lakes could be inconsistent.

4.1.3 Strengths and weaknesses of data lakehouse architecture

Strengths Given that the data lakehouse architecture combines the features of the data warehouse and data lake architectures, it inherits some of their strengths. For instance, like data warehouse, data lakehouse delivers business intelligence allows the implementation of metadata mechanisms, and supports decision-making processes, ACID transactions, data versioning, and indexing. The overlapping strengths between data lakehouse and data lake include the support for the storage of heterogeneous data (Shiyal 2021), advanced analytics and data science techniques (Schneider et al. 2024), and provides a cost-effective storage solution given that the data lake is the storage object.

Besides the overlapping strengths discussed above, there are strengths that are unique to the data lakehouse architecture. Firstly, it supports a wider variety of workloads since it is not optimized just to support for BI and reports, or data science and advanced analytics techniques. In fact, all workloads are supported including modern workloads such as real-time streaming and artificial intelligence

use cases. Another strength of the data lakehouse is the reduction of the level of data redundancy compared to a data lake. This is achieved through a lightweight filter that duplicates the data and only stores data relevant to the business data demands. Additionally, the stored data is of high quality, reliable, and consistent. This results from the meta-data layer that supports management features like ACID transactions, proper data governance controls, and security measures. This can be implemented in the data pipeline to improve the data quality (Armbrust et al. 2021). Having security measures in place is another strength of the data lakehouse. In data lakes it is hardly possible to implement security measures in an effective way. On the contrary, the data lakehouse can implement ACID transactions (Schneider et al. 2024; Errami et al. 2023), a metadata layer, and audit logging which enhances the security of the data (Armbrust et al. 2020). However, it is not possible to implement fine-grained security measures as for data warehouses given that there are different data formats in data lakehouses. Furthermore, multiple optimization techniques are supported to further enhance the management and performance of the data. Examples of such techniques are caching, auxiliary data structures, and data layout which all lead to optimized SQL performance in terms of velocity and accessibility (Armbrust et al. 2021, 2020), and (Harby and Zulkernine 2022).

Weaknesses Although this technology is relatively new, in this study, we man- aged to identify several weaknesses of the data lakehouse architecture. First of all, data lakehouses are perceived to be not very mature since it was first introduced in 2020. Compared to over 40 years of existence of the data warehouse and 10 years of the data lake, the data lakehouse is very new. Hence, the data lakehouse is still updated and improved which means that there is no guarantee that it will live up to its promised advantages in its inception (Kutay 2021a). Moreover, not much research has been done on the data lakehouse architecture and there are limited implementation use cases available. This is perceived as a weakness since this indicates that this storage solution has not yet been exploited to the fullest and not many updates have been done due to the fact that there are limited use cases that can be used as a source of referral to see where the improvements can be made. Furthermore, in the data lakehouses, the latency is dependent on the chosen underlying cloud object storage, and this is one of the weak- nesses in the data lakehouse architecture. Finally, using data lakehouse architecture may require new skills and training. Quite often, some people are reluctant to adopt new technology, therefore this could also be identified as an anticipated weakness.

Table 2 summarizes the strengths and weaknesses of the data warehouse (DW), data lake (DL) and data lakehouse (DLH) architectures.

4.2 Drivers for transitions between data warehouse, data lake, and data lakehouse architectures

As summarised in Table 2, all three architectures have their own strengths and weaknesses that lead to evolve data storage architecture solutions and transitions between them and hence the aim of this section to briefly explore the drivers of these transitions. We argue that the primary drivers of these transitions are the weaknesses of them that lead to explore new architecture solutions whilst strengths are enablers to use them. One of the primary drivers for the transition from data warehouses to data lakes is the complexity of challenges offered by the big data era (Harby and Zulkernine 2022). The lack of flexibility to handle complex unstructured real time data using the ETL pipeline in data warehouses acted as a key driver for using data lakes that offer flexibility in organising and handling unstructured data through different means such as data ponds, and defining metadata to describe data (Ravat and Zhao 2019b; Inmon 2016). In addition to this, other weaknesses of data warehouses such as high implementation and high maintenance cost (Sawadogo and Darmont 2021) seemingly due to factors such as proprietary data warehouse systems, non-support to open source and cloud-based data analytic tools (Harby and Zulkernine 2022), and on-premise implementations (Janssen 2022) have led to organisations resort to data lakes as a viable alternative that provides different storage solutions, high scalability, and low operational costs (Schneider et al. 2024). Additionally, this transition to data lakes was also fueled by data warehouses not supporting advanced analytics with real-time data (Schneider et al. 2024).

With the modern high data volumes and analysis requirements to support business intelligence capabilities keep growing, the metadata management mechanisms, data governance and auditing mechanisms, compliance to data protection standards provided by data lakes identified as not optimal (Oreščanin and Hlupić 2021; Jain et al. 2023). With those sub-optimal features in data lakes, more interest has been focused on data lakehouses that combine strengths of data warehouses and data lakes to provide features such as improved data quality through ACID compliance and data governance to support handling of both structured and unstructured data (Oreščanin and Hlupić 2021).

The key drivers of these transitions from data warehouse to data lake and from data lake to data lakehouse are modeled in Fig. 4 by mapping the weaknesses of the predecessor to the strengths of the successor.

Table 2 Strengths and weaknesses of the DW, DL, and DLH

Architecture	Strengths	Weaknesses
Data Warehouses	Business intelligence capability, Improve decision-making and business processes, ACID transactions, Metadata management mechanisms, Robust and stable solutions, Versioning and access to historical data, Security measures	Lack of flexibility, High implementation costs, High maintenance costs
Data Lakes	Capability to handle heterogeneous data, Cost-effective, Easy to scale, Easy access to data, Advanced analytics and data science techniques	Lack of metadata management, low-security assurance, risk of turning into a data swamp, poor quality and reliability of data, inconsistent performance levels
Data Lakehouses	Business intelligence capability, Improve decision-making and business processes, Metadata management mechanisms, Handle heterogeneous data, Cost-effective, Advanced analytics and data science techniques, Reduced data redundancy, High-quality, reliable, and consistent data, Wide variety of workloads, Real-time data, High-security assurance, ACID transactions, versioning, and indexing, Optimizations techniques like caching, auxiliary data, and data layout, Management and performance features	Not very mature, limited research and use-cases available, new skills required, dependent on underlying cloud-object

4.3 Designing the data storage evolution model

The Data Storage Evolution Model presented in Fig. 4 aims to capture the essence and value of the data warehouse, data lake, and data lakehouse architectures. Moreover, we believe this model would help researchers and practitioners to understand the data storage architecture evolution process. Below, we outline the reasons for developing the model.

- In literature, there is no conceptual model that compares the existing storage architectures and explains their relative strengths and weaknesses.,
- In particular, due to the novelty of the data lakehouse architecture a limited number of research studies are available that investigate its relative value as a data storage architecture to help users understand its true value of it.
- In practice, no tool or model is available to explain the data lakehouse as an option when choosing a data storage architecture that better fits with a particular use-case.

To summarize the designed model, below we use the design template proposed in Wieringa (2014).

Compare *the existing storage architectures.*

By *creating an artefact that summarizes the strengths and limitations of each storage architecture.*

That *enhances and increases the awareness and knowledge of researchers and practitioners.*

In order to *clarify the evolution process and highlight how the newest architecture overcomes the limitations of the existing architectures.*

The model presented in Fig. 4 consists of three separate entities, each representing one of the three storage architectures: Data Warehouse, Data Lake, and Data Lakehouse in orange colour. In each entity, strengths are listed above the orange colour boxes and weaknesses are listed below them. Finally, relationships are modeled by indicating how the strengths of one architecture deal with the weaknesses of another, for instance, which strength(s) of the data lake addresses the weakness(es) of the data warehouse and hence act as drivers of transition from latter to former (see Fig. 4).

5 Treatment validation

Following the research methodology we presented in Sect. 2.1, in this section, we discuss the treatment validation process followed and improvements were done to the different components of the Data Storage Evolution model presented in Fig. 4.

Based on our literature review, the strengths and weaknesses of each data storage architecture were identified and presented in Sect. 4.1. These results were validated with

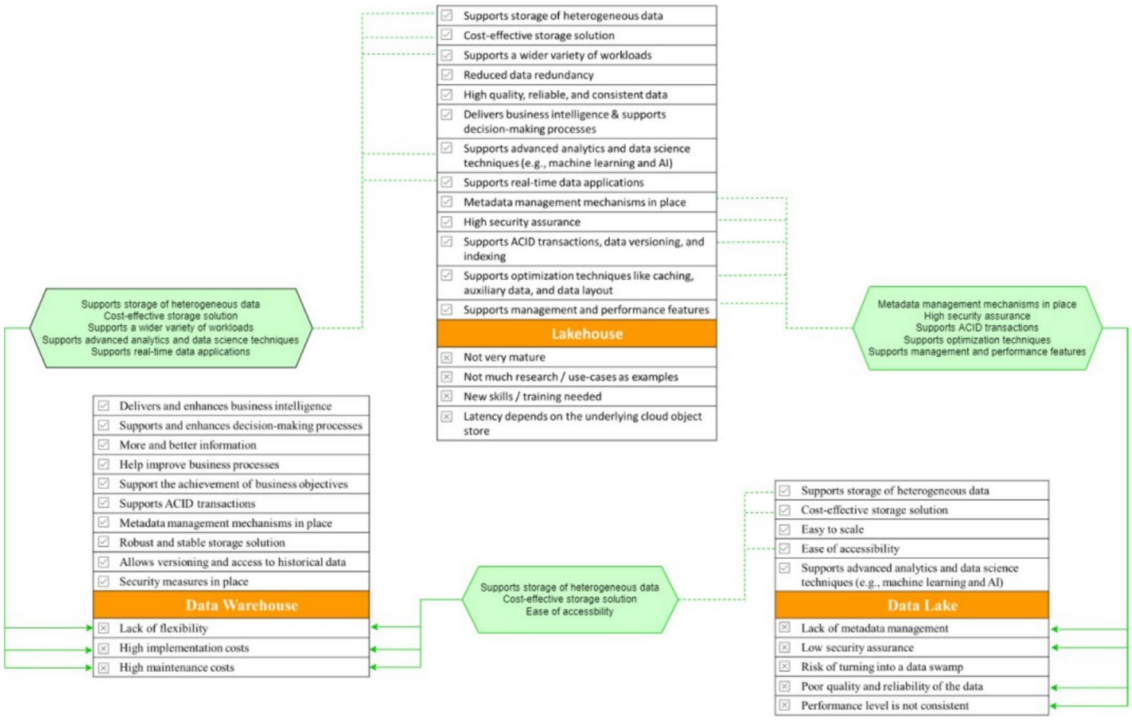

Fig. 4 Data storage evolution model

experts and, in this study, five experts were interviewed to gather insights from practice. Figure 10 in the Appendix contains an overview of the outcome of the expert validation. Given the smaller sample size, as a means to address concerns related to bias to a best possible extent, the expert's suggestions were considered against findings from the literature before accepting or rejecting them. Any concerns of contradicting arguments were discussed and negotiated to arrive conclusions.

Each row in the Fig. 9 represents a strength or a weakness of each storage architecture, and the columns represent the responses of the five experts. The strengths and weaknesses are represented as numbered assumptions (which are listed in Tables 5, 6, and 7 in the Appendix) beginning with the letter "A" (e.g.A1). Each expert respondent's response to the *assumptions* (strength or weakness) indicated by the terms: *accepted* (accepted the strength or weakness), *adjusted* (rephrased the strength or weakness to make it more meaningful), *rejected* (rejected the identified strengths or weakness), or *depends* (the particular assumption's validity depends on certain factors).

The following sub-sections provide a comprehensive discussion about the reasons for rejections and proposed adjustments for the assumptions. As the sample size is an odd number, we consider an assumption as a rejected if and only if three or more respondents reject the assumption. As for the adjustment, depending on the suggestions, one response is enough to adjust an assumption because we believe that it will increase the clarity of the model. Further, it describes which factors are affected by some of the assumptions which are denoted as "Depends" in Fig. 9. In addition to the listed assumptions, new strengths, and weaknesses are also identified by the experts during the validation processes. The summary of the expert validation is given in Table 3 where storage architectures are represented with the abbreviations DW, DL, and LH representing data warehouse, data lake, and data lakehouse, respectively. Further + and – signs are used to denote strengths and weaknesses, respectively. For instance, the strengths of data warehouses listed as A8 (robust and stable storage solutions) and A10 (have security measures in place) are accepted by 100% of the experts while A3 (provide more and better information) is rejected by 60%.

5.1 Insights into data warehouses

During the validation, the experts suggested several adjustments to the strengths and weaknesses listed in the data warehouse entity while agreeing to keep the majority of them the same. An overview of all the adjustments is presented in Fig. 5.

One of the significant changes by the experts in the validation is the removal of "more and better information" which we have originally assumed as a strength of a data warehouse. While it is possible to have more information given that a lot of historical data can be stored, the experts were of the view that quantity does not imply quality. They

Table 3 Validation summary of the assumptions by experts

+ or -	Accepted	Depends	Adjusted	Rejected
DW +	A8,A10: 100% A2,A4,A5,A7:80% A1:60%,A3:40%	A1: 20%, A2,A4,A5,A7:20%	A1: 20%	A3:60%
DW -	A11: 60%, A12, A13:100%		A11: 40%	
DL +	A14, A16, A18:100%, A15:60% A17:40%	A15, A17:40%		A17:20%
DL -	A21:100%, A19:80%, A23:60%, A22:40%	A20:20%, A23:20%, A22:20%	A20:60%, A19:20%, A23:20%	A20:20%, A22:40%
DLH +	A24,A25,A26,A28, A30,A31,A34:100%, A29, A35, A36:80%, A27, A32:60%, A33:40%	A29,A35, A36:20% A33:20%	A27:20% A33:20%	A27:20%, A33:20%
DLH -	A38, A40:100%, A37, A39: 40%		A37:60%	A30:60%

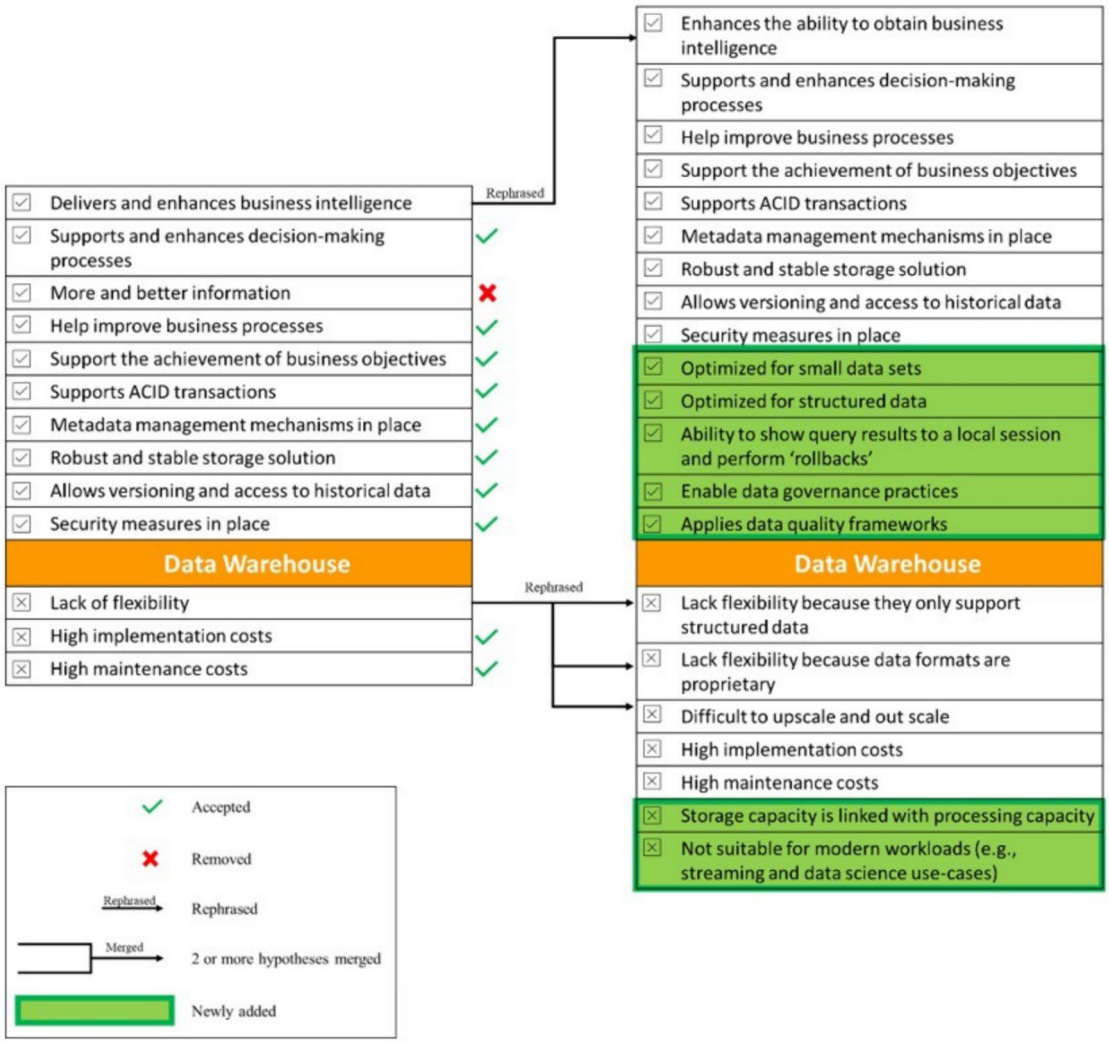

Fig. 5 Revised entity for data warehouses

further argued that the data quality heavily depends on the data extraction, transform, and load (ETL) processes. Considering the validity of their argument, this was delisted from the model. In addition to that, based on their suggestions, one of the strengths; delivering and enhancing business intelligence, was rephrased for more clarity, and the weakness; lack of flexibility, was split up into three distinct weaknesses (refer Fig. 5) which are in line with the strengths and weaknesses discussed in detail in literature (Schneider et al. 2024; Harby and Zulkernine 2022).

Besides the adjustments discussed above, several new strengths and weaknesses were introduced during the validation process. Firstly, experts were of the view that data warehouses are known to be optimized for small and structured data sets thus, optimized for small data sets, and optimized for structured data sets are added to the model as strengths of to the data warehouse entity. According to Harby and Zulkernine (2022), variants of data warehouses such as data marts are used to contain smaller subsets of structured data for faster and focused query processing and hence the expert views can be considered as sound suggestions. Furthermore, the experts were of the view that next to having metadata management controls in place to manage the data, it is also possible to have data governance controls in place. Also, one of the supported features, that is often used in practice, is 'rollbacks. This feature allows the user to view the result of data operations in a local session before pushing and updating the actual database. Thereby, the ability to show query results to a local session and perform rollbacks was added as a strength to the model. This prevents pushing bugs and incorrect data to the database and enhances the quality of the data. Finally, in order to maintain and enhance data quality, data warehouses also support the implementation of quality frameworks. This can be incorporated in the schema-on-write structure so that the moment when data arrives in the database it already adheres to a minimum level of quality standards. Thereby, two more strengths; enabling data governance practices, and applying data quality frameworks were added to the model under the data warehouse entity.

In addition to the above strengths, the experts also proposed two weaknesses to be added to the model. Firstly, they were of the view that the storage capacity which links with the processing capacity as a weakness. This, they argued as a weaknesses because in order to scale the storage capacity, the processing capacity also needs to be scaled. This entanglement is very costly and complex when it is desired to upscale the data warehouse for instance in Oreščanin and Hlupić (2021) this weakness is identified as"expensive for large data volumes". Thereby, two more weaknesses; storage capacity is linked with processing capacity, and not suitable for modern workloads were added to the mode.

5.2 Insights into data lakes

Similar to the data warehouse, several adjustments were introduced to the data lake by the experts. A summarized overview of these changes is presented in Fig. 6 (newly

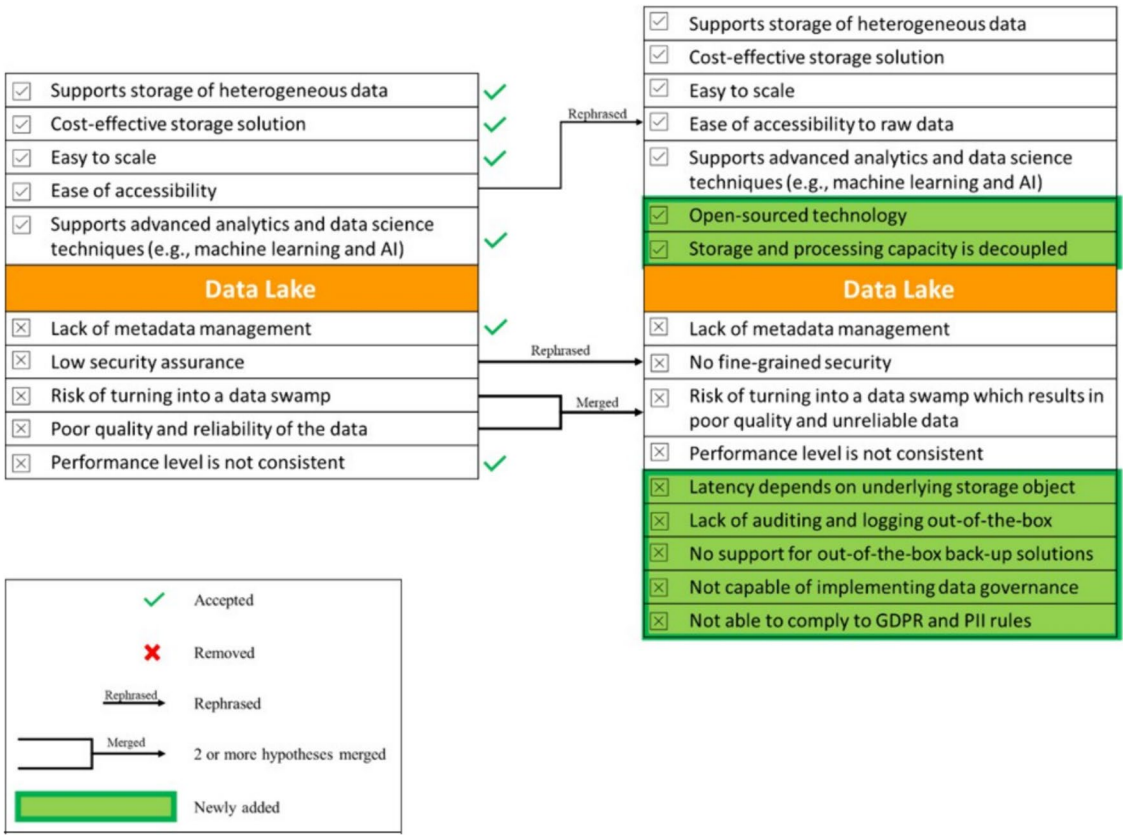

Fig. 6 Revised entity for data lakes

added strengths, and weaknesses highlighted in green). Firstly, the experts agreed that none of the assumed strengths or weaknesses needed to be eliminated from the model. However, they suggested that two assumptions need to be rephrased to make them more specific, and two weaknesses should be merged into one due to a causal relationship between them.

Experts shared the view that open-source focus of data lakes should be added as a strength of data lakes. This means that anyone can implement and adjust a data lake's source code. Thus making the data lake very flexible (Schneider et al. 2024; Harby and Zulkernine 2022) and enabling the user to customize their data lake to suit their business requirements. They further suggested adding the decoupling of the storage and processing capacity as a strength in data lakes which is also highlighted by Schneider et al. (2024) as providing different storage options to manage raw data through extract and load processes where transformation could happen later depending on analysis requirements. This was a huge disadvantage in data warehouses, and therefore, the second generation of data storage architecture is designed in such a way that they are decoupled. This increases the level of flexibility and reduces the costs when scaling the data lake.

Further, five new weaknesses of data lakes are also identified by the experts. One is the dependence of latency on the underlying object storage. Object storage in here is a storage architecture that handles large amounts of unstructured data. The second weakness is the inability to do auditing and logging out-of-the-box with a data lake. Having this capability helps to protect the systems' integrity. The third weakness is that the data lakes do not support backup solutions out of the box. This is one of the serious disadvantages and currently, the data lakes are practicing custom-built backup solutions to copy data into different storage accounts. Another weakness raised by most of the experts is not having good data governance practices in data lakes. Finally, experts were also with the view that difficulties in adhering to GDPR (General Data Protection Regulation) and PII (Personally Identifiable Information) rules can be seen as a weakness in data lakes which makes a data lake very vulnerable to security threats.

5.3 Insights into data lakehouses

A summary of all the adjustments to the strengths and weaknesses of the data lakehouse architecture is shown in Fig. 7. As shown there, two original assumptions were excluded from the model based on the experts' validation. One originally assumed strength that the data lakehouses provide high-security assurance was rejected by the experts. They argued that the storage object that is used for a data lakehouse is a data lake and thereby, in essence, the security levels are the same for the data lakehouse and a data lake. However, because of the metadata management and governance layers are implemented on top of the data lake, the data is more secure in a data lakehouse with compared to a data lake. Still, the fact remains that data lakehouses do not provide high-security assurance. Therefore, this assumed strength was excluded from the model, and a new weakness; there is no support for fine-grained security is added. Secondly, the weakness that new skills and training are required for the implementation of data lakehouses was also excluded from the model. The experts were of the view that while it is indeed required to understand some new principles and concepts, having experience with either data warehouse or data lake is sufficient for working with data lakehouses.

Further, based on the experts' validation, three existing assumptions were rephrased to make them more precise, and four new strengths and three weaknesses were added to the data lakehouse entity. The first strength that was added is that data lakehouses are open sourced which experts saw as an advantage because the data lakehouse can be customized according to specific business requirements. Secondly, it is possible to choose any computing language to implement and configure a data lakehouse which enhances flexibility. Thirdly, given that the storage object is a data lake, the storage and process capacity for a data lakehouse is also decoupled. Hence, when scaling a data lakehouse, there are no complexities in terms of interwoven and storing and processing capacity. Finally, a unique strength of the data lakehouse is that it supports data governance out of the box with the metadata and governing layer on top of the data lake.

Additionally, two new weaknesses were included based on the expert suggestions. As previously mentioned, one of the newly introduced weaknesses was that data lakehouses do not provide fine-grained security. Compared to data lakes, the security levels in the data lakehouses are relatively improved, however, it is not as fine-grained as in data warehouses. Finally, there is only one data lakehouse provider which means that there is a risk of vendor lock-in. Even though one of the strengths is that it is open-sourced hence there is no tight lock-in. Still, due to the fact that Databricks is the only vendor, it is perceived as a current weakness of data lakehouses.

5.4 Revised data storage evolution model

In the final stage of validation, the experts were asked to evaluate how the model was constructed. In particular, they were asked to comment on the way in which the entities were put into context, the level of clarity, ease of understanding, and how the relationships have been modeled. Based on their comments, a list of specifications was formulated to design the revised Data Storage Evolution Model. Then the revised model was developed with expert validation. The model is presented in Fig. 8.

Fig. 7 Revised entity for data lakehouses

1. **Colouring:** Based on the expert suggestions, it is decided to visualize each storage solution with a different colour. It makes more convenient to visualize which advantage of one data storage architecture is related to other architectures, for instance, which strength of the data warehouse and the data lake is correlated with the data lakehouse.
2. **More emphasis on the message:** Experts suggested that to make more emphasis on the message, we need to boldface the icon displayed in front of the strengths and weaknesses. Having bold-faced icons, a check symbol for the strengths, and a cross symbol for the weaknesses, make them more readable to clearly identify what the strengths and weaknesses are.
3. **Improve relationships:** to improve the relationships between the entities, it was suggested to use a weakness as a starting point. From there, an arrow is drawn to the strength of the relevant storage architecture to indicate how the second architecture tackles and solves the weakness in the first one. Then, the message will also better come across because the emphasis is now put on how the subsequent storage architecture tries to solve the issues of the preceding architecture.

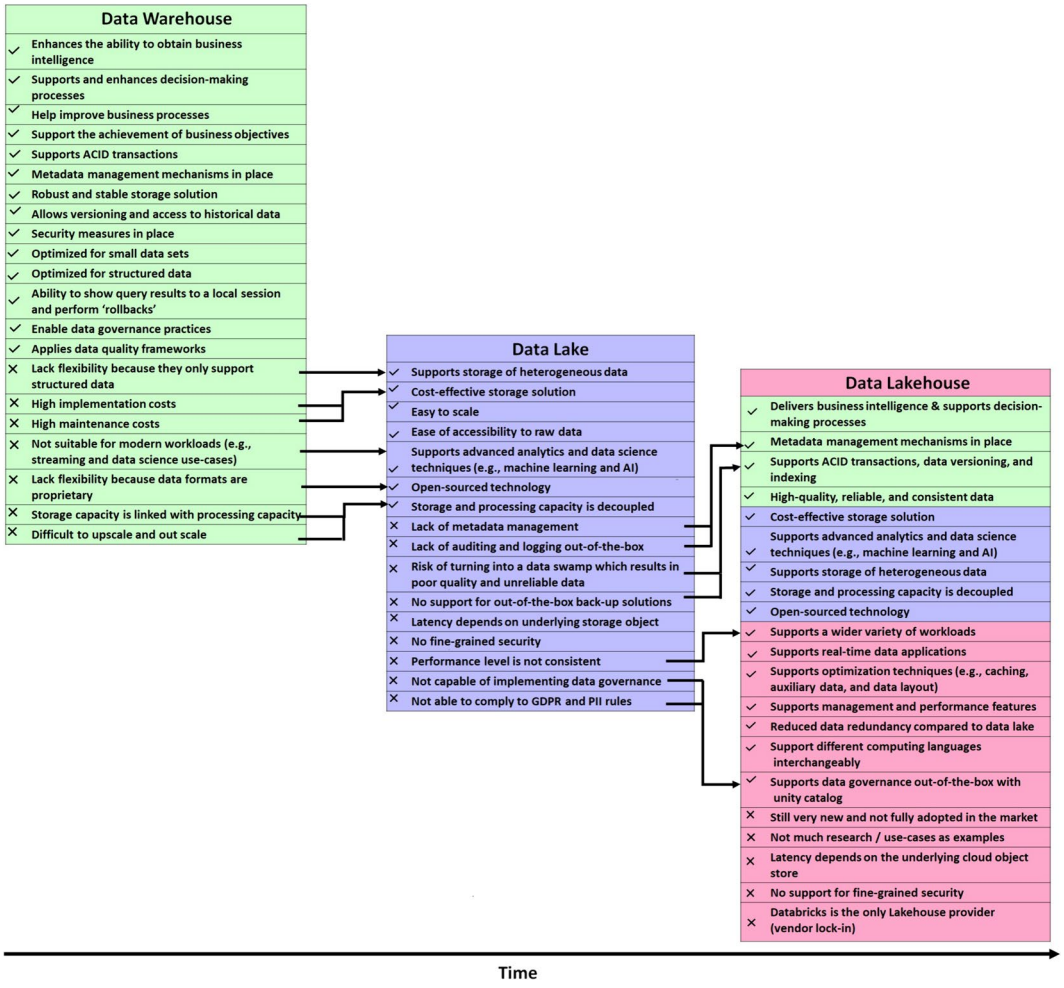

Fig. 8 Revised data storage evolution model

4. **Evolution from left to right:** Based on the insights provided by the experts, we have chosen to display the model from left. Then, it will be more apparent that there is an evolutionary process. Moreover, the experts were of the view that the message of showing how a subsequent storage architecture solves problems of the preceding architecture will be better expressed when the model is displayed from left to right.
5. **One high-level model and one detailed model:** Based on the expert comments, one high-level model was designed to visualize the concept behind the data lakehouse by combining data warehouse and data lake. Though this is not a requirement for the revised model, the high-level model serves as an introduction to the data lakehouse concept.
6. **Place naming's on top of the entity:** This improvement was recommended to enhance clarity since an entity is generally read from top to bottom. Therefore, it is more convenient to have the name of the storage architecture on top instead of in between the advantages and disadvantages.

One of the suggestions of the experts was to create a high-level model to present the idea of the lakehouse adopting best practices from the two previous data storage architectures. This is to visualise the relationships between the three data storage architectures on a higher level before the more detailed model would be presented. In Fig. 9, we model this by showing the overlapping strengths between the data warehouse and a data lakehouse, and the overlapping strengths between the data lake and data lakehouse architectures.

6 Discussion

6.1 Insights on existing data lakehouse architectures

In Sect. 2.1.3 three distinct architectures representing the data lakehouse were discussed and evaluated based on the explanations provided by Armbrust et.al. (2021), Inmon et al. (2021), and Lavrentyeva and Sherstnev (2022).

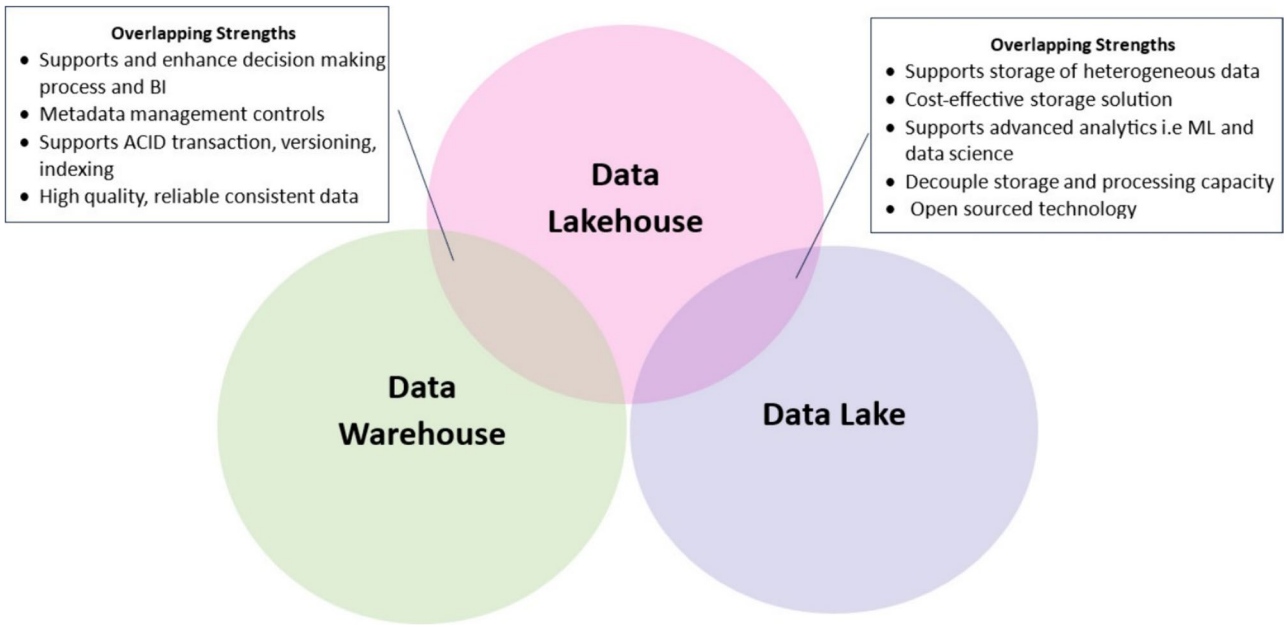

Fig. 9 Overlapping strengths of data lakhouse with two other architectures

Each of these architectures takes a different perspective and hence different functioning. Two out of five experts argued that the architecture presented in Armbrust et.al. (2021) best describes how the data lakehouse should be constructed. Further, these experts were of the view that this architecture contains all the essential parts necessary to build a data lakehouse architecture and clearly indicates the layers of the lakehouse architecture. As for the second and third architecture, generally, the experts were of the view that these represented more the data lake and not the lakehouse. Although in both architectures, an extra layer is put on top of the data lake, it is not sophisticated enough to explain what and how the construct represents the idea of the data lakehouse.

Two of the other three experts argued that the architecture proposed by Lavrentyeva and Sherstnev (2022) best represents the data lakehouse as it is straightforward and clear and aggregates the best features of the data warehouse and data lake despite missing data zones in the data lake. One of these experts was of the view that the first two architectures take data flow and data format perspectives, respectively, and hence do not precisely represent data lakehouse architecture. The second of these two experts argued that, as it is represented in the study, the storage layer of the architecture by Armbrust et al. (2021) can practically be any object storage while the purpose of the lakehouse is to have a data lake implemented where the data management and governance layer is built on top and hence does not well represent data lakehouse architecture. Additionally, the expert was also of the view that the architecture by Inmon et al. (2021) vague and minimalistic due to it having too little information to be clear and understandable and hence does not clearly show the concept of the lakehouse.

Finally, the remaining expert did not think that any of the analysed architectures best represent data lakehouse architecture all the architectures failed to answer the question 'what is the data lakehouse concept?'. The first architecture by Armbrust et al. (2021) is acceptable in terms of showing the implemented layers, and it is clear how the data flows through the architecture. However, no emphasis is put on what part of the architecture is uniquely related to the data lakehouse. The second by Inmon et al. (2021) shows that different data formats can be put into some storage, which can then be used via different APIs for machine-learning purposes. This is a very minimalistic view and not fine-grained at all. Most importantly, it does not show what part of the architecture is lakehouse-specific. Lastly, the third architecture by Lavrentyeva and Sherstnev (2022) is a very high-level overview of having different data formats and use-cases that utilize the data. However, it does not clearly show how this architecture can be distinguished from a data lake. Hence, This expert believes that, due to the above issues, all the architectures are not sufficient enough to explain the data lakehouse concept.

6.2 Challenges and current shortcomings of the data lakehouse architecture

Besides the promising features of the data lakehouse architecture, the experts also evaluated its challenges

and shortcomings. These findings were classified among people-related and technical-related challenges and shortcomings as depicted in Table 4. As a new technology, several challenges were identified related to people. With regard to people, one of the most challenging factors is the reluctance to learn new skills and stimulate adoption of this new technology due to various reasons including age related and technical competence (Barnard et al. 2013). Further, it is hard to mark aware that bad skills lead to a bad implementation instead of blaming the new technology, and finally understand when to implement this new technology so that it really solves a business problem. These challenges are common for most new technologies at the stage of introduction. However, the technical-related challenges are more data lakehouse specific.

First of all, compared to data lakes latency is higher because every transaction in the data lakehouse is recorded in a log. Secondly, as it is distributed storage it leads to a higher latency in general (Jain et al. 2023). Thirdly, compared to data warehouses the latency is higher in data lakehouses when dealing with small datasets due to factors such as storage format differences, metadata management, etc., leading to different data retrieval and update strategies as indicated in Schneider et al. (2024); Jain et al. 2023). Another technical-related challenge is the fact that there are not many proof and track records. Due to the novelty of this architecture, there are not many implementations hence it is not possible to learn from previous use-cases when dealing with bugs and errors that have never occurred before. The final challenge is how to maximize security when fine-grained security controls are not supported. The question is whether this will ever be possible, however, no technology is a magical solution that solves all existing problems. Hence, maximizing the security levels will remain a challenge for now.

6.3 Future perspectives

Two directions were examined to evaluate the future perspectives of the data lakehouse. Firstly, the experts were asked whether they believed a data lakehouse could replace data warehouses and/or data lakes. The outcome of this interview shows all of them believe that data lakes can be replaced by data lakehouses and 60% of them believe that data lakehouses will replace data warehouses in the future. Given that all the strengths of data lakes are incorporated in data lakehouses, with additional strengths that tackle certain weaknesses of data lakes, nothing would be lacking if a data lake is replaced by a data lakehouse.

As for the data warehouse, the experts had divided opinions. They were of the opinion that depending on the priorities and the business problem that is to be solved with the data storage architecture, one should be chosen. With regard to replacing existing data warehouses, it is very complex to convert existing files into the right format and the right version so that they can be stored in the data lakehouse. Also, when dealing with small datasets or only structured data, the data warehouse will definitely outperform the data lakehouse. Additionally, when the business requirement is to generate reports and business intelligence, the data warehouse will also outperform the data lakehouse. Hence, the choice depends on the type of use case.

The second direction that was examined was what the experts believe to be the next evolutional step. Among researchers in the technology field, the concept of 'data mesh' has gained quite some popularity. Also, the experts believed it to be the next generation of data storage architectures. The rationale behind a data mesh as a platform is that it is "distributed data products oriented around domains and owned by independent cross-functional teams who have embedded data engineers and data product owners, using common data infrastructure as a platform to host, prep, and serve their data assets. The platform is an intentionally designed distributed data architecture, under centralized governance and standardization for interoperability, enabled by a shared and harmonized self-serve data infrastructure (Dehghani 2019). Therefore, the idea of what data management means for organizations is redefined (Strengholt 2022). Instead of centrally managing data, specific domain teams become responsible for governing their data. The way the data lakehouse fits in this story is that the design of the data

Table 4 Challenges and shortcomings of the Data Lakehouse

People Related Challenges	Technology Related Challenges
Slow adoption at the clients	Latency is higher compared to the data lake
Reluctant to obtain new skills and capabilities	Latency is higher due to distributed storage
Poor skills influences bad implementation	Latency is higher with small data compared to data warehouse
Good understanding of the business problem	New technology, no proof, and track records Implementing security

lakehouse lends itself easily to the idea of having distributed structures and data products. This is supported because all datasets are directly accessible from the storage object, specifically a data lake, without connecting users to the same compute resources (Armbrust et al. 2021). Hence, sharing data is very straightforward and does not depend on which teams produce or consume it. In essence, the data mesh could consist of multiple data lakehouses that are connected. Therefore, the assumption is that the data mesh will be the next storage architecture evolution. Due to the scope of this research, this topic was not further examined however this is very interesting for future research.

6.4 Threats to validity

This research was completed with the utmost integrity, and we minimized the threats to the validity within our power. Nonetheless, we address possible threats to validity in the following paragraphs.

We acknowledge that the sample for this study might be biased because they all have very similar profiles and backgrounds and work for the same company. However, the bias has been mitigated in the following ways: 1) by using a structured set of interview questions, 2) by systematically analysing the results with a narrative approach and structuring the results in tables, 3) by validation checks by the researchers during the interviews, and 4) by recording the interviews to ensure the interview results were not based on the recollection of the researcher.

A second threat is the validity of the designed Data Storage Evolution Model. We tried to mitigate this threat by avoiding vendor-specific characteristics, extracting information from sources that are perceived as trustworthy, and performing validation interviews with experts whose experience in this field ranges from 6 to 16 years. However, this threat is not eliminated completely as the sample was relatively small and not very diverse. Moreover, only limited resources are published with regard to the data lakehouse. Hence, despite our best efforts to minimise it, there is still a threat to the validity of the Data Storage Evolution Model.

6.5 Contribution

Currently, the research into the data lakehouse is very limited in the literature. We found quite a few research articles and books that were done on the data lakehouse architecture. These studies referenced each other as well as other articles and blogs that were posted on technology forums. Moreover, we did not find an in-depth comparative study that examines the strengths and weaknesses of data warehouse, data lake, and data lakehouse architectures to provide a comprehensive picture of the evolution of the data storage architectures and the rationale behind the evolution process.

Thus, this study could contribute to fill this research gap in several ways. Firstly, in this study, we've done an in-depth comparison of the three existing data storage architectures. The findings are presented in a conceptual model that visualizes the evolution process and how the weaknesses of a preceding storage solution are solved in the next generation. This model can be used to explain the rationale behind the evolution process, and it can also be used for future research to explain the next evolutionary step to eliminate or address weaknesses with respect to handling modern data flows and advance data analysis requirements. Hence we argue that this study contributes data storage architecture related research by providing a conceptual foundation to foster further research and discussion on the data lakehouse architecture that would ultimately help in defining a generic data lakehouse reference architecture. Secondly, the results of this study can help practitioners to make use of the evolution model that explains the strengths and weaknesses of each storage architecture to obtain foundational knowledge on different data storage architectures by clearly identifying their respective strengths and weaknesses. Additionally, this is typically convenient when they are to consult clients on new projects on choosing which architecture suits their purpose. Finally, based on all the findings, this research contributes to practice by presenting a set of recommendations.

7 Conclusion

The rapid growth of the data inspired the need of sustainable data storage architectures specifically in terms of security and scalability. Therefore, in this research, we deeply examined three storage architectures namely data warehouse, data lake, and data lakehouse to explore the secure value provided by the latter over the other architectures. Thus, this research was guided by the main research question, "How to model secure value of the data lakehouse architecture by studying its comparative strengths and weaknesses with respect to other data storage architectures". To answer this question and to find out what secure values are offered by the data lakehouse architecture, a conceptual model was presented that explains the storage architectures' evolution process and each storage architecture's strengths and weaknesses. We argue that the secure value of a data lakehouse solution is that it incorporates best practices from data warehouses and data lakes next

to the implementation of the metadata and data governance layer. In essence, it supports the storage of all data formats and incorporates best practices of typical database management features to support data management, data governance, and securing the data. Moreover, based on the expert's evaluation, the added value can be explained by the fact that data lakehouses are capable of replacing data lakes. Additionally, it might possibly replace data warehouses in the near future.

This research study has provided many insights and information on the evolution of storage architectures, particularly the potential of the newest architecture: the data lakehouse. Based on the findings, it can be concluded that implementing the data lakehouse in data management platforms is valuable. Given the novelty of this concept and the fact that it has not been implemented by many practitioners yet, we recommend all practitioners invest time and resources into this concept. This entails facilitating and providing reimbursement for workshops that employees can follow, and participate in summits, and other technology-related events. This should motivate the employees to keep invested in this topic and up to date with the latest developments to become experts on this newest storage architecture.

Additionally, we specifically recommend running pilots and studying different case studies to discover the potential and challenges of implementing a data lakehouse. Given the interest of the experts that were included in our sample and the enthusiasm and optimism that is shared in published articles and web blogs, it is of high importance to become more familiar with the implementation of the data lakehouse. Therefore, in addition to knowing what the concept entails, actual practical experience is crucial.

Thirdly, we recommend practitioners to develop a change management plan that will facilitate a smooth transition from existing data lakes to a data lakehouse. This research shows that a data lakehouse is capable of replacing a data lake, hence we believe it is of high value to have a complete advisory report on how the transition process works. This possibly enhances and strengthens their current client relations. For the data warehouse, it is slightly more complicated, and not all experts are convinced that a data lakehouse is capable of replacing the data warehouse. However, by staying up to date with future developments and updates we recommend reacting immediately if certain developments prove that data warehouses can also be replaced.

Finally, it is recommended to utilize this study as a foundation for understanding the differences between the three storage architectures and their value. By knowing how to put each of them into context and understanding the evolution process, practitioners will be better aware of the strengths of each architecture. This will enhance their consults for future clients when selecting appropriate storage architecture.

Appendix

Table 5 Assumptions of Data warehouse

Data warehouse		
Code	Strengths	Weaknesses
A1	enhance business intelligence	
A2	improve decision-making processes	
A3	provide more and better information	
A4	help improve business process	
A5	support the achievement of business objectives	
A6	support ACID transactions	
A7	metadata management mechanisms	
A8	robust and stable storage solutions	
A9	allow versioning and access to historical data	
A10	have security measures in place	
A11		lack flexibility
A12		incur high implementation costs
A13		incur high maintenance costs

Table 6 Assumptions of Data lake

Code	Strengths	Weaknesses
A14	support the storage of heterogeneous data	
A15	cost-effective storage solutions	
A16	easy to scale	
A17	facilitate easy access to the data	support advanced analytics and data
A18	science techniques	
A19		lack metadata management
A20		provide low-security assurance
A21		risk of turning into a data swamp
A22		poor quality and reliability of the data
A23		inconsistent performance levels

Table 7 Assumptions of Data Lakehouse

Code	Strengths	Weaknesses
A24	support the storage of heterogeneous data	
A25	cost-effective storage solution	
A26	support a wider variety of workloads	
A27	reduce the level of data redundancy	
A28	contain high-quality, reliable and consistent data	
A29	deliver business intelligence support the decision making process	
A30	support advanced analytics and data science techniques	
A31	support real-time data applications	
A32	metadata management mechanisms in place	
A33	provide high-security assurance	
A34	support ACID transactions, data versioning, and indexing	
A35	support optimization techniques like caching, auxiliary data, and data layout	
A36	support management and performance features	
A37		not very mature
A38		not that much research or use-cases as examples available
A39		require training for new skills
A40		latency of a data lakehouse depends on the underlying cloud object-store

Fig. 10 Overview of viewpoints of each respondent for every assumption

		Respondent 1	Respondent 2	Respondent 3	Respondent 4	Respondent5
DW +	A1	Depends	Accepted	Accepted	Adjusted	Accepted
	A2	Depends	Accepted	Accepted	Accepted	Accepted
	A3	Rejected	Rejected	Accepted	Rejected	Accepted
	A4	Depends	Accepted	Accepted	Accepted	Accepted
	A5	Depends	Accepted	Accepted	Accepted	Accepted
	A6	Accepted	Depends	Accepted	Depends	Accepted
	A7	Accepted	Accepted	Accepted	Depends	Accepted
	A8	Accepted	Accepted	Accepted	Accepted	Accepted
	A9	Accepted	Depends	Accepted	Depends	Accepted
	A10	Accepted	Accepted	Accepted	Accepted	Accepted
DW -	A11	Accepted	Adjusted	Adjusted	Accepted	Accepted
	A12	Accepted	Accepted	Accepted	Accepted	Accepted
	A13	Accepted	Accepted	Accepted	Accepted	Accepted
DL +	A14	Accepted	Accepted	Accepted	Accepted	Accepted
	A15	Depends	Depends	Accepted	Accepted	Accepted
	A16	Accepted	Accepted	Accepted	Accepted	Accepted
	A17	Accepted	Depends	Accepted	Rejected	Depends
	A18	Accepted	Accepted	Accepted	Accepted	Accepted
DL -	A19	Accepted	Accepted	Accepted	Accepted	Adjusted
	A20	Depends	Adjusted	Rejected	Adjusted	Adjusted
	A21	Accepted	Accepted	Accepted	Accepted	Accepted
	A22	Rejected	Depends	Rejected	Accepted	Accepted
	A23	Accepted	Depends	Adjusted	Accepted	Accepted
DLH +	A24	Accepted	Accepted	Accepted	Accepted	Accepted
	A25	Accepted	Accepted	Accepted	Accepted	Accepted
	A26	Accepted	Accepted	Accepted	Accepted	Accepted
	A27	Adjusted	Accepted	Accepted	Rejected	Accepted
	A28	Accepted	Accepted	Accepted	Accepted	Accepted
	A29	Depends	Accepted	Accepted	Accepted	Accepted
	A30	Accepted	Accepted	Accepted	Accepted	Accepted
	A31	Accepted	Accepted	Accepted	Accepted	Accepted
	A32	Accepted	Accepted	Accepted	Depends	Depends
	A33	Accepted	Depends	Accepted	Adjusted	Rejected
	A34	Accepted	Accepted	Accepted	Accepted	Accepted
	A35	Accepted	Accepted	Accepted	Depends	Accepted
	A36	Depends	Accepted	Accepted	Accepted	Accepted
DLH -	A37	Accepted	Adjusted	Adjusted	Adjusted	Accepted
	A38	Accepted	Accepted	Accepted	Accepted	Accepted
	A39	Rejected	Accepted	Rejected	Rejected	Accepted
	A40	Accepted	Accepted	Accepted	Accepted	Accepted

Data availability The authors declare that the data supporting the findings of this study are available within the paper, its supplementary information are available (Janssen 2022).

Declarations

Conflict of interest/Competing interests On behalf of all authors, the corresponding author states that there is no conflict of interest.

References

Adelman S (2021) Data Warehouse Costs. EW Solutions - DataManagementU. [Online]. Available: https://www.ewsolutions.com/data-warehouse-costs/

Al-Okaily A, Al-Okaily M, Teoh AP, Al-Debei MM (2023) An empirical study on data warehouse systems effectiveness: the case of Jordanian banks in the business intelligence era. EuroMed J Bus 18(4):489–510. https://doi.org/10.1108/EMJB-01-2022-0011

Armbrust M, Das T, Sun L, Yavuz B, Zhu S, Murthy M, Torres J, Hovell H, Ionescu A, Łuszczak A et al (2020) Delta lake: high-performance acid table storage over cloud object stores. Proc VLDB Endowment 13(12):3411–3424

Armbrust M, Ghodsi A, Xin R, Zaharia M (2021) Lakehouse: a new genera- tion of open platforms that unify data warehousing and advanced analytics. In: Proceedings of CIDR, vol 8

Barnard Y, Bradley MD, Hodgson F, Lloyd AD (2013) Learning to use new technologies by older adults: Perceived difficulties, experimentation behaviour and usability. Comput Hum Behav 29(4):1715–1724. https://doi.org/10.1016/j.chb.2013.02.006

Begoli E, Goethert I, Knight K (2021) A lakehouse architecture for the management and analysis of heterogeneous data for biomedical research and mega-biobanks. In: 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 2021, pp 4643–4651. https://doi.org/10.1109/BigData52589.2021.9671534

Charmaz K (2006) Constructing grounded theory: a practical guide through qualitative analysis. London: Sage Publications

Chen J, Chen S, Rundensteiner EA (2002) A transactional model for data warehouse maintenance. In: Spaccapietra S, March ST, Kambayashi Y (eds) Conceptual Modeling — ER 2002. Lecture Notes in Computer Science, vol 2503. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-45816-6_27

Chen CP, Zhang C-Y (2014) Data-intensive applications, challenges, techniques and technologies: a survey on big data. Inf Sci 275:314–347

Dehghani Z (2019) How to move beyond a monolithic data lake to a distributed data mesh. Available: https://martinfowler.com/articles/data-monolith-to-mesh.html

Dworkin SL (2012) Sample size policy for qualitative studies using in-depth interviews. Arch Sex Behav 41:1319–1320. https://doi.org/10.1007/s10508-012-0016-6

Errami SA, Hajji H, El Kadi KA, Badir H (2023) Spatial big data architecture: from data warehouses and data lakes to the lakehouse. J Parallel Distrib Comput 176:70–79

Fang H (2015) Managing data lakes in big data era: what's a data lake and why has it became popular in data management ecosystem. In: 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), Shenyang, China, pp 820–824. https://doi.org/10.1109/CYBER.2015.7288049

Gosain A, Arora A (2015) Security issues in data warehouse: a systematic review. Procedia Comput Sci 48:149–157

Harby A, Zulkernine F (2022) From data warehouse to lakehouse: a comparative review, In: 2022 IEEE International Conference on Big Data (Big Data), Osaka, Japan, pp 389–395. https://doi.org/10.1109/BigData55660.2022.10020719

Hassan I (2024) Storage structures in the era of big data: from data warehouse to lakehouse. J Theor Appl Inf Technol 102(6). Available: https://www.jatit.org/volumes/Vol102No6/16Vol102No6.pdf

Inmon B (2016) Data lake architecture: Designing the data lake and avoiding the garbage dump

Inmon B, Levins M, Srivastava R (2021) Building the data lakehouse

Jain P, Kraft P, Power C, Das T, Stoica I, Zaharia M (2023) Analyzing and comparing lakehouse storage systems. CIDR

Janssen NE (2022) The evolution of data storage architectures: examining the value of the data lakehouse. Master's Thesis, University of Twente

Jarke M, Lenzerini M, Vassiliou Y, Vassiliadis P (2002) Fundamentals of data warehouses

Khine PP, Wang ZS (2018) Data lake: a new ideology in big data era. In: ITM Web of Conferences 17(03025). https://doi.org/10.1051/itmconf/20181703025

Kutay J (2021a) Data warehouse vs. data lake vs. data lakehouse: an overview of three cloud data storage patterns. Available: https://www.striim.com/blog/data-warehouse-vs-data-lake-vs-data-lakehouse-an-overview/

Kutay J (2021b) Data mart vs data warehouse vs database vs data lake. Available: https://www.zuar.com/blog/data-mart-vs-data-warehouse-vs-database-vs-data-lake/#:~:text=Data%20Lake%20vs.%20Data%20Mart%20The%20key%20differences,structured%20essential%20data%20for%20a%20department%20or%20function

Laney D et al (2001) 3d data management: Controlling data volume, velocity and variety. META Group Res Note 6(70):1

Lavrentyeva Y, Sherstnev A (2022) Cutting through the confusion: data ware- housevs.datalakevs.datalakehouse. Available: https://itrexgroup.com/blog/data-warehouse-vs-data-lake-vs-data-lakehouse-differences-use-cases-tips/#

Lu J, Liu ZH, Xu P, Zhang C (2018) UDBMS: road to unification for multi-model data management. In: Woo C, Lu J, Li Z, Ling T, Li G, Lee M (eds) Advances in Conceptual Modeling. ER 2018. Lecture Notes in Computer Science(LNCS), vol 11158. Springer, Cham. https://doi.org/10.1007/978-3-030-01391-2_33

Madera C, Laurent A (2016) The next information architecture evolution: the data lake wave. Proceedings of the 8th International Conference on Management of Digital Ecosystems. pp 174–180

Mason M (2010) Sample size and saturation in PhD studies using qualitative interviews. For Qual Sozialforschung/For: Qual Soc Res 11(3). https://doi.org/10.17169/fqs-11.3.1428

Mehmood H, Gilman E, Cortes M, Kostakos P, Byrne A, Valta K, Tekes S, Riekki J (2019) Implementing big data lake for heterogeneous data sources. In: 2019 Ieee 35th International Conference on Data Engineering Workshops (icdew). IEEE, pp 37–44

Nargesian F, Zhu E, Miller RJ, Pu KQ, Arocena PC (2019) Data lake man- agement: challenges and opportunities. Proc VLDB Endowment 12(12):1986–1989

Oreščanin D, Hlupić T (2021) Data lakehouse - a Novel Step in Analytics Architecture. In: 2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, pp 1242–1246. https://doi.org/10.23919/MIPRO52101.2021.9597091

Peffers K, Tuunanen T, Rothenberger MA, Chatterjee S (2007) A design science research methodology for information systems research. J Manag Inf Syst 24(3):45–77

Ravat F, Zhao Y (2019a) Data lakes: trends and perspectives. In: Database and Expert Systems Applications: 30th International Conference, DEXA 2019, Linz, Austria, August 26–29, 2019,

Proceedings, Part I. Springer-Verlag, Berlin, Heidelberg, 304–313. https://doi.org/10.1007/978-3-030-27615-7_23
Ravat F, Zhao Y (2019b) Metadata management for data lakes. In: Welzer T et al. New Trends in Databases and Information Systems. ADBIS 2019. Communications in Computer and Information Science, vol 1064. Springer, Cham. https://doi.org/10.1007/978-3-030-30278-8_5
Roelofs E, Persoon L, Nijsten S, Wiessler W, Dekker A, Lambin P (2013) Ben- efits of a clinical data warehouse with data mining tools to collect data for a radiotherapy trial. Radiother Oncol 108(1):174–179
Rosenthal A, Sciore E (2000) View security as the basis for data warehouse security. In: DMDW. p 8
Sawadogo P, Darmont J (2021) On data lake architectures and metadata manage- ment. J Intell Inf Syst 56:97–120
Schneider J, Gröger C, Lutsch A, Schwarz H, Mitschang B (2024) The lakehouse: State of the art on concepts and technologies. SN Comput Sci 5(5):1–39
Schneider J, Gröger C, Lutsch A, Schwarz H, Mitschang B (2023) Assessing the lakehouse: Analysis, requirements and definition. In: ICEIS (1). pp 44–56
Shiyal B (2021) Modern data warehouses and data lakehouses. pp 21–48
Strengholt P (2022) Data mesh: topologies and domain granu- larity. Towards Data Science. Available: https://towardsdatascience.com/data-mesh-topologies-and-domain-granularity-65290a4ebb90
Vishnu B, Manjunath T, Hamsa C (2014) An effective data warehouse security framework. Int J Comput Appl 975:8887
Walker C, Alrehamy H (2015) Personal data lake with data gravity pull. In: 2015 IEEE Fifth International Conference on Big Data and Cloud Computing. IEEE, pp 160–167. https://doi.org/10.1109/BDCloud.2015.62
Watson HJ, Goodhue DL, Wixom BH (2002) The benefits of data warehousing: why some organizations realize exceptional payoffs. Information & Management 39(6):491–502
Wieringa RJ (2014) Design science methodology for information systems and software engineering