INTERNATIONAL MULTI DISCIPLINARY JOURNAL FOR RESEARCH & DEVELOPMENT

eISSN 2394-6334 Impact factor: 7,854 Volume 12, issue 11 (2025)

CLOUD-NATIVE COLUMNAR DATA WAREHOUSES: COMPARATIVE ARCHITECTURAL ANALYSIS OF AMAZON REDSHIFT, AZURE SYNAPSE, AND GOOGLE BIGQUERY IN MODERN ANALYTICS ECOSYSTEMS

Prof. Rashid Mahmood

Department of Information Systems, University of Barcelona, Spain

Abstract: Cloud-native data warehousing has emerged as a foundational paradigm for contemporary analytics, driven by the explosive growth of digital data, the globalization of enterprise operations, and the increasing sophistication of business intelligence and machine learning workloads. The convergence of scalable cloud infrastructure with advanced column-oriented database architectures has enabled a new generation of analytical systems that promise elasticity, high performance, and operational simplicity. Yet, despite widespread adoption of platforms such as Amazon Redshift, Microsoft Azure Synapse Analytics, and Google BigQuery, there remains significant conceptual ambiguity and methodological inconsistency in how these systems are evaluated, designed, and governed within modern data ecosystems. This research addresses that gap by developing a theoretically grounded and empirically informed framework for understanding cloud-native data warehouses as socio-technical systems that integrate architectural principles, economic models, and organizational practices.

Drawing on a synthesis of foundational database theory, cloud computing literature, and platform-specific technical documentation, this study situates contemporary data warehouses within the historical evolution of column-oriented systems, massively parallel processing, and distributed query execution. The analysis is deeply informed by practical design knowledge articulated in Worlikar, Patel, and Challa’s Amazon Redshift Cookbook, which provides a detailed exposition of how modern data warehousing patterns are implemented in production-scale environments and how architectural decisions interact with workload characteristics, governance constraints, and cost optimization strategies (Worlikar et al., 2025). By embedding such practitioner-oriented insights within a broader theoretical discourse, the research bridges the persistent divide between academic models of database systems and the operational realities of cloud platforms.

Methodologically, the study employs a qualitative comparative architecture analysis that systematically examines Redshift, Synapse, and BigQuery across multiple dimensions including storage abstraction, query execution models, workload isolation, scalability mechanisms, and data governance. Rather than relying on benchmark metrics or synthetic performance tests, the analysis interprets architectural behaviors through the lens of design trade-offs articulated in both scholarly and industrial literature, recognizing that performance, reliability, and cost are co-produced by technology and organizational context. This approach enables a richer understanding of why ostensibly similar systems often yield divergent outcomes when deployed in real enterprises.

The results demonstrate that although all three platforms rely on columnar storage and distributed processing, they embody fundamentally different philosophies of control and abstraction. Redshift emphasizes explicit architectural tuning and cluster-based resource management, Synapse integrates tightly with broader enterprise data ecosystems through hybrid transactional-analytical processing models, and BigQuery advances a serverless, highly abstracted paradigm that redefines the relationship between users and infrastructure. These differences have profound implications for data governance, reproducibility, cost predictability, and the epistemology of analytics itself.

The discussion extends these findings into a theoretical critique of cloud data warehousing, arguing that contemporary platforms are not merely technological artifacts but institutional infrastructures that shape how organizations know, measure, and act upon their data. By articulating limitations, unresolved tensions, and future research trajectories, this article provides a comprehensive foundation for both scholars and practitioners seeking to understand and advance the state of cloud-native analytics.

Keywords: Cloud data warehousing; Column-oriented databases; Amazon Redshift; Azure Synapse Analytics; Google BigQuery; Distributed query processing; Analytics infrastructure

Introduction

The transformation of data into a central economic and strategic resource has been one of the defining features of the contemporary digital era, and nowhere is this more evident than in the rapid evolution of data warehousing technologies that now operate almost entirely within cloud computing environments (Borra, 2024). Historically, data warehouses were conceived as large, monolithic repositories that aggregated transactional data for reporting and decision support, typically running on dedicated on-premise hardware under the control of specialized database administrators. Over time, however, the increasing volume, velocity, and variety of data, combined with the globalization of business processes and the democratization of analytics, rendered such architectures increasingly inadequate. Cloud computing emerged as a response to these pressures, offering elastic compute, distributed storage, and a pay-as-you-go economic model that fundamentally reconfigured how data infrastructure could be provisioned and governed (Amazon Web Services, 2024; Azure, 2024; Google Cloud, 2024).

Within this broader shift, the rise of cloud-native data warehouses represents not merely a technological substitution but a profound epistemic and organizational change in how data is produced, curated, and interpreted. Systems such as Amazon Redshift, Microsoft Azure Synapse Analytics, and Google BigQuery are often described in marketing and technical documentation as scalable, fully managed analytical engines, yet such descriptions obscure the deep architectural and philosophical differences that distinguish them. These platforms are built upon decades of research in column-oriented storage, massively parallel processing, and query optimization, yet they also embody new logics of abstraction, automation, and economic governance that have few historical precedents in traditional database systems (Abadi et al., 2013; Stonebraker et al., 2005).

The academic literature on column-oriented databases provides a critical foundation for understanding the performance characteristics of modern analytical engines. Seminal work on C-Store and MonetDB demonstrated that storing data by columns rather than rows dramatically improves compression efficiency and query execution for read-heavy analytical workloads, particularly those involving aggregation and filtering across large datasets (Stonebraker et al., 2005; Boncz et al., 2005). Subsequent research on vectorized execution, late materialization, and integrated compression further refined these principles, enabling modern systems to exploit CPU caches and memory hierarchies more effectively (Abadi et al., 2006; Sompolski et al., 2011). These ideas are deeply embedded in the execution engines of Redshift, Synapse, and BigQuery, yet their realization in cloud environments introduces new constraints and opportunities that are not fully captured by classical database theory.

Cloud-native data warehouses are simultaneously software systems and economic platforms, operating within complex ecosystems of data ingestion pipelines, business intelligence tools, and organizational governance structures. This duality is particularly evident in the practical guidance provided by Worlikar, Patel, and Challa, who emphasize that building effective solutions in Amazon Redshift requires not only an understanding of SQL and schema design but also a nuanced appreciation of workload patterns, cluster sizing, data distribution styles, and cost management strategies (Worlikar et al., 2025). Their work illustrates how architectural decisions that appear purely technical are in fact deeply intertwined with business priorities, regulatory requirements, and organizational culture, a theme that resonates with broader debates in information systems research.

Despite the wealth of documentation and case studies available for individual platforms, there remains a significant gap in the scholarly literature regarding how cloud data warehouses should be compared and evaluated in a theoretically rigorous manner. Many existing analyses focus narrowly on performance benchmarks or feature checklists, approaches that implicitly assume that data warehousing platforms are interchangeable commodities differentiated only by technical specifications. Such perspectives overlook the fact that each platform embodies a distinct philosophy of data management, rooted in different historical trajectories and institutional contexts. Amazon Redshift, for example, evolved from the lineage of traditional data warehouse appliances, emphasizing explicit control over clusters and storage layouts, whereas Google BigQuery reflects Google's internal culture of large-scale, serverless computation and highly abstracted data services (AWS, 2024; Google Cloud, 2024).

The problem that motivates this research is therefore not merely how to choose among Redshift, Synapse, and BigQuery, but how to conceptualize what it means to design, operate, and govern a data warehouse in the cloud era. As organizations increasingly rely on these platforms for mission-critical analytics, regulatory reporting, and machine learning pipelines, the stakes of architectural decisions become ever higher. Issues of data sovereignty, reproducibility, cost predictability, and analytical transparency are no longer peripheral concerns but central to the legitimacy and sustainability of digital enterprises (Gartner, 2011; IDC, 2012).

This article seeks to address this gap by developing a comprehensive, theoretically grounded analysis of cloud-native columnar data warehouses, drawing on both foundational database research and contemporary cloud platform documentation. By integrating the practical insights of Worlikar et al. (2025) with the architectural principles articulated in the scholarly literature and vendor documentation, the study aims to illuminate the deep structures that underlie modern analytics infrastructures. The central argument advanced here is that cloud data warehouses should be understood not simply as technological tools but as socio-technical assemblages that mediate between data, computation, and organizational decision-making.

To pursue this argument, the article adopts a comparative framework that examines Amazon Redshift, Azure Synapse Analytics, and Google BigQuery across multiple dimensions of architecture and governance. This approach is grounded in the recognition that each platform represents a particular instantiation of broader design patterns such as massively parallel processing, shared-nothing architectures, and columnar storage, yet implements these patterns in ways that reflect different assumptions about user expertise, workload variability, and economic risk. By unpacking these assumptions, the analysis contributes to a more nuanced understanding of why cloud data warehouses behave as they do and how they might evolve in the future.

The remainder of this article is organized as a continuous analytical narrative that moves from methodological considerations through detailed interpretive results and an extended discussion of theoretical and practical implications. Throughout, every effort is made to situate technical features within their historical and conceptual contexts, thereby avoiding the reductive tendency to treat cloud platforms as black boxes. In doing so, the study aspires to provide a foundation for future research that is both empirically grounded and theoretically rich, advancing the discourse on data warehousing in an era where cloud computing has become the default infrastructure for knowledge production.

Methodology

The methodological foundation of this research is rooted in qualitative comparative architecture analysis, a mode of inquiry that treats technological systems not merely as collections of components but as coherent design philosophies embedded within broader institutional and historical contexts. In contrast to experimental benchmarking or quantitative performance modeling, which often abstract away from real-world complexity, this approach seeks to understand how architectural decisions shape and are shaped by organizational practices, economic incentives, and theoretical traditions in database research (Abadi et al., 2013). Such a methodology is particularly appropriate for the study of cloud-native data warehouses, where performance, scalability, and usability cannot be meaningfully separated from issues of cost governance, data integration, and regulatory compliance.

The primary units of analysis in this study are three major cloud data warehousing platforms: Amazon Redshift, Microsoft Azure Synapse Analytics, and Google BigQuery. These systems were selected because they represent the dominant offerings in the global market and embody distinct architectural paradigms within the shared space of column-oriented, massively parallel analytics engines (Borra, 2024). Redshift is grounded in a cluster-based model that exposes many low-level tuning parameters to users, Synapse integrates data warehousing with broader analytics and transactional processing capabilities within the Azure ecosystem, and BigQuery adopts a serverless model that abstracts away infrastructure management almost entirely (AWS, 2024; Microsoft, 2024; Google Cloud, 2024). The methodological goal is not to rank these platforms but to explicate the logic of their design and the implications of those logics for data-intensive organizations.

The data sources for this analysis consist of three interrelated bodies of literature. First, foundational scholarly works on column-oriented databases, query execution, and storage structures provide the theoretical baseline against which contemporary platforms can be interpreted (Stonebraker et al., 2005; Abadi et al., 2006; Neumann, 2011). These works articulate principles such as vectorized execution, late materialization, and compression-aware query planning that remain central to modern analytical engines, even as they are adapted to cloud environments. Second, vendor documentation from Amazon Web Services, Microsoft Azure, and Google Cloud offers detailed descriptions of system architectures, service boundaries, and operational practices, reflecting the official self-representation of each platform (AWS, 2024; Microsoft, 2024; Google Cloud, 2024). While such documentation is necessarily promotional, it also provides crucial insight into how vendors conceptualize their own systems and the use cases they prioritize. Third, practitioner-oriented texts, most notably Worlikar, Patel, and Challa's Amazon Redshift Cookbook, supply granular, experience-based knowledge about how these systems are actually deployed and managed in production settings (Worlikar et al., 2025).

The analytical process involves an iterative triangulation of these sources. Architectural features described in vendor documentation are interpreted through the lens of database theory, and both are cross-examined against the practical realities documented by practitioners. For example, when Redshift is described as using a massively parallel processing architecture with leader and compute nodes, this is not taken at face value but analyzed in relation to the shared-nothing design principles articulated by Orenstein and Merrett and later refined in column-store research (Orenstein and Merrett, 1984; Stonebraker et al., 2005). Similarly, the serverless abstraction of BigQuery is interpreted in light of historical debates about compilation versus vectorization in query execution, revealing how cloud-scale resource pooling changes the meaning of these trade-offs (Sompolski et al., 2011; Google Cloud, 2024).

A critical component of the methodology is the use of interpretive coding to identify recurring themes across sources. These themes include data distribution strategies, workload isolation mechanisms, cost management models, and governance affordances. Each theme is examined within and across platforms to reveal both commonalities and divergences. For instance, while all three platforms rely on columnar storage and distributed execution, they differ markedly in how they expose or conceal these mechanisms from users, a difference that has profound implications for reproducibility and performance tuning (Worlikar et al., 2025; AWS, 2024).

The limitations of this methodology must also be acknowledged. Because the study does not involve direct experimentation or proprietary performance data, its conclusions are necessarily interpretive rather than predictive. Moreover, vendor documentation may omit or obscure certain architectural details for competitive or security reasons, introducing potential biases into the analysis (Microsoft, 2024; Google Cloud, 2024). To mitigate these limitations, the study relies heavily on cross-referencing multiple sources and situating claims within established theoretical frameworks. By grounding interpretations in the well-documented principles of column-oriented database design, the analysis seeks to avoid overreliance on any single vendor's narrative (Abadi et al., 2013).

Another important methodological constraint arises from the rapidly evolving nature of cloud platforms. Features and pricing models change frequently, and architectural components are often updated without extensive public documentation. In this sense, the study captures a snapshot of these systems as they were conceptualized in the early to mid-2020s, drawing on documentation accessed in 2024 and the practitioner insights of Worlikar et al. (2025). While some details may become outdated, the underlying design philosophies and trade-offs are likely to remain relevant, making the analysis valuable for understanding long-term trends rather than transient configurations.

By adopting this qualitative, theory-informed approach, the study aims to contribute a richer and more nuanced understanding of cloud-native data warehouses than is possible through purely technical or economic analyses. The methodology recognizes that platforms like Redshift, Synapse, and BigQuery are not merely tools but institutionalized infrastructures that shape how organizations engage with data, and it is this broader perspective that informs the results and discussion that follow (Gartner, 2011; IDC, 2012).

Results

The comparative analysis of Amazon Redshift, Azure Synapse Analytics, and Google BigQuery reveals a complex landscape in which ostensibly similar technologies embody profoundly different architectural and organizational logics. Although all three platforms are built upon the foundational principles of column-oriented storage and distributed query execution articulated in the database literature, their implementations diverge in ways that reflect distinct philosophies of control, abstraction, and economic governance (Abadi et al., 2013; Stonebraker et al., 2005). These divergences are not merely technical curiosities but have tangible consequences for how data warehouses are designed, optimized, and experienced by users.

In the case of Amazon Redshift, the results of the analysis indicate a strong continuity with the tradition of appliance-style data warehousing, albeit reimagined within a cloud context. Redshift organizes compute resources into clusters composed of a leader node and multiple compute nodes, each responsible for storing and processing a subset of the data (AWS, 2024). Data is distributed across nodes according to explicit distribution styles, such as key-based or all-replicated, which users must choose based on their query patterns and join characteristics. This design reflects the shared-nothing architecture championed in early parallel database research, where careful data placement is essential for minimizing network traffic and maximizing parallelism (Orenstein and Merrett, 1984; Stonebraker et al., 2005).

Worlikar et al. (2025) emphasize that effective use of Redshift requires a deep understanding of these distribution mechanisms, as poorly chosen distribution keys can lead to data skew, increased query latency, and inflated costs. Their practitioner-oriented recipes illustrate how schema design, workload management queues, and sort keys interact to shape query execution plans, underscoring the degree to which Redshift exposes low-level architectural details to its users. This transparency can be empowering for expert practitioners, enabling fine-grained optimization, but it also imposes a cognitive and operational burden that can be challenging for less specialized teams (Worlikar et al., 2025).

Azure Synapse Analytics, by contrast, presents a more hybridized architectural model that integrates data warehousing with broader analytics and transactional processing capabilities. At its core, Synapse employs a massively parallel processing engine similar in principle to Redshift, distributing data across compute nodes and executing queries in parallel (Microsoft, 2024). However, Synapse also incorporates features such as serverless SQL pools and tight integration with Azure Data Lake Storage, allowing users to query data in place without explicit loading into a traditional warehouse. This reflects a growing trend toward decoupling storage and compute, a design philosophy that seeks to provide greater flexibility and cost efficiency in heterogeneous data environments (Azure, 2024).

The results suggest that Synapse's architecture embodies a compromise between the explicit control of traditional data warehouses and the high-level abstraction of serverless analytics. On one hand, users can still create dedicated SQL pools with defined performance characteristics, enabling predictable query behavior for mission-critical workloads. On the other hand, the availability of serverless pools encourages exploratory and ad hoc analysis without the need for upfront resource provisioning (Microsoft, 2024). This duality aligns with broader developments in database research that explore the integration of transactional and analytical processing, as exemplified by systems like Hekaton in SQL Server (Diaconu et al., 2013).

Google BigQuery represents the most radical departure from traditional data warehousing paradigms among the three platforms. BigQuery is fundamentally serverless, meaning that users do not manage clusters, nodes, or even explicit storage volumes; instead, they interact with a global pool of compute resources orchestrated by Google's internal infrastructure (Google Cloud, 2024). Data is stored in a proprietary, columnar format and queries are executed by dynamically allocated resources that scale automatically based on workload. From the user's perspective, this creates an experience in which infrastructure is almost entirely invisible, replaced by a pay-per-query economic model that charges for the amount of data processed.

The results of the analysis indicate that this level of abstraction fundamentally alters the relationship between users and the underlying system. Whereas Redshift users must think in terms of nodes, slices, and distribution keys, BigQuery users are encouraged to think in terms of datasets and queries, trusting the platform to handle optimization and scaling. This design reflects Google's long-standing emphasis on large-scale, automated infrastructure management, as seen in its internal systems for distributed computing and storage (Google Cloud, 2024). From a theoretical perspective, BigQuery can be seen as an extreme instantiation of the trend toward late binding and just-in-time resource allocation discussed in the database literature, albeit implemented at cloud scale (Neumann, 2011; Sompolski et al., 2011).

Across all three platforms, column-oriented storage remains a unifying principle. Data is stored by column rather than by row, enabling high compression ratios and efficient execution of analytical queries that scan large portions of a table but only a few attributes (Abadi et al., 2006). The results confirm that this design choice is not merely a legacy of earlier research but a living foundation that continues to shape the performance characteristics of modern cloud data warehouses. However, the way in which columnar storage is integrated with distributed execution varies. Redshift and Synapse rely on explicit data distribution to align columnar storage with parallel processing, while BigQuery abstracts this alignment away, relying on its internal execution engine to manage data locality (Worlikar et al., 2025; Microsoft, 2024; Google Cloud, 2024).

Another important result concerns workload isolation and resource management. In Redshift, workload management queues allow administrators to allocate memory and concurrency slots to different classes of queries, effectively partitioning the cluster's resources among competing workloads (AWS, 2024). Worlikar et al. (2025) describe how careful tuning of these queues is essential for maintaining predictable performance in multi-tenant environments. Synapse offers similar capabilities through its SQL pools, but also extends workload isolation to serverless contexts, where queries compete for a shared pool of resources managed by the platform (Microsoft, 2024). BigQuery, in contrast, relies on project-level quotas and reservations to control resource usage, reflecting its more centralized and abstracted approach to governance (Google Cloud, 2024).

These differences have significant implications for cost predictability and governance. Redshift's cluster-based pricing model encourages users to think in terms of reserved capacity, making costs relatively predictable but also creating incentives to keep clusters running even when workloads are light (Worlikar et al., 2025). Synapse's hybrid model introduces more variability, as serverless queries are billed per unit of data processed, while dedicated pools incur fixed costs (Microsoft, 2024). BigQuery's pay-per-query model offers maximum flexibility but can lead to unpredictable expenses if queries are not carefully managed, a concern frequently raised in practitioner communities (Google Cloud, 2024).

Taken together, these results suggest that cloud-native data warehouses cannot be understood solely in terms of their technical features. Each platform embodies a distinct vision of how data, computation, and economic value should be organized in the cloud. Redshift prioritizes explicit control and optimization, Synapse seeks to integrate diverse analytics modalities within a unified ecosystem, and BigQuery advances a highly abstracted, service-oriented model that redefines the boundaries of the data warehouse. These differences set the stage for a deeper theoretical discussion of what cloud data warehousing means in practice and how it reshapes the epistemology of analytics.

Discussion

The findings of this study invite a deeper theoretical reflection on the nature of cloud-native data warehouses as both technological systems and institutional infrastructures. While the comparative analysis of Amazon Redshift, Azure Synapse Analytics, and Google BigQuery reveals important architectural differences, these differences acquire their full significance only when situated within the broader history of database research, the political economy of cloud computing, and the epistemological practices of data-driven organizations

(Abadi et al., 2013; Borra, 2024). In this discussion, the results are interpreted not as isolated observations but as manifestations of enduring tensions between control and abstraction, performance and convenience, and technical rigor and organizational pragmatism.

At the theoretical level, the persistence of column-oriented storage across all three platforms underscores the enduring relevance of ideas first articulated in the early 2000s. The success of C-Store, MonetDB, and related systems demonstrated that analytical workloads fundamentally differ from transactional ones in their access patterns, justifying specialized storage and execution models (Stonebraker et al., 2005; Boncz et al., 2005). Cloud data warehouses inherit this insight, yet they must also contend with the distributed, elastic, and multi-tenant nature of cloud infrastructure. The result is a hybridization of classical database theory with cloud-native design patterns, producing systems that are at once familiar and radically new.

Amazon Redshift’s emphasis on explicit data distribution and cluster management can be interpreted as a conservative extension of the appliance model into the cloud. By exposing distribution keys, sort orders, and workload queues, Redshift allows skilled practitioners to apply the same optimization techniques that were developed for on-premise parallel databases, albeit within a more flexible and scalable infrastructure (Worlikar et al., 2025; AWS, 2024). From a theoretical perspective, this reflects a commitment to the idea that performance emerges from the alignment of data layout and query execution, an idea deeply rooted in the literature on shared-nothing architectures and cost-based optimization (Orenstein and Merrett, 1984; Neumann, 2011).

However, this commitment also carries risks. The need for manual tuning and architectural foresight can become a liability in environments where workloads are unpredictable or where organizations lack specialized expertise. As Worlikar et al. (2025) note, misconfigured clusters and poorly chosen distribution keys can lead to cascading performance problems that are difficult to diagnose and resolve. In this sense, Redshift embodies a trade-off between power and complexity, offering high potential performance at the cost of increased operational burden.

Azure Synapse Analytics occupies a more ambivalent position in this landscape. By combining dedicated SQL pools with serverless query capabilities and deep integration with data lake storage, Synapse reflects a recognition that modern analytics workloads are heterogeneous and cannot be neatly contained within a single architectural paradigm (Microsoft, 2024; Azure, 2024). This hybridity resonates with scholarly efforts to bridge transactional and analytical processing, as seen in systems like Hekaton, which seek to eliminate the traditional divide between operational and analytical data stores (Diaconu et al., 2013).

Yet hybridity also introduces its own challenges. The coexistence of multiple execution models within a single platform can create confusion about best practices, cost management, and performance tuning. From a governance perspective, organizations must decide which workloads belong in which pools, a decision that has both technical and political dimensions. In this sense, Synapse exemplifies the growing complexity of cloud analytics ecosystems, where flexibility is purchased at the price of increased coordination and oversight.

Google BigQuery’s serverless model represents the most radical reimagining of the data warehouse. By abstracting away infrastructure management and charging users based on data processed, BigQuery shifts the locus of control from the organization to the platform provider (Google Cloud, 2024). This aligns with broader trends in cloud computing toward managed services and platformization, where vendors assume responsibility for scaling, optimization, and availability in exchange for greater influence over how resources are used (Amazon Web Services, 2024; Azure, 2024).

From a theoretical standpoint, BigQuery challenges traditional notions of database tuning and performance engineering. Classical research on query optimization, vectorization, and compilation assumes a relatively stable hardware environment in which algorithms can be tailored to specific resource constraints (Sompolski et al., 2011; Neumann, 2011). In a serverless context, however, the hardware is ephemeral and opaque, making such tailoring both difficult and arguably unnecessary. Instead, optimization becomes a service provided by the platform, raising questions about transparency, reproducibility, and trust.

These questions are not merely technical but epistemological. Data warehouses are not neutral repositories of facts; they are instruments through which organizations construct and validate knowledge about their operations, customers, and environments. The degree to which users can understand and control the behavior of their data warehouse affects the credibility and interpretability of the insights it produces (Gartner, 2011; IDC, 2012). Redshift's transparency enables detailed forensic analysis of query plans and data distribution, supporting a culture of technical accountability. BigQuery's abstraction, while convenient, may obscure the causal mechanisms behind performance and cost, potentially undermining users' ability to reason about their own data practices.

The practitioner insights of Worlikar et al. (2025) are particularly illuminating in this regard. Their detailed recipes for schema design, workload management, and performance tuning reveal a tacit knowledge that cannot be easily codified in vendor documentation or automated by platform services. This tacit knowledge represents a form of organizational capital, developed through experience and experimentation, that shapes how data warehouses are used and understood. In a highly abstracted environment like BigQuery, the scope for such experiential learning may be reduced, shifting the balance of expertise from users to vendors.

At the same time, abstraction can be empowering, especially for organizations that lack the resources to maintain specialized data engineering teams. By lowering the barrier to entry for large-scale analytics, serverless platforms democratize access to data-driven decision-making, a goal that has long been championed in both academic and industry discourse (Borra, 2024; Google Cloud, 2024). The challenge is to balance this democratization with the need for accountability, governance, and methodological rigor.

The comparative results also highlight the importance of economic models in shaping architectural outcomes. Redshift's cluster-based pricing encourages users to think in terms of capacity planning and long-term commitments, reinforcing a relatively static conception of workload demand (Worlikar et al., 2025). BigQuery's pay-per-query model, by contrast, treats computation as a fluid commodity, aligning costs more closely with usage but also introducing volatility and the risk of runaway expenses (Google Cloud, 2024). Synapse's hybrid model reflects an attempt to reconcile these competing logics, but it also inherits their respective tensions (Microsoft, 2024).

From a policy and governance perspective, these economic models have far-reaching implications. Organizations must decide not only which platform to use but how to structure internal incentives and controls to align data usage with strategic objectives. In environments where data access is cheap and easy, there is a risk of analytical sprawl, where redundant queries and poorly designed pipelines consume resources without delivering commensurate value (Gartner, 2011). Conversely, overly restrictive controls can stifle innovation and discourage exploratory analysis.

Future research must therefore move beyond narrow technical evaluations to address the socio-technical dynamics of cloud data warehousing. One promising avenue is the study of how organizations negotiate the boundary between platform-provided automation and user-driven optimization, a boundary that is continually shifting as vendors introduce new features and abstractions (Worlikar et al., 2025; AWS, 2024). Another important area is the investigation of data governance in multi-tenant, globally distributed environments, where legal, ethical, and technical considerations intersect in complex ways (Azure, 2024; Google Cloud, 2024).

In theoretical terms, cloud-native data warehouses invite a rethinking of classic database concepts such as schema, index, and query plan. When storage and compute are decoupled and managed by a platform, these concepts become less about physical structures and more about logical contracts between users and services. This shift echoes broader trends in software engineering toward service-oriented and microservice architectures, suggesting fruitful opportunities for cross-disciplinary dialogue between database researchers and scholars of distributed systems (Abadi et al., 2013; Neumann, 2011).

Ultimately, the significance of platforms like Redshift, Synapse, and BigQuery lies not only in their ability to process large volumes of data but in their role as infrastructures of knowledge. They mediate what can be known, how quickly it can be known, and at what cost, shaping the epistemic practices of organizations in subtle but profound ways. By situating these platforms within a rich theoretical and historical context, this study seeks to illuminate those mediations and to provide a foundation for more reflective and responsible use of cloud-native analytics.

Conclusion

This research has sought to provide a comprehensive, theoretically grounded examination of cloud-native columnar data warehouses through the comparative analysis of Amazon Redshift, Azure Synapse Analytics, and Google BigQuery. By integrating foundational database theory, cloud computing literature, vendor documentation, and practitioner insights, particularly those articulated in Worlikar, Patel, and Challa's Amazon Redshift Cookbook, the study has demonstrated that these platforms are far more than interchangeable technical tools; they are embodiments of distinct architectural philosophies and institutional logics (Worlikar et al., 2025; Abadi et al., 2013).

The analysis has shown that while all three platforms draw on the same core principles of column-oriented storage and massively parallel processing, they diverge in how they balance control and abstraction, performance and convenience, and predictability and flexibility. These divergences have significant implications for data governance, cost management, and the epistemology of analytics in contemporary organizations. As cloud computing continues to reshape the landscape of data-intensive work, understanding these deeper structures becomes ever more critical.

By framing cloud data warehouses as socio-technical systems that mediate between data, computation, and organizational decision-making, this study contributes a richer and more nuanced perspective to the discourse on modern analytics infrastructure. It is hoped that this perspective will inform both scholarly inquiry and practical design, supporting the development of data platforms that are not only powerful and scalable but also transparent, accountable, and aligned with the values of the organizations and societies they serve.

References

1. Abadi, D., Boncz, P., Harizopoulos, S., Idreos, S., and Madden, S. The Design and Implementation of Modern Column-Oriented Database Systems. Foundations and Trends in Databases, 2013, 5(3), 197–280.
2. Microsoft Azure. Introduction to Azure Synapse Analytics. Available at https://learn.microsoft.com/en-us/azure/synapseanalytics/overview-what-is. Accessed May 31, 2024.
3. Sompolski, J., Zukowski, M., and Boncz, P. A. Vectorization vs. compilation in query execution. In DaMoN, 2011, 33–40.
4. Worlikar, S., Patel, H., and Challa, A. Amazon Redshift Cookbook: Recipes for building modern data warehousing solutions. Packt Publishing Ltd., 2025.
5. Google Cloud. BigQuery Architecture. Available at https://cloud.google.com/blog/products/data-analytics/new-blog-series-bigquery-explained-overview. Accessed May 31, 2024.
6. Orenstein, J. A., and Merrett, T. H. A class of data structures for associative searching. In Proceedings of PODS, 1984, 181–190.
7. Amazon Web Services. What is Cloud Computing? Available at https://aws.amazon.com/what-is-cloud-computing/. Accessed May 31, 2024.
8. Diaconu, C., Freedman, C., Ismert, E., Larson, P.-Å., Mittal, P., Stonecipher, R., Verma, N., and Zwilling, M. Hekaton: SQL Server's memory-optimized OLTP engine. SIGMOD Conference, 2013, 1243–1254.
9. Google Cloud. Introduction to Google BigQuery. Available at https://cloud.google.com/bigquery/docs/introduction. Accessed May 31, 2024.
10. Borra, P. An Overview of Cloud Computing and Leading Cloud Service Providers. International Journal of Computer Engineering and Technology, 2024, 15(3), 122–133.
11. Neumann, T. Efficiently Compiling Efficient Query Plans for Modern Hardware. PVLDB, 2011.
12. Amazon Web Services. Amazon Redshift Overview. Available at https://docs.aws.amazon.com/redshift/latest/dg/c_redshift_system_overview.html. Accessed May 31, 2024.

13. Stonebraker, M., Abadi, D. J., Batkin, A., Chen, X., Cherniack, M., Ferreira, M., Lau, E., Lin, A., Madden, S. R., O'Neil, E. J., O'Neil, P. E., Rasin, A., Tran, N., and Zdonik, S. B. C-Store: A Column-Oriented DBMS. Proceedings of VLDB, 2005, 553–564.

14. Gartner. User Survey Analysis: Key Trends Shaping the Future of Data Center Infrastructure Through 2011.

15. Microsoft Azure. Azure Synapse Architecture Overview. Available at https://learn.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/massively-parallel-processing-mpp-architecture. Accessed May 31, 2024.

16. Google Cloud. What is Cloud Architecture? Available at https://cloud.google.com/learn/what-is-cloud-architecture. Accessed May 31, 2024.

17. Abadi, D. J., Madden, S. R., and Ferreira, M. Integrating compression and execution in column-oriented database systems. Proceedings of the ACM SIGMOD Conference on Management of Data, 2006, 671–682.

18. IDC. Worldwide Business Analytics Software 2012–2016 Forecast and 2011 Vendor Shares.

19. Amazon Web Services. Amazon Redshift Management Guide. Available at https://docs.aws.amazon.com/redshift/latest/dg/c_high_level_system_architecture.html. Accessed May 31, 2024.

20. Microsoft Azure. What is Cloud Computing? Available at https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-cloud-computing. Accessed May 31, 2024.

21. Boncz, P., Zukowski, M., and Nes, N. MonetDB/X100: Hyper-pipelining query execution. Proceedings of CIDR, 2005.

22. Guido Moerkotte. Small Materialized Aggregates: A Light Weight Index Structure for Data Warehousing. VLDB Proceedings, 1998, 476–487.