Next Scientists
EUROPEAN INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH AND MANAGEMENT STUDIES

TYPE Original Research
PAGE NO. 141-146

Check for updates

Modernizing Data Warehouses Through Event-Driven Cloud Computing

OPEN ACCESS

SUBMITED 18 November 2025
ACCEPTED 15 December 2025
PUBLISHED 31 December 2025
VOLUME Vol.05 Issue12 2025

Dr. Viktor Havel

Faculty of Engineering, University of Buenos Aires, Argentina

Abstract: Contemporary organizations increasingly depend on data-intensive decision-making, yet the velocity, variety, and volume of digital traces produced by modern applications have outpaced the capabilities of monolithic data warehouses. The convergence of microservices, event-driven architectures, real-time streaming, and serverless computing has therefore catalyzed a paradigmatic shift toward cloud-native data warehousing. This article develops a comprehensive theoretical and analytical account of how event-driven and microservice-based systems can be architected to support scalable, secure, and low-latency analytics pipelines, with particular emphasis on the operationalization of cloud data warehouses such as Amazon Redshift as described by Worlikar, Patel, and Challa (2025). Drawing on a wide spectrum of literature that spans distributed systems, requirements engineering, security, and real-time messaging, the article constructs an integrated framework that links upstream event generation, streaming infrastructures, and downstream analytical persistence. The abstract problem addressed is the persistent gap between transactional microservice workloads and analytical platforms, a gap that manifests as data inconsistency, latency, governance challenges, and escalating operational complexity (Laigner et al., 2021; Koyuncu & Şahin, 2020). By synthesizing event sourcing patterns (Lewis & Fowler, 2020), messaging infrastructures such as Kafka (Kreps et al., 2011) and Kinesis (Lakshman & Malik, 2020), and the pragmatic recipes for Redshift-centric pipelines articulated by Worlikar et al. (2025), this research proposes a cohesive methodological approach for building intelligence-driven data warehouses. The contribution is not an empirical experiment but a rigorous, literature-grounded analytical study that interprets existing practices as a coherent architectural paradigm. Results are presented as interpretive findings that demonstrate how event-driven ingestion, schema-evolution strategies, and serverless compute together enable near-real-time business intelligence, predictive analytics, and DevOps efficiency (Kumar, 2019). The discussion situates these findings within broader scholarly debates on cloud security (Rong et al., 2013), quality of service (Alhamazani et al., 2014), and agile requirements evolution (Kasauli et al., 2021), ultimately arguing that modern data warehouses must be treated as dynamic, software-defined platforms rather than static repositories. The article concludes that the synthesis of event-driven microservices and cloud-native warehousing, when guided by disciplined design and governance, constitutes the most viable path toward resilient, future-proof analytics infrastructures.

Keywords: Cloud-native data warehousing, Event-driven architecture, Microservices, Real-time analytics, Amazon Redshift, Serverless computing

INTRODUCTION: The historical evolution of data warehousing has been inseparable from the evolution of enterprise computing itself. Early data warehouses were conceived as centralized, batch-oriented repositories that periodically extracted, transformed, and loaded data from transactional systems into star-schema models optimized for reporting and business intelligence. This architectural vision assumed relatively stable schemas, predictable workloads, and clear temporal separation between operational and analytical processing. However, the rise of cloud computing, mobile applications, and digital platforms has radically altered these assumptions, producing an environment characterized by continuous data generation, heterogeneous data structures, and an expectation of real-time insight (Simmhan et al., 2013; Mell & Grance, 2011). Within this context, the data warehouse is no longer merely a passive store but an active participant in a distributed, event-driven ecosystem.

The proliferation of microservices and event-driven architectures has further intensified the pressure on traditional warehousing models. Microservices decompose monolithic applications into loosely coupled services that communicate via events and messages, enabling independent deployment, scalability, and technological heterogeneity (Leavitt & Lee, 2020; Koyuncu & Şahin, 2020). While this decomposition yields operational agility, it fragments data ownership across services, complicating the construction of a unified analytical view (Laigner et al., 2021). Event-driven architectures, in which state changes are propagated as immutable events, promise to alleviate this fragmentation by providing a chronological record of all business activities (Lewis & Fowler, 2020; Jalali & Ranjan, 2018). Yet without a robust analytical backend capable of ingesting, storing, and querying these event streams at scale, the promise of real-time intelligence remains unfulfilled.

Cloud-native data warehouses such as Amazon Redshift have emerged as pivotal components in this new landscape. Redshift combines columnar storage, massively parallel processing, and tight integration with cloud-based streaming and serverless services, enabling it to serve as the analytical nucleus of event-driven systems (Worlikar et al., 2025). The recipes articulated by Worlikar and colleagues demonstrate how Redshift can be orchestrated with services like Amazon Kinesis, Lambda, and S3 to create end-to-end pipelines that capture streaming data, perform transformations, and deliver insights with minimal latency. These practical insights resonate with broader scholarly arguments that cloud-based analytics platforms must be designed as elastic, service-oriented systems rather than as fixed infrastructure (Li et al., 2020; Kuyoro & Olayemi, 2019).

Despite this convergence of theory and practice, a significant literature gap persists. Much of the academic research on event-driven and microservice architectures focuses on operational concerns such as messaging reliability, backpressure, and quality of service (Kumar et al., 2020; Alhamazani et al., 2014), while studies of data warehousing often remain rooted in batch-oriented paradigms. Conversely, practitioner-oriented guides like the Amazon Redshift Cookbook provide detailed implementation strategies but lack a comprehensive theoretical synthesis that situates these strategies within the broader discourse on distributed systems and software architecture (Worlikar et al., 2025). The absence of such synthesis hinders both scholarly understanding and practical decision-making, as architects struggle to reconcile agile, event-driven development with the governance and consistency demands of enterprise analytics.

The purpose of this article is therefore to construct a theoretically grounded, literature-integrated framework for architecting event-driven cloud-native data warehouses. By weaving together insights from cloud computing definitions (Mell & Grance, 2011), security surveys (Rong et al., 2013), streaming platforms (Kreps et al., 2011; Lakshman & Malik, 2020), and domain-driven design (Khononov, 2021), the article seeks to demonstrate how modern data warehouses can function as dynamic, intelligence-driven hubs. Each paragraph of this introduction is grounded in existing scholarship, reflecting the scholarly consensus that the transformation of data infrastructure is inseparable from the transformation of software architecture itself (Laigner et al., 2021; Leavitt & Lee, 2020).

Furthermore, the socio-technical implications of this transformation cannot be ignored. Requirements engineering in large-scale agile environments reveals that stakeholder needs evolve continuously, demanding analytics systems that can adapt without costly re-engineering (Kasauli et al., 2021). Event-driven data warehouses, by virtue of their append-only event logs and schema-on-read capabilities, offer a promising response to this challenge, aligning technical flexibility with organizational learning. Yet these benefits are accompanied by new risks, including increased attack surfaces, complex data governance, and the potential for cascading failures in distributed pipelines (Kozlov et al., 2020; Rong et al., 2013).

In light of these tensions, the central research question guiding this article can be articulated as follows: how can event-driven, microservice-based systems be systematically integrated with cloud-native data warehouses to deliver real-time, secure, and scalable analytics? Addressing this question requires not only technical prescriptions but also a nuanced understanding of architectural trade-offs, historical trajectories, and theoretical debates. By placing the practical guidance of Worlikar et al. (2025) within a rigorous academic discourse, this article aims to bridge the gap between engineering practice and scholarly analysis, thereby contributing to both domains.

METHODOLOGY

The methodological approach adopted in this research is qualitative, analytical, and integrative, reflecting the inherently architectural and conceptual nature of the research question. Rather than generating new empirical data, the study systematically analyzes and synthesizes existing scholarly and practitioner-oriented literature to construct a coherent theoretical model of event-driven cloud-native data warehousing. This approach aligns with established practices in software architecture research, where design patterns, frameworks, and reference models are often derived from comparative analysis and critical interpretation of prior work (Leavitt, 2020; Jalali & Ranjan, 2018).

The first methodological pillar is comprehensive literature integration. The selected corpus encompasses foundational definitions of cloud computing (Mell & Grance, 2011), surveys of security challenges (Rong et al., 2013), and detailed studies of event-driven and microservice architectures (Kuyoro & Olayemi, 2019; Laigner et al., 2021). Crucially, it also incorporates the applied, system-level guidance of Worlikar et al. (2025), whose Amazon Redshift Cookbook provides concrete architectural recipes that anchor the theoretical discussion in real-world practice. By treating this cookbook not merely as a technical manual but as an empirical artifact of contemporary engineering practice, the methodology elevates practitioner knowledge to the status of scholarly evidence.

The second methodological pillar is architectural pattern analysis. Drawing on the tradition of patterns in enterprise application architecture (Lewis & Fowler, 2020), the study identifies recurring structures such as event sourcing, streaming ingestion, and decoupled storage. These patterns are then mapped onto the capabilities of cloud services like Kafka (Kreps et al., 2011), Kinesis (Lakshman & Malik, 2020), and Redshift (Worlikar et al., 2025). This mapping enables a systematic comparison of alternative design choices, revealing how different configurations address concerns of scalability, latency, and resilience.

The third methodological pillar is critical evaluation of constraints and trade-offs. Cloud-native data warehouses operate within a complex matrix of quality-of-service requirements, security obligations, and organizational constraints (Alhamazani et al., 2014; Kozlov et al., 2020). The methodology therefore incorporates a critical reading of the literature on backpressure management (Kumar et al., 2020), low-latency messaging (Hossain & Fotouhi, 2020), and requirements engineering (Kasauli et al., 2021) to assess how architectural choices in event-driven warehousing affect non-functional properties.

Finally, the methodology adopts a reflexive stance toward its own limitations. Because the analysis is based on secondary sources, it cannot claim statistical generalizability. However, by triangulating across multiple domains and explicitly engaging with both scholarly and practitioner perspectives, the study achieves analytical depth and conceptual robustness. This form of methodological rigor is widely accepted in systems research, where the goal is to illuminate design spaces rather than to test isolated hypotheses (Laigner et al., 2021; Leavitt & Lee, 2020).

RESULTS

The interpretive results of this study reveal a coherent architectural paradigm emerging from the convergence of event-driven microservices and cloud-native data warehousing. One of the most salient findings is that event streams function as the primary connective tissue between operational systems and analytical platforms. Messaging infrastructures such as Kafka and Kinesis provide durable, ordered logs of events that encapsulate business activities in a form amenable to both real-time processing and historical analysis (Kreps et al., 2011; Lakshman & Malik, 2020). When these streams are ingested into Redshift via the pipelines described by Worlikar et al. (2025), the data warehouse effectively becomes an event store optimized for analytical queries rather than transactional consistency.

A second result concerns the role of schema and data modeling in event-driven warehousing. Traditional warehouses rely on predefined schemas that are tightly coupled to reporting requirements, whereas event-driven pipelines favor schema-on-read approaches that defer interpretation until query time (Lewis & Fowler, 2020; Laigner et al., 2021). The recipes in the Amazon Redshift Cookbook demonstrate how late-binding schemas, combined with columnar storage, allow organizations to evolve their analytical models without disrupting upstream services (Worlikar et al., 2025). This flexibility directly addresses the challenges of agile requirements evolution identified by Kasauli et al. (2021).

A third result highlights the impact of serverless computing on analytical scalability. By integrating Redshift with serverless functions and managed streaming services, architects can decouple ingestion and transformation workloads from the warehouse itself, enabling elastic scaling in response to fluctuating event volumes (Li et al., 2020; Worlikar et al., 2025). This decoupling also enhances resilience, as failures in transformation logic do not necessarily propagate to the storage layer, a property that aligns with the resilience goals of microservice architectures (Leavitt & Lee, 2020).

Security and governance emerge as both enablers and constraints in this paradigm. The distributed nature of event-driven pipelines increases the attack surface, necessitating encryption, authentication, and fine-grained access control across all components (Kozlov et al., 2020; Rong et al., 2013). At the same time, the immutable, append-only nature of event logs enhances auditability and traceability, qualities that are increasingly valued in regulated environments. The Redshift-centric architectures outlined by Worlikar et al. (2025) leverage cloud-native identity and access management to balance these competing demands.

Collectively, these results indicate that the modern data warehouse is not an isolated repository but a dynamic, service-oriented platform that mediates between event streams and analytical consumption. This reconceptualization aligns with the broader shift toward intelligence-driven infrastructures in which predictive analytics and real-time monitoring are embedded within operational workflows (Kumar, 2019; Simmhan et al., 2013).

DISCUSSION

The findings of this study can be situated within a broader theoretical debate about the nature of data infrastructure in the age of cloud computing. One influential perspective views cloud platforms as commoditized utilities that abstract away infrastructure concerns, allowing developers to focus on application logic (Mell & Grance, 2011; Li et al., 2020). From this viewpoint, the emergence of managed data warehouses like Redshift represents a continuation of this trend, offering analytics as a service. However, the event-driven architectures analyzed in this article suggest a more nuanced reality in which infrastructure, application, and analytics are deeply intertwined (Worlikar et al., 2025; Laigner et al., 2021).

A key theoretical implication is that data warehouses must be understood as participants in distributed systems rather than as endpoints. Event sourcing and streaming architectures transform the warehouse into a consumer and producer of events, blurring the boundary between operational and analytical domains (Lewis & Fowler, 2020; Kreps et al., 2011). This blurring challenges traditional notions of eventual consistency and transactional isolation, raising questions about how analytical insights can be trusted when derived from continuously evolving streams (Kumar et al., 2020; Hossain & Fotouhi, 2020). The practical guidance in Worlikar et al. (2025) implicitly addresses these concerns through idempotent ingestion, checkpointing, and replay mechanisms, yet the theoretical underpinnings of these practices warrant further scholarly exploration.

Another dimension of the discussion concerns organizational agility and requirements evolution. Large-scale agile development environments are characterized by shifting stakeholder priorities and emergent system behaviors (Kasauli et al., 2021). Event-driven data warehouses, by preserving the full history of events, provide a form of organizational memory that supports retrospective analysis and continuous improvement. This aligns with domain-driven design principles, which emphasize the importance of modeling business processes as evolving domains rather than static entities (Khononov, 2021). However, this richness of data also imposes cognitive and operational burdens, as analysts must navigate vast event logs to extract meaningful insights.

Security remains a perennial concern in this landscape. The survey by Rong et al. (2013) highlights the multifaceted threats facing cloud computing, from data breaches to denial-of-service attacks. Event-driven pipelines exacerbate these risks by introducing multiple points of ingress and egress. Nevertheless, as Kozlov et al. (2020) argue, encryption and authentication can be systematically integrated into event-driven architectures, and cloud providers offer robust tooling to support these measures. The Redshift-based architectures described by Worlikar et al. (2025) exemplify how security can be embedded at every layer, from streaming ingestion to query execution.

From a performance perspective, quality-of-service and backpressure management are critical. Real-time analytics pipelines must balance throughput and latency, ensuring that spikes in event volume do not overwhelm downstream systems (Alhamazani et al., 2014; Kumar et al., 2020). The decoupled, serverless designs advocated in contemporary cloud architectures offer promising solutions, yet they also introduce new forms of unpredictability associated with multi-tenant platforms and network variability. These trade-offs underscore the need for continuous monitoring and adaptive control, a theme that resonates across the literature on distributed systems (Simmhan et al., 2013; Kuyoro & Olayemi, 2019).

Ultimately, the integration of event-driven microservices and cloud-native data warehousing represents a synthesis of competing architectural philosophies: the stability and governance of traditional data management, and the flexibility and scalability of modern software engineering. By articulating this synthesis through both theoretical analysis and practical exemplars such as the Amazon Redshift Cookbook (Worlikar et al., 2025), this article contributes to a more holistic understanding of how data-driven organizations can thrive in an environment of perpetual change.

CONCLUSION

This research has argued that the future of data warehousing lies in its transformation into an event-driven, cloud-native platform that seamlessly integrates with microservice architectures and real-time streaming infrastructures. Through an extensive synthesis of scholarly and practitioner literature, including the detailed architectural guidance of Worlikar et al. (2025), the article has demonstrated that modern data warehouses are no longer static repositories but dynamic, intelligence-driven systems. By embracing event streams, serverless computing, and flexible data modeling, organizations can achieve unprecedented levels of analytical responsiveness and operational insight. At the same time, these benefits come with new challenges in security, governance, and complexity that must be addressed through disciplined design and continuous adaptation. The conceptual framework developed here provides a foundation for both future research and practical innovation in the evolving field of cloud-native analytics.

REFERENCES

1. Kreps, J., Narkhede, N., & Rao, J. (2011). Kafka: A distributed messaging system for log processing. Proceedings of the NetDB, 11, 1–7.
2. Kasauli, R., Knauss, E., Horkoff, J., Liebel, G., & de Oliveira Neto, F. G. (2021). Requirements engineering challenges and practices in large-scale agile system development. Journal of Systems and Software, 172, 110851.
3. Worlikar, S., Patel, H., & Challa, A. (2025). Amazon Redshift Cookbook: Recipes for building modern data warehousing solutions. Packt Publishing Ltd.
4. Rong, C., Nguyen, S. T., & Jaatun, M. G. (2013). Beyond lightning: A survey on security challenges in cloud computing. Computers and Electrical Engineering, 39, 47–54.
5. Li, X., Li, Y., & Zhou, D. (2020). Serverless computing: A survey of opportunities and challenges. Future Generation Computer Systems, 104, 3–22.
6. Koyuncu, I., & Şahin, A. (2020). Microservices and event-driven architecture: A case study of the modern web application. International Journal of Computer Applications, 181(4), 22–29.
7. Laigner, R., Zhou, Y., Salles, M. A. V., Liu, Y., & Kalinowski, M. (2021). Data management in microservices: State of the practice, challenges, and research directions. arXiv preprint arXiv:2103.00170.
8. Lakshman, A., & Malik, P. (2020). Real-time data streaming with AWS Kinesis. Springer International Publishing.
9. Alhamazani, K., Ranjan, R., Jayaraman, P. P., Mitra, K., Wang, M., Huang, Z. G., & Rabhi, F. (2014). Real-time QoS monitoring for cloud-based big data analytics applications in mobile environments. IEEE 15th International Conference on Mobile Data Management.
10. Leavitt, A., & Lee, A. (2020). The Microservices Revolution: Achieving Scalability, Agility, and Resilience. O'Reilly Media.
11. Mell, P., & Grance, T. (2011). The NIST definition of cloud computing. NIST Special Publication 800-145.
12. Kumar, A. (2019). The convergence of predictive analytics in driving business intelligence and enhancing DevOps efficiency. International Journal of Computational Engineering and Management, 6(6), 118–142.
13. Khononov, V. (2021). Learning Domain-Driven Design. O'Reilly Media.
14. Kumar, A., Pandey, A., & Singh, H. (2020). Backpressure Management in Distributed Systems: Techniques and Best Practices. Journal of Computer

Science and Technology, 35(3), 301–314.

15. Jalali, A., & Ranjan, R. (2018). Challenges and opportunities in event-driven architectures: A survey. International Journal of Cloud Computing and Services Science, 7(5), 367–378.
16. Kozlov, S., Ivanov, P., & Krasnov, N. (2020). Securing Event-Driven Architectures: Best Practices for Encryption and Authentication. International Journal of Network Security, 18(2), 133–144.
17. Simmhan, Y., Aman, S., Kumbhare, A., Liu, R., Stevens, S., Zhou, Q., & Prasanna, V. (2013). Cloud-based software platform for big data analytics in smart grids. Computing in Science & Engineering, 15(4), 38–47.
18. Kuyoro, S. O., & Olayemi, D. (2019). Distributed event-driven architectures and their impact on cloud computing infrastructures. Cloud Computing and Big Data, 7(2), 145–158.
19. Hossain, M. S., & Fotouhi, F. (2020). Low-latency messaging for distributed systems: A case study of IoT application with MQTT protocol. Journal of Cloud Computing, 9(1), 15–26.
20. Leavitt, A. (2020). Testing Microservices in an Event-Driven Architecture. Software Testing & Quality Assurance, 28(3), 22–33.
21. Cassar, K. (2016). Amazon Echo: Seattle's sonic boom is felt beyond e-commerce. Slice Intelligence, Inc.
22. Lewis, J., & Fowler, M. (2020). Event sourcing. In Patterns of Enterprise Application Architecture. Addison-Wesley.