Vol 7 No 3 (2025) E-ISSN: 2682-860X

International Journal on Robotics, Automation and Sciences

A Narrative Review of Data Mesh Architecture Principles and Implementation Outcomes

Kai Liang Lew, Chean Khim Toa[*1], Cheng Hong Chew, Xi Yuan Wong and Suleiman Aliyu Babale[*2]

***Abstract* – Centralised data architectures often create operational bottlenecks that limit organisational agility. Data Mesh offers a distributed alternative through domain ownership and federated governance. This narrative review synthesises 52 sources published between 2001 and 2024, examining the evolution from traditional data architectures to Data Mesh implementations across financial services, healthcare, e-commerce, and technology sectors. The review traces the progression from centralised data warehouses through distributed computing frameworks to Data Mesh's emergence, identifying four foundational principles domain-oriented decentralisation, data as a product, self-serve infrastructure, and federated governance. Analysis of recent implementation studies reveals mixed outcomes. Successful adoptions demonstrate improved domain autonomy and reduced central bottlenecks. However, multiple case reports significant coordination complexity and extended implementation timelines, with transformations requiring substantial investments in platform engineering. Consistent challenges emerge, including skill gaps in domain teams transitioning to data ownership, policy conflicts in federated governance structures, infrastructure investments that exceed traditional architectures, and cultural resistance to distributed accountability. Implementation success correlates with existing DevOps maturity, sustained executive sponsorship, phased adoption approaches, and robust metadata management capabilities. The review identifies critical research gaps in standardised success metrics, quantitative failure analysis, privacy-preserving techniques for federated environments, and long-term sustainability assessment. Based on the analysed cases, Data Mesh appears most suitable for large enterprises with diverse data domains and established platform engineering capabilities. Smaller organisations may find centralised approaches more appropriate given the complexity and resource requirements of distributed architectures. This synthesis provides practitioners with evidence-based insights while highlighting priorities for future research.**

Keywords— Data Mesh, Decentralised Data Architecture, Data Governance, Enterprise Architecture, Literature Review, Data As A Product.

I. INTRODUCTION

Data has become a critical asset that helps organisations improve decision-making and gain a competitive advantage, with advanced analytical approaches being applied across various domains to

*1Corresponding Author 1 email: cheankhim.toa@xmu.edu.my, ORCID: 0000-0003-0879-4848
*2Corresponding Author 2 email: sababale.ele@buk.edu.ng, ORCID: 0000-0002-1913-5445
Kai Liang Lew is with Faculty of Engineering and Technology, Multimedia University, Melaka, Malaysia (e-mail: 1132703002@student.mmu.edu.my)
Chean Kim Toa is with School of Computer and Data Science, Xiamen University Malaysia, Jalan Sunsuria, Bandar Sunsuria, 43900 Sepang, Selangor. (e-mail: cheankhim.toa@xmu.edu.my).
Cheng Hong Chew is with School of Computer and Data Science, Xiamen University Malaysia, Jalan Sunsuria, Bandar Sunsuria, 43900 Sepang, Selangor. (e-mail: SWE2204279@xmu.edu.my).
Xi Yuan Wong is with School of Computer and Data Science, Xiamen University Malaysia, Jalan Sunsuria, Bandar Sunsuria, 43900 Sepang, Selangor. (e-mail: SWE2202093@xmu.edu.my).
Suleiman Aliyu Babale is with Department of Electrical Engineering, Bayero University Kano, Kano, Nigeria (email: sababale.ele@buk.edu.ng)

International Journal on Robotics, Automation and Sciences (2025) 7, 3:114-123
https://doi.org/10.33093/ijoras.2025.7.3.15
Manuscript received: 9 Jun 2025 | Revised: 24 Jul 2025 | Accepted: 30 Jul 2025 | Published: 30 Nov 2025

Published by MMU PRESS. URL: http://journals.mmupress.com/ijoras

extract meaningful insights from complex datasets. Millions of users generate data daily through e-commerce platforms and banking transactions. The variety, volume, and velocity of this data distinguish it from traditional datasets, leading to its classification as "Big Data.". Enterprises use "Big Data" analytics for customer behaviour analysis, demand forecasting, and operational optimisation. However, traditional Big Data architectures, such as data warehouses and data lakes, are struggling to keep pace as businesses scale. These traditional centralised systems often create governance challenges and lack the collaboration and speed required for effective decision-making.

Data Mesh, introduced by Zhamak Dehghani in 2019, aims to improve speed and scalability compared to traditional centralised architectures. Data Mesh shifts the ownership of data to domain-specific teams. Each team will possess ownership of its respective domain of data and will be responsible for acquiring, storing, processing, and serving the data within that domain. The decentralisation of data allows for more efficient use of data, as specialised and experienced personnel can concentrate their efforts more effectively on their specific data domain. While Data Mesh promises to address these challenges through decentralisation, there is a need for an analysis of its effectiveness, implementation challenges, and comparative advantages.

The first objective of this paper is to examine the evolution and core principles of Data Mesh architecture. This objective illustrates the evolution of the data warehouse to the emergence of the Data Fabric. This also highlighted the reason why centralised platforms struggled as data volumes and domain complexity grew. The second objective is to identify key challenges and limitations in the current adoption of Data Mesh. This objective identifies and classifies obstacles encountered by real-world practitioners, as documented in the literature from 2019 to 2025.

The main research question can be stated as follows.

- What are the documented benefits and implementation challenges of Data Mesh architecture in enterprise environments?

One of the contributions is to provide an overview of Data Mesh principles, documented implementations, and reported challenges across various industries. The second contribution is to identify research gaps and future directions for Data Mesh development, providing a roadmap for researchers to follow as they navigate their research.

The paper is organised as follows. The literature review section examines the evolution of enterprise data architecture from data warehouses to data fabrics before focusing on the four core principles of Data Mesh. It summarises major real-world implementations and highlights related technological advancements. The discussion and analysis section outlines the review process employed to extract insights on Data Mesh implementations and their associated challenges. Lastly, the Conclusion and Future Direction section highlights main contributions, outlines strategic implications for practitioners and researchers, and suggests concrete directions for future work.

II. Methodology

This paper presents a narrative review of Data Mesh architecture, examining its emergence, principles, and implementation experiences through analysis of relevant literature published between 2001 and 2024. The review synthesises 52 sources selected for their contribution to understanding the evolution of data architectures and the specific characteristics of Data Mesh implementations.

Sources were identified through iterative searches in Google Scholar and Semantic Scholar, beginning with the term 'Data Mesh' and expanding to related concepts as themes emerged from initial readings. The search process followed citation chains from key papers, particularly Dehghani's foundational work and recent implementation studies. The final selection of 52 sources represents literature published between 2001 and 2024, including earlier works on distributed systems and microservices that provided necessary context for understanding Data Mesh's emergence. Sources were selected based on their relevance to either the theoretical foundations of Data Mesh, such as distributed computing and microservices, or empirical insights into implementation experiences. Implementation outcomes were assessed based on authors' own evaluations and reported metrics, recognizing that definitions of success varied across different organizational contexts and study designs.

The literature selection followed a purposive approach, identifying papers that addressed key aspects of Data Mesh or its foundational concepts. Sources were organised into three thematic categories based on their primary focus. Foundational literature encompassing traditional data warehouses, microservices architectures, and distributed computing systems provided the necessary context for understanding the architectural evolution that led to Data Mesh. Conceptual papers addressing Data Mesh principles, architectural patterns, and enabling technologies formed the second category. The third category comprised studies documenting actual Data Mesh implementations and their outcomes across various organisational contexts.

The review process involved reading and synthesising these sources to identify common themes, implementation patterns, and reported challenges. Papers were examined for their contributions to understanding the benefits of Data Mesh, implementation obstacles, and practical experiences. The synthesis focused on identifying convergent and divergent findings across different sources, with particular attention to gaps between theoretical expectations and reported implementation outcomes.

III. Literature Review

A. Evolution and Core Principles of Data Mesh

In the past, organisations relied on centralised data warehouses to store and manage all their data. This approach worked well when companies were smaller and data volumes were manageable. The centralised model provided a single source of truth for reporting and analytics. However, as businesses grew larger and data became increasingly complex, these systems began to show severe limitations.

Centralised data management creates operational bottlenecks. This is because all data requests must be processed through a central team, which can significantly reduce the response time [1], [2]. Business teams must wait for central data engineers to build reports or implement changes. This delay hinders business agility, making it challenging for companies to respond promptly to market changes. Research shows that centralised data warehouses face serious scalability challenges when organisations grow [3], [4].

Conway's Law explains that centralised systems struggle in large organisations because the designs are copies of the communication structures of these organisations [5]. This means that if the organisation has a distributed structure but a centralised data system, a fundamental mismatch will be encountered between how people work and how data flows.

As organisations scale, the central data teams cannot keep up with all the different business domains that need data. Each domain has its specific requirements and timelines, but the centralised approach forces everyone to follow the same processes. Central teams often lack the in-depth domain knowledge required to develop effective data models [6]. Support requirements increase significantly as more teams need data access [7].

As centralised data architectures struggled with scalability, Data Fabric emerged as an intermediate solution attempting to address integration challenges through unified data management across distributed environments. Data Fabric creates a unified layer that sits above distributed data sources, providing a single access point for data consumers while the underlying data remains physically distributed [8], [9].

Liu et al. [8] developed a metadata-based Data Fabric system that aggregates data from different sources through strong business correlation, creating comprehensive knowledge maps. The approach demonstrates how Data Fabric systems can integrate heterogeneous data sources without requiring physical centralisation. These architectures still rely heavily on centralised metadata management and coordination mechanisms.

Studies on distributed Data Fabric architectures [9] show that while these systems can handle massive data scattered across different departments and systems, they introduce significant complexity in metadata synchronisation and governance. Data Fabric solutions offer improved data discoverability compared to traditional data lakes, but at the expense of increased operational overhead and potential metadata inconsistencies across domains.

Analysis of Data Fabric implementations reveals several fundamental limitations that prevent them from fully addressing organisational scaling challenges [10]. While Data Fabric architectures excel at data integration and virtualisation, they maintain centralised governance models that create bottlenecks similar to those found in traditional data warehouses. Data Fabric solutions struggle with domain-specific requirements and organisational autonomy, as they still require central teams to manage policies and standards across all connected systems [11].

The Data Fabric approach, whilst technically sophisticated, failed to address the fundamental organisational misalignment issues identified by Conway's Law. Organisations implementing Data Fabric solutions often found that the centralised governance model conflicted with their distributed business operations, leading to coordination challenges and reduced agility. This limitation became apparent as enterprises required more domain-specific control over their data assets, prompting the exploration of alternative approaches to address this need.

Data Mesh emerged by adapting successful patterns from software development, particularly microservices architecture, which had effectively addressed similar scaling challenges in application development. Microservices architecture can break down big, monolithic applications into smaller, independent services [12], [13]. Each service is owned by a small team that can make changes quickly without affecting other parts of the system. The microservices approach demonstrates that distributed ownership is often more effective than centralised control in many situations [14], [15].

Studies demonstrate that microservices offer benefits such as faster deployment cycles, improved scalability, and enhanced separation of concerns among services [16]. However, they also introduce new challenges around service coordination, data consistency, and system complexity. These lessons became crucial for the development of Data Mesh - the benefits of distributed ownership were clear, but so was the need for careful coordination mechanisms.

Distributed computing frameworks can handle large volumes of data more effectively than centralised approaches [17], [18]. These systems also introduce new coordination and consistency challenges. The key is to design systems that minimise coordination overhead whilst maintaining necessary standards [19].

Cloud computing research demonstrated that scalable infrastructure could address some traditional limitations. However, technology and changes in organisations were required to realise the benefits of distributed approaches [20]. This experience showed that both technical and organisational alignment are crucial for success.

Data Mesh has emerged as a new approach to data architecture that addresses the limitations of both centralised systems and intermediate solutions, such as Data Fabric, by distributing ownership while maintaining coordination through federated governance. This allows organisations to scale their data capabilities without creating bottlenecks. This approach necessitates substantial changes to how teams operate and the skills they need to develop.

Data Mesh distributes both data ownership and governance responsibilities to domain teams while providing federated coordination mechanisms [5].

B. Core Principles of Data Mesh

Based on Machado et al. [21], Data Mesh is a decentralised data-architecture paradigm that distributes data ownership to domain teams rather than maintaining centralised control. Machado et al. explain that this model builds on principles from domain-driven design and microservices, applying similar decentralisation concepts to data management.

The Data Mesh approach emerged from challenges observed in scaling traditional centralised data architectures. Data Mesh explicitly aligns data architecture with business domain boundaries [5]. This approach shares conceptual foundations with microservices patterns, where distributed ownership aims to reduce coordination overhead [22].

Data As A Product

The first core principle is to treat data as a product. This ensures that data quality management operates at the same level as customer-facing products. This principle promotes clear ownership, comprehensive documentation, and formal service-level agreements [23]. Research on distributed systems shows that clear ownership and interface definitions are crucial for system reliability. The primary benefit of this approach is that data quality steadily improves over time through dedicated ownership and management [24]. Coordination costs increase when multiple teams consume the same data. Data products must define explicit interfaces and versioning strategies to maintain compatibility across consuming systems [25]. Teams must also understand how to access and use the data effectively. Adopting this principle means teams need to develop new skills, which may initially lead to a reduction in productivity as they learn these capabilities.

Domain-Oriented Data Ownership

Domain-oriented data ownership is the second core principle. It shifts responsibility from a centralised engineering team to the business unit that generates and primarily uses the data. This change reduces bottlenecks since the domain team no longer depends on central data engineers for routine updates [26]. However, domain teams may encounter skill gaps because they often lack the specialised data-engineering expertise needed. Both domain and central teams can struggle to maintain consistent standards, given their different backgrounds [27]. To implement this principle successfully, domain teams must invest resources in developing data-engineering capabilities. This represents a significant organisational transformation, requiring structured training and adaptation.

Federated Governance

The third core principle is federated governance. The purpose of this is to balance organisational standards with automation [28]. The principle behind it is to implement automated policy enforcement across domains. This can easily maintain the organisation's regulatory requirements. It also enables teams to make more informed decisions. However, conflicts may arise if two different teams' policies clash. To avoid this, all teams' policies must be standardised. This principle requires developing automation tools to enforce policies. It also requires a continuous monitoring system to detect any violations in real time.

Self-Serve Data Infrastructure

The self-serve data infrastructure is the last core principle. This principle provides domain teams with platforms that abstract underlying complexity while reducing dependencies on central teams and shortening development cycles [29]. The main challenge is that the organisation may not fully understand the complexity of implementing the platform. The platform team has to be flexible in balancing standardisation with domain needs. It also requires a large amount of investment in this infrastructure. [30]. Implementation requires guidance and governance for platform usage. Complete documentation and training are necessary to enable teams to utilise the infrastructure effectively [12].

While the Data Mesh principle can improve scalability and enhance decision-making, it also requires teams to develop new skills to apply its principles effectively. The teams may experience reduced productivity during the transition [31]. Overall, the architecture can increase system complexity, which may not always be beneficial. Additionally, self-serve data infrastructure can be more expensive than centralised approaches [32]. Table 1 shows the four core principles for data mesh. Figure 1 shows the domain-oriented ownership, data products, self-serve platform, and federated governance.

TABLE 1. The Four Core Principles for Data Mesh

Principle	Potential Benefits	Implementation Challenges	Key Requirements
Domain-oriented data ownership	Reduced central bottlenecks	Skills gaps in domain teams	Data engineering training
Data as a Product	Improved data quality through ownership	Increased coordination costs	Product management practices
Self-serve data platform	Reduced infrastructure dependencies	High platform investment	Platform engineering expertise
Federated governance	Balance of autonomy and compliance	Policy conflict resolution	Monitoring systems

C. Technological Advancements Supporting Data Mesh

Data Mesh architectures rely on specific technologies to function effectively. They can help teams manage data more effectively. They also ensure that different systems can interoperate and that governance rules are standardised. Studies have identified the essential technologies required to operate these systems efficiently.

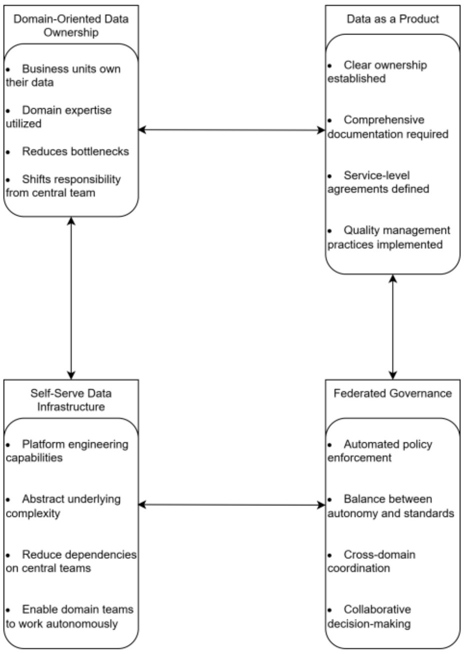

FIGURE 1. Data Mesh architecture showing domain-oriented ownership, data products, self-serve platform, and federated governance

Schema evolution and data interoperability present significant challenges in distributed data systems. Chillón et al. [33] developed an evolution method that works with both NoSQL and relational databases. The technique can automate data migration and keep data consistent across distributed domains. This demonstrates that the approach requires flexibility while maintaining data integrity, ensuring it does not violate constraints in a federated environment. Hu et al. [34] demonstrated a different evolution method using snapshot-based databases to do it online. This method can be done with only minimal performance overhead. It can identify changes across different autonomous domains without introducing additional complexity. This approach appears promising, but it requires effective coordination mechanisms to be in place.

The distributed governance and automation systems must ensure that policies remain consistent across different domains. It is essential to prevent conflicts between them. One major technology company uses a policy-as-code approach to automate enforcement in a distributed architecture [35]. However, this architecture becomes more complex to implement compared to a centralised system. Automated governance systems can enforce general data policies but still require human judgment for specific rules [36]. The security challenges inherent in distributed architectures, particularly those related to policy enforcement across autonomous domains, mirror those encountered in IoT and SDN environments, where distributed security management requires sophisticated coordination mechanisms [50]. Automation can significantly reduce manual work, but it still requires human supervision in complex settings.

Federated learning and distributed data processing technologies offer valuable insights for Data Mesh implementations. Federated learning systems [37] demonstrate that distributed data processing can maintain privacy and reduce data transfer overhead while introducing new challenges in model coordination and performance optimisation. Research on federated data management [38] indicates that management complexity increases substantially as the number of participating domains grows, potentially offsetting the benefits of decentralisation in smaller organisations.

Data quality management in distributed systems requires different approaches than centralised architectures. Recent studies [39] on AI-powered data governance have demonstrated that machine learning techniques can automate data quality assessment across distributed systems, although their effectiveness varies significantly based on the characteristics of the data domain. Automated data quality systems achieve 70-85% accuracy in detecting standard quality issues but struggle with domain-specific quality requirements [40]. Distributed data quality management requires substantially more sophisticated tooling than centralised approaches.

Metadata management and discoverability systems must handle the complexity of federated environments. Maintaining consistent metadata across autonomous domains requires automated synchronisation mechanisms, though these can introduce performance overhead [41]. Federated metadata systems can enhance data discoverability but at the expense of increased system complexity and potential metadata inconsistencies across domains [42].

The performance implications of distributed architectures have been extensively documented in multiple studies. Well-designed federated architectures can achieve comparable query performance to centralised systems for domain-specific queries, but experience significant performance degradation for cross-domain operations [43]. Performance analysis reveals that distributed architectures provide better fault isolation but complicate debugging and system monitoring compared to centralised approaches [44].

These technological enablers make Data Mesh implementation feasible, but research consistently demonstrates that distributed architectures introduce substantial operational complexity. Studies indicate that organisations must invest significantly in platform engineering and automation capabilities to realise the benefits of decentralised data management [45].

D. *Implementation Experiences and Lessons Learned*

Implementing the Data Mesh requires a thorough understanding of the foundational documentation. Understanding the documentation can increase the success rate of implementation and can help tackle challenges during the process. Recent studies have provided valuable insights into transitioning from a centralised system to a distributed data architecture.

Systematic Analysis of Implementation Challenges

Bode et al. [46] conducted 15 semi-structured interviews with industry experts across multiple organisations implementing Data Mesh architectures. Their research identified significant organisational challenges that contradict much of the promotional literature. The study found that organisations consistently struggled with the transition to federated data governance, with participants reporting difficulties in shifting responsibilities from centralised data teams to domain-specific units. A key finding was that organisations underestimated the cultural and organisational changes required, with many reverting to centralised practices when facing coordination difficulties.

Multi-Organisation Case Study Analysis

Kumar et al. [45] examined Data Mesh implementations across three organisations in the Netherlands and Germany, documenting both technical and organisational challenges that emerged during implementation and providing insights into practical design decisions and their outcomes.

Cross-Industry Implementation Patterns

Tamburri et al. [43] systematically reviewed 43 industrial implementations, revealing common patterns and failure modes while identifying several recurring implementation challenges across different industries and organisational contexts. Organisations consistently struggled with data product discovery and cataloguing, with many implementations failing to achieve the promised benefits due to poor discoverability of distributed data assets [50].

The study documented technology platform challenges that were consistently underestimated. Organisations require substantial investment in monitoring and observability tools specifically designed for distributed data architectures. The research found that debugging and troubleshooting distributed data systems proved significantly more complex than centralised approaches, leading to increased operational overhead. Many organisations reported higher-than-expected infrastructure costs due to the need for data replication and distributed storage.

Sector-Specific Adaptation Challenges

Research examining the adoption of Data Mesh in healthcare settings [47] revealed domain-specific implementation challenges. Regulatory compliance requirements significantly complicated the implementation of federated governance, with organisations struggling to maintain audit trails across distributed domains. Healthcare organisations reported particular difficulties in implementing data-sharing protocols while preserving patient privacy protections, requiring custom governance frameworks that increased implementation complexity.

Military and defence applications [48] demonstrated unique challenges in disconnected environments. While Data Mesh concepts can function in bandwidth-constrained tactical edge environments, they require substantial modifications to standard architectures. Maintaining data consistency across intermittently connected domains introduced technical complexities not present in traditional enterprise environments.

Implementation Success Factors and Limitations

Analysis across multiple studies [49] identified several critical success factors for Data Mesh implementations. Organisations with strong existing DevOps and platform engineering capabilities achieved more successful transitions than those attempting to build these capabilities during the adoption of Data Mesh.

Organisations with fewer than 100 data practitioners often found that the implementation costs of Data Mesh exceeded its benefits due to coordination overhead. The studies showed that smaller organisations frequently reverted to centralised approaches after encountering the complexities of distributed governance.

Critical Analysis of Implementation Outcomes

Common failure modes include inadequate platform investment, underestimation of organisational change requirements, and insufficient technical capabilities within domain teams. The success of Data Mesh depends heavily on the organisational context, existing technical capabilities, and sustained leadership commitment, rather than being universally applicable [36].

IV. Discussion and Analysis

A. Synthesis of Key Findings

Analysis of recent empirical research reveals that Data Mesh implementation outcomes vary significantly across organisational contexts, with success dependent on multiple interconnected factors rather than being universally applicable. While some organisations achieve benefits from distributed data ownership, implementation challenges are substantial and success rates are lower than those initially suggested in promotional literature.

A systematic analysis of industry implementations reveals a complex picture of both successes and failures [43], [45], [46]. Research by Bode et al. [46] demonstrates that organisations consistently underestimate the organisational transformation required, with their 15 expert interviews revealing that implementations take significantly longer than initially anticipated by most organisations.

Cross-sector analysis reveals common implementation patterns. Organisations frequently struggle with the transition to federated governance, requiring more than 18 months of sustained effort to achieve stable governance frameworks [43], [44]. Financial services organisations, while gaining some benefits in specific domains, report that the complexity of regulatory compliance increases substantially in distributed architectures. E-commerce implementations demonstrate promise in enabling domain autonomy, albeit at the expense of increased infrastructure complexity and coordination overhead.

Research synthesis identifies several factors that differentiate successful from unsuccessful implementations. Organisations with strong existing DevOps and platform engineering capabilities demonstrate higher success rates, while those attempting to build these capabilities during Data Mesh adoption face significant challenges.

Organisational readiness emerges as a critical determinant of success. Organisations with fewer than 100 data practitioners often find implementation costs exceed benefits due to coordination overhead. Domain teams consistently require 3-6 months to develop effective data product management capabilities, during which productivity may decrease compared to centralised approaches [47], [49].

Analysis of technological enablers reveals that organisations consistently underestimate the complexity of building self-service data platforms, contradicting simplified architectural presentations. The selection of appropriate platform technologies and vendors requires systematic multi-criteria decision-making approaches to evaluate competing alternatives effectively [51]. Distributed architectures typically introduce 15-30% higher infrastructure costs compared to centralised systems due to data replication and coordination requirements [44].

Federated governance implementation proves particularly challenging in practice. Organisations struggle to balance domain autonomy with organisational standards, often experiencing policy conflicts and inconsistent implementation across domains. Automated governance systems can enforce common policies, but they require sophisticated tooling to handle complex business rules. Studies reveal that debugging and monitoring distributed data systems significantly increases operational complexity compared to centralised approaches.

The research identifies organisational context as a primary determinant of Data Mesh viability. Large organisations with diverse data domains and substantial technical resources demonstrate higher success rates than smaller organisations with limited platform engineering capabilities. The industry sector influences implementation complexity, with heavily regulated industries facing additional challenges in maintaining compliance across multiple distributed domains.

Geographic and organisational culture factors also influence outcomes. Organisations with strong software engineering cultures tend to adapt more successfully than those with traditional data warehouse backgrounds. Executive sponsorship and dedicated change management resources are essential for overcoming organisational resistance to distributed data ownership, although these investments are often underestimated during planning phases.

The synthesis of empirical evidence indicates that Data Mesh effectiveness is highly contextual rather than universally applicable. While the four foundational principles provide a practical architectural framework, their implementation requires substantial organisational capability development and sustained investment. The evidence suggests that Data Mesh is most beneficial for large, technically sophisticated organisations with diverse data domains and sufficient resources to manage the complexity of distributed systems.

B. Critical Analysis of Implementation Patterns

Successful Data Mesh implementations often share several critical enablers. First, organisations that allocated dedicated platform engineering resources achieved smoother rollouts and fewer production incidents [43], [45]. Second, embedding automated quality checks and metadata generation into CI/CD pipelines has proven essential in maintaining trust as the mesh scales [44]. Third, federated governance bodies that meet regularly, comprising domain stewards, legal, and central data leadership, are used to avoid policy drift by continuously reconciling global standards with local requirements [36]. In some mid-market firms, the absence of a clear phased adoption roadmap led to domains reverting to centralised backups, undermining autonomy. These patterns also require disciplined operating practices.

C. Identification of Critical Research Gaps

Although the literature review indicates clear benefits, several research gaps impede broader adoption. First, no standardised ROI framework exists to quantify long-term returns on Data Mesh investments. Existing studies rely primarily on anecdotal evidence or single-organisation case metrics, making cross-case comparisons difficult. Second, maturity models for federated governance remain underdeveloped. They lack checklists that guide organisations from the pilot phase to complete decentralisation.

Third, privacy-preserving techniques within a Data Mesh context are largely unexplored. Practical methods for embedding differential privacy or federated learning into domain pipelines have not been well codified, leaving a gap in heavily regulated sectors. Fourth, change-management methodologies tailored to Data Mesh adoption lack specificity. Although general frameworks are referenced, no studies have validated which tactics most effectively drive the necessary cultural transformation. Finally, integration strategies for legacy systems need further elaboration, as only a few documented approaches successfully transform data warehouses into a federated mesh without extensive reengineering.

D. Implications for Practice and Theory

For practitioners, the findings underscore the importance of investing early in platform engineering capabilities and integrating automated governance checks to ensure effective management and control. Building reusable SDKs, templated pipeline scaffolds, and low-code interfaces can drastically reduce onboarding friction for domain teams. Governance councils should be chartered with clear decision-making rights, established communication protocols, and mechanisms for ongoing policy refinement. From a theoretical standpoint, Data Mesh challenges existing assumptions in data governance and organisational design. The shift toward domain-driven stewardship calls for new models in organisational behaviour that account for cross-functional accountability and product-centric data thinking. Information systems research can explore how federated governance bodies balance control and autonomy, as well as how knowledge transfer occurs between central and domain teams. Additionally, socio-technical theories of change management must be extended to account for the hybrid skill sets required in Data Mesh, which blends data engineering, product management, and domain expertise.

V. Conclusion and Future Directions

Data Mesh represents a decentralised data architecture approach that distributes ownership to domain teams and implements federated governance through self-serve platforms. Research indicates mixed implementation outcomes, as detailed in this review. Successful implementations require substantial investment in platform engineering and cultural transformation.

A. *Strategic Implications*

Organisations considering Data Mesh should conduct thorough readiness assessments of platform engineering capabilities and domain team skills. Given the documented implementation challenges, Data Mesh appears most suitable for large organisations with diverse data domains and strong technical resources. Organisations should expect extended platform development periods and higher costs than initially estimated. A phased pilot approach can validate readiness before implementing it more broadly.

B. *Future Research Directions*

There are several critical research needs to advance the practice of Data Mesh. First, longitudinal studies are necessary to track the implementation over two to three years. This allows a better understanding of their long-term viability. Second, it is essential to conduct systematic analyses of projects that did not succeed to pinpoint the failure modes and learn from mistakes. Third, contextual frameworks should be developed to help organisations determine whether Data Mesh is suitable for their context. Fourth, standardised models for measuring return on investment should be designed so companies can reliably compare costs and benefits. Finally, the finance and healthcare industries should utilise privacy-preserving techniques that comply with stringent regulations while upholding the principles of Data Mesh.

C. *Limitations and Considerations*

This review relies primarily on early-stage implementations and documented case studies, which may overrepresent successful outcomes while underreporting failures. Geographic bias toward North American and European implementations limits the generalizability of the findings. Most studies examine short-term outcomes within 6 to 18 months, which are insufficient to assess long-term sustainability. The rapid evolution of supporting technologies and the lack of standardised success metrics further limit current understanding.

Acknowledgment

We want to thank the peer reviewers for their assistance in reviewing the paper.

Funding Statement

There is no funding agencies supporting the research work

Author Contributions

Kai Liang Lew: Validation, Writing – Review & Editing;

Chean Khim Toa: Project Administration, Supervision, Writing – Review & Editing;

Chew Cheng Hong: Writing – Original Draft Preparation, Review & Editing;

Wong Xi Yuan: Writing – Original Draft Preparation, Review & Editing;

Suleiman Aliyu Babale: Writing – Review & Editing.

Conflict of Interests

No conflict of interests were disclosed.

Ethics Statements

Ethical approval was not applicable to this research since it did not involve human participants, animals, or sensitive data.

References

[1] N.E. Moukhi, I.E. Azami and A. Mouloudi, "Data warehouse state of the art and future challenges," *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, pp. 1–6, 2015. DOI: https://doi.org/10.1109/CloudTech.2015.7337004

[2] M. Rifaie, K. Kianmehr, R. Alhajj and M. J. Ridley, "Data warehouse architecture and design," *2008 IEEE International Conference on Information Reuse and Integration*, pp. 58–63, 2008. DOI: https://doi.org/10.1109/IRI.2008.4583005

[3] H.-F. Qin, Z.-M. Qian and Y.-C. Zhao, "On the Research of Data Warehouse in Big Data," *2015 International Conference on Network and Information Systems for Computers*, pp. 354–357, 2015. DOI: https://doi.org/10.1109/ICNISC.2015.126

[4] G. Furlow, "The case for building a data warehouse," *IT Professional*, vol. 3, no. 4, pp. 31–34, 2001. DOI: https://doi.org/10.1109/6294.946616

[5] M.E. Conway, "How Do Committees Invent?," *Datamation*, vol. 14, no. 4, pp. 28–31, 1968. URL: https://www.melconway.com/Home/pdf/committees.pdf

[6] B.K. Seah and N.E. Selan, "Design and implementation of data warehouse with data model using survey-based services data," *Fourth edition of the International Conference on the Innovative Computing Technology (INTECH 2014)*, pp. 58–64, 2014. DOI: https://doi.org/10.1109/INTECH.2014.6927748

[7] R.J. Santos, J. Bernardino and M. Vieira, "A survey on data security in data warehousing: Issues, challenges and opportunities," *2011 IEEE EUROCON - International Conference on Computer as a Tool*, pp. 1–4, 2011. DOI: https://doi.org/10.1109/EUROCON.2011.5929314

[8] K. Liu, M. Yang, X. Li, K. Zhang, X. Xia and H. Yan, "M-Data-Fabric: A Data Fabric System Based on Metadata," *2022 IEEE 5th International Conference on Big Data and Artificial Intelligence (BDAI)*, pp. 57–62, 2022. DOI: https://doi.org/10.1109/BDAI56143.2022.9862807

[9] X. Li, M. Yang, X. Xia, K. Zhang and K. Liu, "A Distributed Data Fabric Architecture based on Metadate Knowledge Graph," *2022 5th International Conference on Data Science and Information Technology (DSIT)*, pp. 1–7, 2022. DOI: https://doi.org/10.1109/DSIT55514.2022.9943831

[10] T. Priebe, S. Neumaier and S. Markus, "Finding Your Way Through the Jungle of Big Data Architectures," *2021 IEEE International Conference on Big Data (Big Data)*, pp. 5994–5996, 2021. DOI: https://doi.org/10.1109/BigData52589.2021.9671862

[11] A. Macías, D. Muñoz, E. Navarro and P. González, "Data fabric and digital twins: An integrated approach for data fusion design and evaluation of pervasive systems," *Information Fusion*, vol. 103, p. 102139, 2024. DOI: https://doi.org/10.1016/j.inffus.2023.102139

[12] N. Dragoni, M. Caramia, A. Meluzzi, F. Mosca, R. Rapini, and A. Spalazzi, "Microservices: yesterday, today, and tomorrow," *Present and Ulterior Software Engineering*, 2017. DOI: https://doi.org/10.1007/978-3-319-67425-4_12

[13] K. Bakshi, "Microservices-based software architecture and approaches," *2017 IEEE Aerospace Conference*, pp. 1–8, 2017. DOI: https://doi.org/10.1109/AERO.2017.7943959

[14] G. Liu, B. Huang, Z. Liang, M. Qin, H. Zhou and Z. Li, "Microservices: architecture, container, and challenges," *2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C)*, pp. 629–635, 2020. DOI: https://doi.org/10.1109/QRS-C51114.2020.00107

[15] P.D. Francesco, "Architecting Microservices," *2017 IEEE International Conference on Software Architecture Workshops (ICSAW)*, pp. 224–229, 2017. DOI: https://doi.org/10.1109/ICSAW.2017.65

[16] R.M. Munaf, J. Ahmed, F. Khakwani and T. Rana, "Microservices Architecture: Challenges and Proposed Conceptual Design," *2019 International Conference on Communication Technologies (ComTech)*, pp. 82–87, 2019. DOI: https://doi.org/10.1109/COMTECH.2019.8737831

[17] D. Talia, "A view of programming scalable data analysis: from clouds to exascale," *Journal of Cloud Computing*, vol. 8, no. 1, p. 4, 2019. DOI: https://doi.org/10.1186/s13677-019-0127-x

[18] C.S. Lee, Y. Kim, Y. Kim, J. Kim, S. Lee and B.R. Lee, "A Case Study of Data Management Challenges Presented in Large-Scale Machine Learning Workflows," *2023 IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*, pp. 71–81, 2023. DOI: https://doi.org/10.1109/CCGrid57682.2023.00017

[19] D. Wang, Q. Li, C. Xu, P. Wang and Z. Wang, "Research of Data Warehouse for Science and Technology Management System," *2021 International Conference on Service Science (ICSS)*, pp. 65–69, 2021. DOI: https://doi.org/10.1109/ICSS53362.2021.00018

[20] G. Garani, A. Chernov, I. Savvas and M. Butakova, "A Data Warehouse Approach for Business Intelligence," *2019 IEEE 28th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, pp. 70–75, 2019. DOI: https://doi.org/10.1109/WETICE.2019.00022

[21] I.A. Machado, C. Costa and M.Y. Santos, "Data Mesh: Concepts and Principles of a Paradigm Shift in Data Architectures," *Procedia Computer Science*, vol. 196, pp. 263–271, 2022. DOI: https://doi.org/10.1016/j.procs.2021.12.013

[22] S. Lungu and M. Nyirenda, "Current Trends in the Management of Distributed Transactions in Micro-Services Architectures: A Systematic Literature Review," *Open Journal of Applied Sciences*, vol. 14, no. 09, pp. 2519–2543, 2024. DOI: https://doi.org/10.4236/ojapps.2024.149167

[23] N. Polyzotis, S. Roy, S.E. Whang and M. Zinkevich, "Data Management Challenges in Production Machine Learning," *Proceedings of the 2017 ACM International Conference on Management of Data*, pp. 1723–1726, 2017. DOI: https://doi.org/10.1145/3035918.3054782

[24] R.Y. Wang and D.M. Strong, "Beyond Accuracy: What Data Quality Means to Data Consumers," *Journal of Management Information Systems*, vol. 12, no. 4, pp. 5–33, 1996. DOI: https://doi.org/10.1080/07421222.1996.11518099

[25] F. Li, B.C. Ooi, M.T. Özsu and S. Wu, "Distributed data management using MapReduce," *ACM Computing Surveys*, vol. 46, no. 3, 2014. DOI: https://doi.org/10.1145/2503009

[26] D. Taibi, V. Lenarduzzi and C. Pahl, "Processes, Motivations, and Issues for Migrating to Microservices Architectures: An Empirical Investigation," *IEEE Cloud Computing*, vol. 4, no. 5, pp. 22–32, 2017. DOI: https://doi.org/10.1109/MCC.2017.4250931

[27] L. Chen and M.A. Babar, "A systematic review of evaluation of variability management approaches in software product lines," *Information and Software Technology*, vol. 53, no. 4, pp. 344–362, 2011. DOI: https://doi.org/10.1016/j.infsof.2010.12.006

[28] M. Janssen and J.V.D. Hoven, "Big and Open Linked Data (BOLD) in government: A challenge to transparency and privacy?," *Government Information Quarterly*, vol. 32, no. 4, pp. 363–368, 2015. DOI: https://doi.org/10.1016/j.giq.2015.11.007

[29] J.M. Hellerstein, V. Sreekanti, J.E. Gonzalez, J. Dalton, A. Dey, S. Nag, K. Ramachandran, S. Arora, A. Bhattacharyya, S. Das, M. Donsky, G. Fierro, C. She, C. Steinbach, V. Subramanian and E. Sun, "Ground: A Data Context Service," *8th Biennial Conference on Innovative Data Systems Research*, 2017. URL:http://cidrdb.org/cidr2017/papers/p111-hellerstein-cidr17.pdf

[30] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, and M. Young, "Hidden Technical Debt in Machine Learning Systems," *Advances in Neural Information Processing Systems 28*, pp. 2503–2511, 2015. URL:https://proceedings.neurips.cc/paper_files/paper/2015/file/86df7dcfd896fcaf2674f757a2463eba-Paper.pdf

[31] T. Dybå and T. Dingsøyr, "Empirical studies of agile software development: A systematic review," *Information and Software Technology*, vol. 50, no. 9–10, pp. 833–859, 2008. DOI: https://doi.org/10.1016/j.infsof.2008.01.006

[32] T. Kraska, A. Talwalkar, J. Duchi, R. Griffith, M.J. Franklin, and M. Jordan, "MLbase: A Distributed Machine-learning System," *Proceedings of the 6th Biennial Conference on Innovative Data Systems Research*, pp. 1–7, 2013. URL:https://www.cidrdb.org/cidr2013/Papers/CIDR13_Paper118.pdf

[33] A.H. Chillón, M. Klettke, D.S. Ruiz and J.G. Molina, "A Generic Schema Evolution Approach for NoSQL and Relational Databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 36, no. 7, pp. 2774–2789, 2024. DOI: https://doi.org/10.1109/TKDE.2024.3362273

[34] T. Hu, T. Wang and Q. Zhou, "Online schema evolution is (almost) free for snapshot databases," *Proceedings of the VLDB Endowment*, vol. 16, no. 2, pp. 140–153, 2022. DOI: https://doi.org/10.14778/3565816.3565818

[35] A. Wider, S. Verma and A. Akhtar, "Decentralised Data Governance as Part of a Data Mesh Platform: Concepts and Approaches," *2023 IEEE International Conference on Web Services (ICWS)*, pp. 746–754, 2023. DOI: https://doi.org/10.1109/ICWS60048.2023.00101

[36] A. Goedegebuure, J. Hillebrand, J. Garrevoet and M. Van Keulen, "Data Mesh: A Systematic Gray Literature Review," *ACM Computing Surveys*, vol. 57, no. 1, 2024. DOI: https://doi.org/10.1145/3687301

[37] S.F. Lameh, W. Noble, Y. Amannejad and A. Afshar, "Analysis of Federated Learning as a Distributed Solution for Learning on Edge Devices," *2020 International Conference on Intelligent Data Science Technologies and Applications (IDSTA)*, pp. 66–74, 2020. DOI: https://doi.org/10.1109/IDSTA50958.2020.9264060

[38] H. Jeung, A.K. Mokashi, H.V. Jagadish and J.M. Hellerstein, "Effective Metadata Management in Federated Sensor Networks," *2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, pp. 107–114, 2010. DOI: https://doi.org/10.1109/SUTC.2010.29

[39] V. Yandrapalli, "AI-Powered Data Governance: A Cutting-Edge Method for Ensuring Data Quality for Machine Learning Applications," *2024 Second International Conference on Emerging Trends in Information Technology and Engineering (ICETITE)*, pp. 1–6, 2024. DOI: https://doi.org/10.1109/ic-ETITE58242.2024.10493601

[40] Y.A. Bena, A. Benatia, Y. Ghoumari, S. Oughdir and M.E. Koutbi, "Big Data Governance Challenges Arising From Data Generated by Intelligent Systems Technologies: A Systematic Literature Review," *IEEE Access*, vol. 13, pp. 12859–12888, 2025. DOI: https://doi.org/10.1109/ACCESS.2025.3528941

[41] S. Shrivastava, R. Singh, S. Mittal, N. Agrawal and S. Chaudhuri, "DQA: Scalable, Automated and Interactive Data Quality Advisor," *2019 IEEE International Conference on Big Data (Big Data)*, pp. 2913–2922, 2019. DOI: https://doi.org/10.1109/BigData47090.2019.9006187

[42] H. Hong, S. Yoo, Y. Jin, and Y. Jang, "How Can We Improve Data Quality for Machine Learning? A Visual Analytics System using Data and Process-driven Strategies," *2023 IEEE 16th Pacific Visualization Symposium (PacificVis)*, pp. 112–121, 2023.
DOI: https://doi.org/10.1109/PacificVis56936.2023.00020

[43] T.V. Eijk, I. Kumara, D.D. Nucci, D.A. Tamburri and W.-J.V.D. Heuvel, "Architectural Design Decisions for Self-Serve Data Platforms in Data Meshes," *2024 IEEE 21st International Conference on Software Architecture (ICSA)*, pp. 135-145, 2024.
DOI:https://doi.org/10.1109/ICSA59870.2024.00021

[44] A. Wider, K. Jarmul and A. Akhtar, "Towards Automating Federated Data Governance," *2024 IEEE International Conference on Web Services (ICWS)*, pp. 10–19, 2024.
DOI: https://doi.org/10.1109/ICWS62655.2024.00019

[45] I. Kumara, S. Driessen, T.V. Eijk, D.D. Nucci, D.A. Tamburri and W.-J.V.D. Heuvel, "Data Mesh Architecture: From Theory to Practice," *2024 IEEE 21st International Conference on Software Architecture Companion (ICSA-C)*, pp. 375–376, 2024.
DOI: https://doi.org/10.1109/ICSA-C63560.2024.00068

[46] J. Bode, N. Kühl, D. Kreuzberger, S. Hirschl and C. Holtmann, "Towards Avoiding the Data Mess: Industry Insights from Data Mesh Implementations," *IEEE Access*, vol. 12, pp. 95402-95416, 2024.
DOI: https://doi.org/10.1109/ACCESS.2024.3417291

[47] M. Falconi and P. Plebani, "Adopting Data Mesh principles to Boost Data Sharing for Clinical Trials," *2023 IEEE International Conference on Digital Health (ICDH)*, pp. 298–306, 2023.
DOI: https://doi.org/10.1109/ICDH60066.2023.00051

[48] S. Dahdal, F. Poltronieri, M. Tortonesi, C. Stefanelli and N. Suri, "A Data Mesh Approach for Enabling Data-Centric Applications at the Tactical Edge," *2023 International Conference on Military Communications and Information Systems (ICMCIS)*, pp. 1–9, 2023.
DOI: https://doi.org/10.1109/ICMCIS59922.2023.10253568

[49] A. Ashraf, A. Hassan and H. Mahdi, "Key Lessons from Microservices for Data Mesh Adoption," *2023 International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC)*, pp. 1–8, 2023.
DOI: https://doi.org/10.1109/MIUCC58832.2023.10278300

[50] S.A. Khan, I.H. Syed and J.I. Jawaid Iqbal, "From Signatures to AI: A Comprehensive Review of DDoS Detection Strategies in IoT & SDN," *International Journal on Robotics, Automation and Sciences*, vol. 7, no. 1, pp. 19–26, 2025.
DOI: https://doi.org/10.33093/ijoras.2025.7.1.3

[51] B.A. Rachman, A.L. Maukar and J.K. Runtuk, "Vendor Evaluation and Selection for Forwarding Activities Using Stepwise Weight Assessment Analysis-Combined Compromise Solution (SWARA-CoCoSo) Method," *International Journal on Robotics, Automation and Sciences*, vol. 7, no. 1, pp. 27–34, 2025.
DOI: https://doi.org/10.33093/ijoras.2025.7.1.4