ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

A Systematic Literature Review on the Performance Trade-offs of Centralized, Distributed, and Cloud Database Systems

Muhammad Shidiq Pamungkas

Magister Ilmu Komputer, Universitas Budi Luhur
Jakarta
Email: 2211601006@student.budiluhur.ac.id

ABSTRACT

The rapid growth of modern information systems, cloud computing, and data-intensive applications has increased the demand for database architectures capable of delivering high performance, scalability, reliability, and efficient resource management. This study presents a Systematic Literature Review (SLR) on the performance trade-offs among centralized, distributed, and cloud database systems. The research adopted a PRISMA-inspired Google Scholar Screening Workflow to identify, screen, and synthesize relevant studies published between 2018 and 2026. Literature selection was conducted using predefined keywords related to database performance, benchmarking, scalability, and architectural evaluation. A total of 29 selected studies were analyzed using qualitative synthesis methods focusing on evaluation metrics, architectural characteristics, benchmarking approaches, and performance outcomes. The findings indicate that centralized databases remain effective for environments requiring strong consistency, simplified administration, and controlled workloads, although they face limitations in scalability and fault tolerance. Distributed databases demonstrate superior performance in horizontal scalability, redundancy, and distributed workload processing but involve greater synchronization and transaction management complexity. Meanwhile, cloud databases provide deployment flexibility, elastic resource allocation, and service-based management models, while introducing trade-offs concerning operational cost, security, consistency, and deployment complexity. The review also identifies latency, throughput, response time, scalability, availability, and resource utilization as the most commonly used database performance metrics, with benchmarking frameworks such as Yahoo! Cloud Serving Benchmark (YCSB) playing a significant role in performance evaluation. The study concludes that no single database architecture is universally optimal, and database selection should be aligned with workload characteristics, business requirements, and deployment environments.

Keywords : *Centralized Database, Distributed Database, Cloud Database, Database Performance, Systematic Literature Review*

ABSTRAK

Perkembangan pesat sistem informasi modern, komputasi awan, dan aplikasi berbasis data telah meningkatkan kebutuhan terhadap arsitektur basis data yang mampu menyediakan performa tinggi, skalabilitas, reliabilitas, serta pengelolaan sumber daya yang efisien. Penelitian ini menyajikan Systematic Literature Review (SLR) mengenai trade-off performa antara centralized database, distributed database, dan cloud database systems. Penelitian menggunakan pendekatan PRISMA-inspired Google Scholar Screening Workflow untuk mengidentifikasi, menyaring, dan mensintesis literatur relevan yang dipublikasikan pada rentang tahun 2018–2026. Proses seleksi literatur dilakukan menggunakan kata kunci yang berkaitan dengan performa basis data, benchmarking, skalabilitas, serta evaluasi arsitektur database. Sebanyak 29 studi terpilih dianalisis menggunakan pendekatan sintesis kualitatif dengan fokus pada metrik evaluasi, karakteristik arsitektur, metode benchmarking, dan hasil performa. Hasil penelitian menunjukkan bahwa centralized database tetap efektif pada lingkungan yang membutuhkan konsistensi tinggi, administrasi sederhana, dan workload terkontrol, namun memiliki keterbatasan pada aspek scalability dan fault tolerance. Distributed database menunjukkan keunggulan pada horizontal scalability, redundancy, dan pemrosesan distributed workloads, meskipun melibatkan kompleksitas sinkronisasi dan transaction management yang lebih tinggi. Sementara itu, cloud database menawarkan fleksibilitas deployment, elastic resource allocation, dan model layanan berbasis cloud, namun menghadirkan trade-off terkait biaya operasional, keamanan, konsistensi, dan kompleksitas implementasi. Penelitian ini juga menemukan bahwa latency, throughput, response time, scalability, availability, dan resource utilization merupakan metrik evaluasi yang paling umum digunakan, dengan framework benchmarking seperti Yahoo! Cloud Serving Benchmark (YCSB) berperan penting dalam evaluasi performa database. Penelitian menyimpulkan bahwa tidak terdapat satu arsitektur basis data yang optimal untuk seluruh skenario implementasi sehingga pemilihan database perlu disesuaikan dengan karakteristik workload, kebutuhan bisnis, dan lingkungan deployment.

Kata Kunci: *Centralized Database, Distributed Database, Cloud Database, Database Performance, Systematic Literature Review*

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

1. Introduction

The rapid development of information technology has significantly increased the volume, velocity, and complexity of data across various sectors, including enterprise systems, cloud services, web applications, big data analytics, and Internet of Things (IoT) environments. Consequently, organizations require database management systems capable of delivering high performance, scalability, reliability, and efficient data management to support modern information systems (Kleppmann, 2017). In practice, several database architectures are commonly employed, including centralized databases, distributed databases, and cloud databases, each offering distinct characteristics, advantages, and limitations.

A centralized database represents the traditional approach in which all data are stored and managed within a single central location. This architecture simplifies administration, consistency control, and data governance. However, centralized systems may face challenges related to scalability, fault tolerance, and availability when operating under large-scale workloads or geographically distributed environments (Redžepagić et al., 2026). In contrast, distributed databases provide data distribution across multiple nodes, improving scalability, redundancy, and fault tolerance. Nevertheless, distributed environments introduce additional challenges concerning synchronization, consistency maintenance, and system complexity (Guo et al., 2024).

The emergence of cloud computing has further accelerated the adoption of cloud databases and Database-as-a-Service (DBaaS) platforms due to their flexibility, elastic scalability, and service-oriented deployment models. Cloud databases enable organizations to reduce infrastructure costs while dynamically adapting resources according to changing workloads (Khan et al., 2022). Despite these advantages, cloud databases also involve performance trade-offs associated with operational costs, query processing, deployment complexity, security, and reliability (Haubenschild & Leis, 2025).

Numerous studies have investigated database performance from different perspectives. Makris et al. (2021) compared PostgreSQL and MongoDB regarding query execution and data management efficiency. Abu Kausar et al. (2022) evaluated MongoDB, Cassandra, and Redis using the Yahoo! Cloud Serving Benchmark (YCSB) to analyze throughput and response time across different workloads. Khan et al. (2023) conducted a systematic literature review on SQL and NoSQL database architecture performance, while Taipalus (2024) presented a systematic review of database management system performance comparisons. In the context of cloud databases, Ferreira et al. (2024) examined the impact of consistency levels on cloud-based NoSQL performance, whereas Hentschel et al. (2025) discussed the development of cloud-oriented data management systems.

Although database performance has been extensively studied, most previous works primarily focus on SQL versus NoSQL comparisons, benchmarking of specific DBMS technologies, or isolated cloud database evaluations. There remains a limited number of studies that comprehensively synthesize the performance trade-offs among centralized, distributed, and cloud database systems using an integrated evaluation framework involving latency, throughput, scalability, availability, reliability, and cost considerations. This research gap highlights the need for a systematic literature review that consolidates empirical findings across different database architectures.

Based on this background, the research problems addressed in this study are as follows: (1) how do centralized, distributed, and cloud database systems differ in terms of performance characteristics according to recent scientific literature; (2) what evaluation metrics are most commonly used in database performance assessment; and (3) what performance trade-offs exist among database architectures in modern information system environments.

This study aims to conduct a Systematic Literature Review (SLR) of studies concerning centralized, distributed, and cloud database performance, identify commonly used performance evaluation metrics, compare the characteristics of each architecture, and synthesize the major performance trade-offs relevant to modern information systems.

The research adopts a Systematic Literature Review approach guided by a PRISMA-inspired Google Scholar Screening Workflow. Literature identification was primarily conducted through Google Scholar using predefined keywords such as centralized database performance, distributed database performance, cloud database performance, database benchmarking, and SQL and NoSQL performance comparison. The screening process involved title and abstract filtering, full-text eligibility assessment, and final study selection based on predefined inclusion and exclusion criteria. Selected studies focused on publications from 2018–2026 discussing performance metrics, benchmarking methods, scalability, latency, throughput, and database architecture evaluation within centralized, distributed, and cloud computing environments.

The expected outcome of this study is to provide a structured literature synthesis regarding the performance characteristics of centralized, distributed, and cloud database systems, while also offering academic and practical insights for selecting suitable database architectures in modern information systems.

2. Discussion

2.1 Research Methodology

This study employed a Systematic Literature Review (SLR) approach to identify, evaluate, analyze, and synthesize previous studies related to the performance of centralized databases, distributed databases, and cloud database systems. The SLR method was selected because it enables a systematic, transparent, and structured examination of research findings, allowing researchers to identify research trends, empirical evidence, and research gaps across existing studies (Khan et al., 2023; Taipalus, 2024).

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

The research process adopted a PRISMA-inspired Google Scholar Screening Workflow, with Google Scholar serving as the primary literature source. Google Scholar was selected because of its broad indexing coverage, including journal articles, conference proceedings, academic books, and multidisciplinary scholarly publications.

The literature search was conducted using combinations of the following keywords: centralized database performance, distributed database performance, cloud database performance, database benchmarking, SQL and NoSQL performance comparison, database architecture trade-offs.

The inclusion criteria applied in this study were (1) publications published between 2018 and 2026, (2) journal articles, conference proceedings, or academic books relevant to the research topic, (3) studies discussing database performance, benchmarking, scalability, throughput, latency, availability, or database architecture evaluation, and (4) studies focusing on centralized databases, distributed databases, cloud databases, SQL/NoSQL systems, or Database-as-a-Service (DBaaS).

The exclusion criteria included: (1) non-English publications; (2) studies primarily discussing cloud infrastructure without substantial database performance analysis; (3) studies lacking performance evaluation, benchmarking results, or database architecture discussion; (4) publications with weak relevance to the objectives of this study; and (5) duplicate records or studies falling outside the scope of centralized, distributed, and cloud database architectures.

The literature analysis employed a qualitative synthesis approach, in which studies were grouped according to database category, evaluation methodology, performance metrics, and major research findings. Comparative analysis was then conducted to identify performance trade-offs among different database architectures.

2.2 Literature Identification and Screening

The literature identification and screening procedures followed a workflow inspired by the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) framework and adapted to the Google Scholar search process.

The first stage consisted of identification, during which publications were collected using predefined keywords. The second stage involved screening, where studies were filtered according to title, abstract, and topic relevance. The third stage comprised eligibility assessment, in which full-text evaluation was performed to determine whether studies satisfied the predefined inclusion criteria. The final stage consisted of selecting the included studies used in the qualitative synthesis.

Based on the literature retrieval process, an initial total of 126 records was identified. Following duplicate removal and preliminary filtering, 88 studies proceeded to the screening stage. Subsequently, 43 full-text articles underwent eligibility assessment, resulting in 29 final studies included in this systematic literature review.

Figure 1 illustrates the sequential literature selection procedure applied to ensure study quality, methodological consistency, and relevance to the research objectives. This multi-stage screening process improved the transparency and reliability of the study selection process.

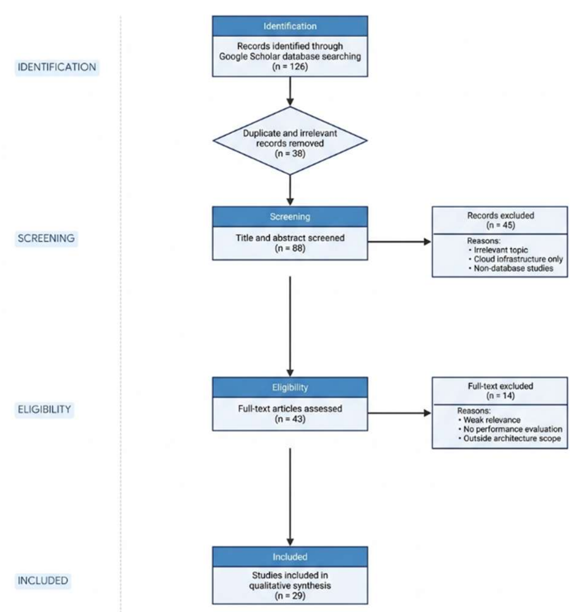

Figure 1. *PRISMA-Inspired Google Scholar Literature Screening Workflow*

2.3 Literature Mapping Analysis

The literature mapping stage was conducted to identify the main characteristics of the studies included in this review. The mapping process categorized studies according to database type, research methodology, evaluation metrics, and primary research findings.

Table 1. *Literature Mapping of Selected Studies*

Study	Focus & Method	Key Findings
Makris et al. (2021)	SQL–NoSQL comparison; comparative evaluation	MongoDB improved workload speed; PostgreSQL ensured stronger consistency
Abu Kausar et al. (2022)	NoSQL benchmarking using YCSB	Cassandra achieved high throughput under distributed workloads
Khan et al. (2023)	Systematic review of SQL	Database performance depends on

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

Study	Focus & Method	Key Findings
	and NoSQL architectures	architectural trade-offs and use cases
Ferreira et al. (2024)	Cloud NoSQL experimental evaluation	Consistency settings significantly influence cloud database performance
Guo et al. (2024)	Distributed database energy-performance modeling	Query execution affects both performance and energy efficiency
Taipalus (2024)	Systematic review of DBMS comparisons	Benchmarking is a critical factor in database performance evaluation
Haubenschild & Leis (2025)	Cloud OLTP architecture analysis	Cloud databases involve performance–cost trade-offs
Redžepagić et al. (2026)	Centralized vs distributed empirical study	Distributed architectures provide superior scalability
Pantelic et al. (2026)	SQL/NoSQL benchmarking in microservices	Performance varies across heterogeneous workloads

Source: synthesized from selected studies included in the systematic literature review.

Based on Table 1, it can be observed that modern database research predominantly employs benchmarking approaches, experimental evaluations, and systematic literature reviews. The most frequently used performance metrics include latency, throughput, response time, scalability, availability, and resource utilization. In addition, many studies emphasize the existence of trade-offs among performance, consistency, operational cost, and implementation complexity.

The literature mapping results indicate that distributed databases and cloud databases have attracted substantial research attention between 2021 and 2026, particularly in relation to modern information systems requiring high scalability, flexibility, and high-performance processing within cloud and distributed environments.

2.4 Centralized Database Performance Analysis

Centralized databases represent the traditional database architecture in which all data storage, processing, and management activities are handled within a single server or centralized location. This model remains widely adopted in small-scale and medium-scale information systems because it offers simplified administration, centralized control, and efficient transaction consistency management (Kleppmann, 2017).

One of the primary advantages of centralized databases lies in their ability to maintain data consistency, simplify query management, and reduce administrative complexity compared to distributed environments. Andrea et al. (2025) demonstrated that relational database management systems such as MySQL, PostgreSQL, Oracle, and SQL Server can achieve strong performance in processing large-scale text data when operating within controlled environments. Their findings indicate that centralized relational databases remain highly relevant in scenarios requiring strong consistency guarantees and structured data processing.

Database performance in centralized systems is also influenced by security mechanisms and data protection techniques. Carvalho et al. (2025) investigated the impact of AES encryption on query read performance across Oracle, MySQL, and SQL Server databases. The study revealed that security mechanisms may introduce performance overhead; however, centralized relational DBMS platforms generally preserve efficient query execution under various experimental conditions.

Despite these advantages, centralized database systems possess several inherent limitations. As the volume of data, number of concurrent transactions, and number of users increase significantly, centralized architectures may encounter challenges related to scalability, availability, and fault tolerance. Redžepagić et al. (2026) found that centralized architectures demonstrate limitations in handling distributed workloads when compared with distributed database environments implemented in Kubernetes infrastructures.

These limitations primarily arise from the existence of a single point of failure, whereby failures occurring at the central server can affect the entire database service. Consequently, centralized databases are generally more suitable for environments characterized by moderate workloads, strong governance requirements, and limited distribution complexity.

2.5 Distributed Database Performance Analysis

Distributed databases refer to database architectures in which data storage, processing, and access operations are distributed across multiple nodes or geographically separated locations. The increasing demand for highly scalable, fault-tolerant, and geographically distributed information systems has significantly accelerated the adoption of distributed database technologies (Kleppmann, 2017).

One of the most significant strengths of distributed databases is their capability to provide horizontal scalability through node-level data distribution. This approach allows system capacity to be expanded without relying solely on upgrading a single server. Redžepagić et al. (2026) demonstrated that distributed database architectures outperform centralized architectures in

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

terms of scalability, resource allocation, and distributed workload handling.

Abu Kausar et al. (2022) employed the Yahoo! Cloud Serving Benchmark (YCSB) to compare MongoDB, Cassandra, and Redis within distributed NoSQL environments. Their results showed that Cassandra delivered high throughput under distributed workloads, whereas Redis exhibited lower latency in memory-intensive operations. These findings suggest that distributed database performance strongly depends on workload characteristics and data distribution strategies.

Guo et al. (2024) evaluated distributed database systems using a bottom-up comprehensive energy model to estimate query-related energy consumption. The study indicated that distributed query processing affects not only throughput and latency but also energy efficiency. Therefore, performance optimization in distributed database systems must consider both computational efficiency and resource consumption.

Within cloud computing environments, Prasetio et al. (2023) analyzed distributed databases and found that distributed architectures improve availability and resource utilization in modern cloud systems. Similar findings were reported by Ullah et al. (2024), who demonstrated that hybrid cloud distributed processing frameworks offer substantial improvements in scalability and computational flexibility.

Despite these advantages, distributed databases introduce several operational challenges. Data synchronization, distributed transaction management, cross-node query optimization, and consistency maintenance remain critical issues affecting overall system performance. He et al. (2024), in their benchmarking study of distributed transactional database systems, reported that performance is significantly influenced by workload configuration, transaction models, and concurrency control mechanisms.

Therefore, distributed databases are particularly suitable for large-scale enterprise systems, multi-node applications, geographically distributed infrastructures, and environments requiring high fault tolerance and scalable resource management.

2.6 Cloud Database Performance Analysis

The rapid evolution of cloud computing has accelerated the adoption of cloud databases and Database-as-a-Service (DBaaS) solutions across modern information systems. Cloud databases provide deployment flexibility, elastic scalability, and service-based infrastructure management, enabling organizations to reduce hardware investment and local maintenance costs (Khan et al., 2022).

One of the defining characteristics of cloud databases is resource elasticity, which enables computing and storage resources to be dynamically adjusted according to workload requirements. Haubenschild and Leis (2025) explained that cloud OLTP architectures provide high flexibility for transaction processing but involve trade-offs concerning throughput, operational cost, and deployment complexity.

Cloud databases also support the implementation of modern architectures such as hybrid databases, distributed warehouses, and cloud-native data management systems. Rabelo Ferreira and do Nascimento Fidalgo (2024) demonstrated that hybrid and columnar cloud database architectures can improve schema efficiency within distributed Data Warehouse as a Service environments.

In cloud-based NoSQL systems, Ferreira et al. (2024) observed that data consistency levels significantly affect system performance. More relaxed consistency settings may improve response time and throughput but can potentially reduce data consistency guarantees. These findings illustrate the classical performance–consistency trade-off frequently observed in cloud database environments.

Beyond performance considerations, cloud databases must also address challenges related to security, privacy, and query processing. Khan et al. (2022) conducted an analytical study of DBaaS platforms and found that cloud database architectures involve challenges associated with security management, privacy preservation, query optimization, and service orchestration. Consequently, cloud database evaluation should not focus exclusively on latency or throughput but should also consider reliability, trustworthiness, and regulatory compliance.

Hentschel et al. (2025) and Lutsch et al. (2025) emphasized that future cloud DBMS platforms must integrate high-performance processing, trusted execution mechanisms, and deployment efficiency to satisfy the demands of modern cloud environments. Therefore, cloud databases represent a highly promising solution for contemporary information systems; however, selecting cloud-based architectures requires careful consideration of scalability, cost, security, and operational complexity.

2.7 Comparative Performance Trade-offs

Based on the literature synthesis, each database architecture exhibits distinct performance characteristics depending on system requirements, workload patterns, and deployment environments. Centralized databases, distributed databases, and cloud databases cannot be universally considered superior solutions, as each architecture involves specific trade-offs across multiple evaluation metrics.

Centralized databases generally demonstrate advantages in data consistency, administrative simplicity, and query performance within controlled environments and structured transactional workloads (Andrea et al., 2025). However, centralized architectures face limitations regarding scalability and availability because they rely on a single management location (Redžepagić et al., 2026).

Distributed databases excel in horizontal scalability, fault tolerance, and support for distributed workloads. Studies by Abu Kausar et al. (2022) and Guo et al. (2024) indicate that distributed database systems can achieve high throughput and improved resource utilization within large-scale environments. Nevertheless, distributed systems introduce challenges related to synchronization, distributed transaction complexity, and inter-node communication overhead.

Cloud databases provide deployment flexibility, elastic scalability, and service-oriented management models. These environments allow organizations to dynamically allocate resources according to workload demands (Haubenschild & Leis, 2025). However, cloud environments also introduce trade-offs involving operational costs, deployment complexity, security management, and consistency control (Khan et al., 2022).

Table 2. *Comparative Performance Trade-offs of Database Architectures*

Metric	Centralized Database	Distributed / Cloud Database
Latency & Throughput	Low latency and stable throughput under controlled workloads (Andrea et al., 2025)	Higher throughput but latency varies due to synchronization, networking, and consistency settings (Abu Kausar et al., 2022; Ferreira et al., 2024)
Scalability	Limited by single-server architecture (Kleppmann, 2017)	High scalability through horizontal scaling and elastic resource allocation (Guo et al., 2024; Khan et al., 2022)
Availability & Fault Tolerance	Vulnerable to single point of failure (Redžepagić et al., 2026)	Improved through redundancy, replication, and cloud resilience (Prasetio et al., 2023; Hentschel et al., 2025)
Complexity & Maintenance	Easier administration and lower maintenance complexity (Andrea et al., 2025)	Greater operational complexity due to distributed coordination and orchestration (He et al., 2024; Khan et al., 2022)
Cost & Deployment	Predictable infrastructure model but limited flexibility (Carvalho et al., 2025)	Flexible deployment with subscription or multi-node cost implications (Haubenschild & Leis, 2025)
Security Management	Centralized governance simplifies control (Carvalho et al., 2025)	Shared-responsibility model with more complex security requirements (Khan et al., 2022; Lutsch et al., 2025)

Source: synthesized from Andrea et al. (2025), Abu Kausar et al. (2022), Carvalho et al. (2025), Ferreira et al. (2024), Guo et al. (2024), Haubenschild and Leis (2025), He et al. (2024), Hentschel et al. (2025), Khan et al. (2022), Kleppmann (2017), Lutsch et al. (2025), Prasetio et al. (2023), and Redžepagić et al. (2026).

Based on Table 2, no single database architecture dominates all evaluation dimensions. Performance behavior is influenced by workload characteristics, deployment context, consistency requirements, and resource constraints reported across empirical studies.

2.8 Benchmarking and Performance Metrics

Benchmarking constitutes one of the most widely used approaches in evaluating modern database performance. Most studies employ metrics such as latency, throughput, response time, resource utilization, availability, and scalability to assess database efficiency (Taipalus, 2024).

One of the most frequently adopted benchmarking frameworks in database research is the Yahoo! Cloud Serving Benchmark (YCSB). This framework was designed to evaluate database systems under workload scenarios resembling modern application access patterns. Abu Kausar et al. (2022) applied YCSB to compare MongoDB, Cassandra, and Redis and identified substantial differences in throughput and latency across NoSQL platforms.

Beyond traditional performance measurement, contemporary benchmarking research increasingly incorporates aspects such as tail latency, workload variability, and benchmark reliability. Fruth et al. (2021) argued that database benchmarking methodologies may produce biased evaluations if workload configurations, latency measurements, or experimental setups are improperly designed. Consequently, benchmarking validity represents a critical aspect of database performance interpretation.

Liyanage et al. (2025) proposed benchmarking extensions involving varying value sizes to generate more realistic evaluations of modern database systems. Their findings demonstrated that payload size variation significantly affects throughput, latency, and query execution behavior.

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

In distributed database environments, benchmarking is also used to evaluate distributed transactions, concurrency control, and scalability behavior. He et al. (2024) demonstrated that heterogeneous workloads and transaction models directly influence the performance of distributed transactional database systems.

The literature synthesis indicates that selecting appropriate benchmarking frameworks is essential for achieving consistent, reproducible, and representative database performance evaluations under real-world deployment conditions.

2.9 Discussion of Research Findings

This section addresses the research questions formulated in the introduction.

Answer to RQ1: Database Performance Evaluation Metrics

The literature review reveals that the most frequently employed database performance evaluation metrics include latency, throughput, response time, scalability, availability, resource utilization, consistency, and operational cost.

Taipalus (2024) and Khan et al. (2023) identified latency and throughput as the most dominant indicators in database performance assessment. Additionally, recent studies increasingly incorporate metrics such as energy efficiency, deployment complexity, and security overhead to achieve a more comprehensive understanding of database behavior.

Answer to RQ2: Performance Characteristics of Database Architectures

The findings indicate that centralized, distributed, and cloud database architectures exhibit distinct performance characteristics.

Centralized databases excel in simplicity, governance, and transactional consistency but are constrained in scalability and fault tolerance. Distributed databases provide horizontal scaling, redundancy, and resource distribution capabilities while increasing synchronization and transaction management complexity. Cloud databases offer deployment flexibility and elastic resource provisioning but require additional consideration regarding operational cost, vendor dependency, and security management.

Therefore, database architecture selection should consider business requirements, workload characteristics, data volume, and target deployment environments.

Answer to RQ3: Performance Trade-offs in Modern Information Systems

The literature synthesis demonstrates that no database architecture is universally optimal for all implementation scenarios. Performance trade-offs are highly context-dependent.

For small-scale systems requiring strong consistency guarantees, centralized databases remain effective solutions. For enterprise-scale systems, multi-node infrastructures, and distributed workloads, distributed databases provide stronger scalability and availability performance. Meanwhile, cloud-native applications, big data analytics, and elastic workload environments are generally more suitable for cloud databases or DBaaS models.

These findings are consistent with studies by Khan et al. (2022), Haubenschild and Leis (2025), and Redžepagić et al. (2026), which emphasize the importance of balancing performance, cost, implementation complexity, and operational requirements when selecting database architectures.

Overall, this study demonstrates that database evaluation should adopt a multidimensional approach that considers not only throughput and latency but also scalability, reliability, security, and deployment trade-offs within modern information system environments.

3. Conclusion

This study conducted a Systematic Literature Review (SLR) on centralized databases, distributed databases, and cloud database systems using a PRISMA-inspired Google Scholar Screening Workflow. Based on the synthesis of selected studies, it can be concluded that database system performance is significantly influenced by architectural characteristics, workload requirements, deployment environments, and data management strategies.

The findings indicate that centralized databases remain suitable for environments requiring strong consistency, simplified administration, and controlled workloads; however, they exhibit limitations in terms of scalability, availability, and fault tolerance. Distributed databases provide advantages in horizontal scalability, redundancy, and support for distributed workloads, although they require more sophisticated synchronization mechanisms, transaction management, and system coordination. Meanwhile, cloud databases offer deployment flexibility, elastic resource provisioning, and service-oriented management models, but involve trade-offs related to operational cost, security, consistency, and deployment complexity.

This study also found that the most commonly used evaluation metrics in modern database research include latency, throughput, response time, scalability, availability, and resource utilization. In addition, benchmarking approaches such as the Yahoo! Cloud Serving Benchmark (YCSB) play an important role in producing consistent and comparable performance evaluations across database architectures.

Based on the research findings, it can be concluded that no single database architecture outperforms others across all implementation scenarios. The selection of a database architecture should therefore consider the balance among performance, scalability, cost, security, and operational requirements of the intended information system.

As a recommendation for future work, subsequent studies may expand the literature scope by incorporating additional scholarly databases such as Scopus, IEEE Xplore, SpringerLink, and ScienceDirect to improve the comprehensiveness of systematic reviews. Future research may also develop meta-analysis studies or experimental benchmarking approaches to directly evaluate centralized, distributed, and cloud database systems using standardized workloads and evaluation metrics. Furthermore, emerging topics such as cloud-native databases, serverless databases, distributed SQL systems, and AI-assisted database optimization represent promising directions for future research in modern information systems.

References

Abughazalah, M., Alsaggaf, W., Saifuddin, S., & Sarhan, S. (2024). Centralized vs. decentralized cloud computing in healthcare. *Applied Sciences*, *14*(17), 7765.

Abu Kausar, M., Nasar, M., & Soosaimanickam, A. (2022). A study of performance and comparison of nosql databases: Mongodb, cassandra, and redis using ycsb. *Indian Journal of Science and Technology*, *15*(31), 1532-1540.

Ali, T. E., Ali, F. I., Hekmat, S., Eyvazov, F., Dakić, P., & Zoltan, A. D. (2025). A Comparative Evaluation of PostgreSQL, MongoDB, and MySQL for Ethiopian Migrant Data Management.

Andrea, V. A., Panjaya, J., & Jingga, K. (2025). Performance Comparison of Relational Database Management Systems for Processing Large Amount of Text Data. *Procedia Computer Science*, *269*, 150-160.

Araújo, C., Oliveira Jr, M., Nogueira, B., Maciel, P., & Tavares, E. (2024). Performability evaluation of NoSQL-based storage systems. *Journal of Systems and Software*, *208*, 111885.

Carvalho, M., Sá, F., & Bernardino, J. (2025). Evaluation of the impact of AES encryption on query read performance across Oracle, MySQL, and SQL Server databases. *Cryptography*, *9*(4), 77.

Ciumac, M., Győrödi, C. A., Győrödi, R. Ș., & Costea, F. M. (2026). Performance Evaluation of MongoDB and RavenDB in IIoT-Inspired Data-Intensive Mobile and Web Applications. *Future Internet*, *18*(1), 57.

Ferreira, S., Mendonça, J., Nogueira, B., Tiengo, W., & Andrade, E. (2024). Impacts of data consistency levels in cloud-based NoSQL for data-intensive applications. *Journal of Cloud Computing*, *13*(1), 158.

Franchi, F., Smarra, F., Di Fina, E., & Galassi, A. (2025). On the Performance of Geospatial Services Deployment: An Experimental Benchmark Analysis: F. Franchi et al. *Journal of Grid Computing*, *23*(2), 17.

Fruth, M., Scherzinger, S., Mauerer, W., & Ramsauer, R. (2021, August). Tell-tale tail latencies: pitfalls and perils in database benchmarking. In *Technology Conference on Performance Evaluation and Benchmarking* (pp. 119-134). Cham: Springer International Publishing.

Globa, L., & Kartashov, A. (2024). Optimizing distributed data storage in multi-cloud environments: algorithmic approach. *Information and Telecommunication Sciences*, (2), 4-12.

Guo, B., Wu, J., Pu, Y., Zhang, J., & Yu, J. (2024). Energy consumption estimation and profiling for queries in distributed database systems based on a bottom-up comprehensive energy model. *Future Generation Computer Systems*, *159*, 379-394.

Haubenschild, M., & Leis, V. (2025). Oltp in the cloud: architectures, tradeoffs, and cost. *The VLDB Journal*, *34*(4), 42.

He, H., Weng, S., Zeng, L., Zhang, H., Zhang, R., Cai, P., ... & Xu, Q. (2024, December). Benchmarking Distributed Transactional Database Systems. In *International Symposium on Benchmarking, Measuring and Optimization* (pp. 37-53). Singapore: Springer Nature Singapore.

Hentschel, M., Dees, J., Funke, F., Heimel, M., & Oukid, I. (2025). Building a data management system for the cloud: lessons learned and future directions. *Datenbank-Spektrum*, *25*(1), 17-28.

Khan, F. A., Jamjoom, M., Ahmad, A., & Asif, M. (2022). An analytic study of architecture, security, privacy, query processing, and performance evaluation of database-as-a-service. *Transactions on emerging telecommunications technologies*, *33*(2), e3814.

Khan, W., Kumar, T., Zhang, C., Raj, K., Roy, A. M., & Luo, B. (2023). SQL and NoSQL database software architecture performance analysis and assessments—a systematic literature review. *Big Data and Cognitive Computing*, *7*(2), 97.

Kleppmann, M. (2017). *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. " O'Reilly Media, Inc.".

Liyanage, D., Pandey, S., Goldstein, J., Cahill, M., Dey, A., Fekete, A., & Röhm, U. (2025, September). A Benchmark for Databases with Varying Value Lengths. In *Technology Conference on Performance Evaluation and Benchmarking* (pp. 1-19). Cham: Springer Nature Switzerland.

Lutsch, A., El-Hindi, M., István, Z., & Binnig, C. (2025). Towards High-performance and Trusted Cloud DBMSs. *Datenbank-Spektrum*, *25*(1), 39-50.

Makris, A., Tserpes, K., Spiliopoulos, G., Zissis, D., & Anagnostopoulos, D. (2021). MongoDB Vs PostgreSQL: A comparative study on performance aspects. *GeoInformatica*, *25*(2), 243-268.

Noor, T. F. I., Nugraha, E., Maknun, J., Kustiawan, I., & Kurşun, E. (2025). Hybrid Database Architecture for Retail Big Data Analytics: PostgreSQL vs MongoDB Performance Analysis. *International Journal of Electronics and Communications Systems*, *5*(2), 205-220.

Pantelic, N., Matic, L., Jakovljevic, L., Eric, S., Eric, M., Stefanović, M., & Djordjevic, A. (2026). Benchmarking SQL and NoSQL Persistence in Microservices Under Variable Workloads. *Future Internet*, *18*(1), 53.

ISSN PRINT : 2302-500X
ISSN ONLINE : 2477-3786

Prasetio, N., Ferdinand, D., Wijaya, B. C., Anggreainy, M. S., & Kumiawan, A. (2023, November). Performance Analysis of Distributed Database System in Cloud Computing Environment. In *2023 IEEE 9th International Conference on Computing, Engineering and Design (ICCED)* (pp. 1-6). IEEE.

Rabelo Ferreira, F. E. R., & do Nascimento Fidalgo, R. (2024). A performance analysis of hybrid and columnar cloud databases for efficient schema design in distributed data warehouse as a service. *Data*, *9*(8), 99.

Redžepagić, J., Kapulica, A., Malešević, N., & Dakić, V. (2026). Empirical Performance and Operational Analysis of Monolithic and Distributed Database Architectures in Kubernetes Environments. *Computers*, *15*(5), 282.

Taipalus, T. (2024). Database management system performance comparisons: A systematic literature review. *Journal of Systems and Software*, *208*, 111872.

Ullah, F., Dhingra, S., Xia, X., & Babar, M. A. (2024). Evaluation of distributed data processing frameworks in hybrid clouds. *Journal of Network and Computer Applications*, *224*, 103837.

Zulkifli, A. (2023). ‘Accelerating database efficiency in complex it infrastructures: Advanced techniques for optimizing performance, scalability, and data management in distributed systems. *International Journal of Information and Cybersecurity*, *7*(12), 81-100.