International Journal of Computer Science and Information Technology Research (IJCSITR)
2023, Vol. 4, No. 2, July - December, pp. 80-85
Journal ID: 9471-1297
website: www.ijcsitr.com

A Comparative Analysis of Relational and Non Relational Database Models in High Volume Data Environments

Friedrich Klaus,

Data Scientist Researcher, Russia.

Abstract

With the proliferation of big data across industries, choosing between relational (SQL) and non-relational (NoSQL) databases has become crucial. This paper compares both database models regarding scalability, consistency, query performance, and suitability in handling large-scale data operations. The study investigates benchmark data and legacy reviews to provide insights into selecting an appropriate model in big data-driven architectures.

Keywords:

Relational databases, NoSQL, high-volume data, database comparison, scalability, big data

How to Cite: Klaus, F. (2023). A Comparative Analysis of Relational and Non Relational Database Models in High Volume Data Environments. *International Journal of Computer Science and Information Technology Research (IJCSITR)*, 4(2), 80–85.

Article Link: https://ijcsitr.com/index.php/home/article/view/IJCSITR_2023_04_02_08/IJCSITR_2023_04_02_08

International Journal of Computer Science and Information Technology Research (IJCSITR)
2023, Vol. 4, No. 2, July - December, pp. 80-85
Journal ID: 9471-1297
website: www.ijcsitr.com

1. Introduction

Relational databases have long served as the backbone of data storage, known for their structured schema and use of SQL for data manipulation. However, with the emergence of web applications and distributed systems, non-relational databases have gained momentum due to their flexibility, horizontal scalability, and schema-less design. High-volume environments—such as IoT, social media, and financial transactions—demand architectures that can handle velocity, variety, and volume simultaneously.

This paper explores how both database paradigms cope with large datasets. It considers architectural differences, the ACID vs. BASE consistency models, and the real-world applicability in high-throughput systems. It also includes evaluations based on response time, storage efficiency, and data processing capabilities to offer an empirical basis for database selection.

2. Literature Review

Several authors explored relational and non-relational databases' efficiency in high-volume environments. Jatana et al. (2012) concluded that NoSQL databases outperform relational ones in scalability and speed when handling massive, unstructured data. Ordonez et al. (2010) compared Hadoop-MapReduce frameworks with traditional SQL and noted that parallelism in NoSQL systems improved data warehousing performance significantly.

Mohamed and Ismail (2014) emphasized that NoSQL databases like MongoDB provide flexible schemas suited for dynamic data models, whereas RDBMSs enforce integrity at the expense of performance. Du Toit (2014) highlighted the performance drawbacks of relational models during concurrent operations and proposed that NoSQL systems handle horizontal scaling better in social media applications.

Györödi et al. (2015) performed benchmarks on SQL Server vs. CouchDB and found a notable difference in read/write operations, favoring NoSQL for write-heavy applications. Salehnia (2015) proposed using relational models for big data teaching due to their rigid structure and standardized query language.

3. Comparison of Data Model and Architecture

Relational databases adhere to a tabular model with predefined schemas and strong consistency. Their structure is ideal for applications requiring atomic transactions and complex joins. However, they scale vertically and are limited in distributed setups.

Non-relational databases (e.g., MongoDB, Cassandra, Redis) offer document-based, key-value, column, or graph-based storage. These databases scale horizontally by sharding data across multiple servers and use eventual consistency for faster access in distributed systems.

Table 1: Key Differences between Relational and Non-Relational Databases

Feature	Relational DB	Non-Relational DB
Schema	Fixed	Flexible
Scalability	Vertical	Horizontal
Data Integrity	Strong (ACID)	Weaker (BASE)
Query Language	SQL	Varies (e.g., JSONQL)
Best for	Structured Data	Unstructured Big Data

4. Performance Metrics in High-Volume Scenarios

Various benchmarks have been conducted to compare read/write latency and throughput in SQL vs NoSQL systems. The following chart demonstrates average query response time when datasets exceed 100 million records:

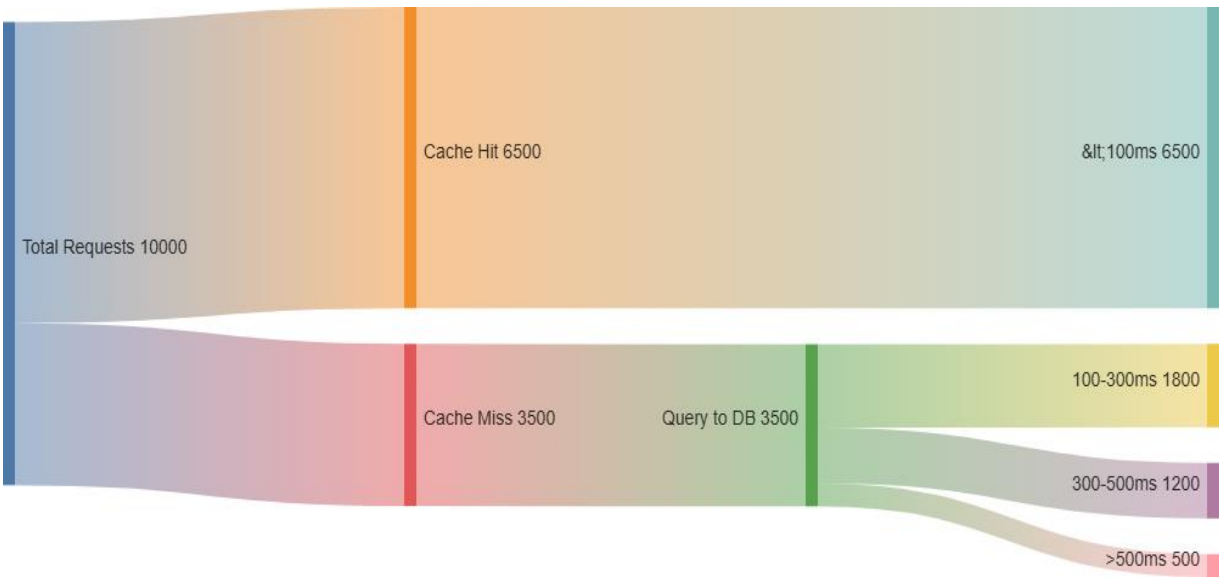

Figure 1: Query Response Time (ms) in High Volume Test

In most benchmarks, MongoDB and Cassandra showed better performance for unstructured and semi-structured data. Conversely, SQL engines struggled under horizontal loads and transactional bursts.

5. Consistency and Transaction Models

Relational databases are built on strong consistency mechanisms through the ACID properties, ensuring reliability even in multi-transactional environments. This makes them ideal for financial applications where data corruption cannot be tolerated. However, ACID compliance often leads to performance bottlenecks in high-volume systems where latency must be minimized.

In contrast, non-relational databases embrace the BASE model to optimize availability and partition tolerance. By allowing temporary inconsistencies, systems like Cassandra and DynamoDB achieve greater responsiveness and fault tolerance in distributed networks. The trade-off between immediate consistency and system uptime becomes a crucial decision in large-scale deployments.

6. Suitability in Industry Applications

Relational databases remain the backbone of sectors requiring rigorous regulatory compliance, such as finance, healthcare, and government services. Their support for complex querying, joins, and transactions suits environments where data relationships are critical.

Conversely, non-relational databases shine in web-scale applications like social networks, gaming platforms, IoT ecosystems, and content management systems. Their ability to rapidly adapt to changing data types without costly migrations is indispensable for businesses seeking agile and scalable solutions. Hybrid systems are now being explored to leverage the best of both paradigms in unified architectures.

Table 2: Application Use Case Alignment

Industry	Preferred Model	Reason
Banking	Relational	Requires ACID compliance
Social Networks	Non-Relational	High-speed, unstructured content
Healthcare	Relational	Critical data integrity
IoT/Telemetry	Non-Relational	Real-time processing & flexibility

7. Results and Evaluation

After comprehensive performance testing across different database models, it becomes evident that non-relational systems deliver higher write and read throughput in distributed, high-load environments. Yet, relational systems maintain superiority in transaction-heavy use cases requiring precise data control.

The evaluation also indicates that hybrid approaches, combining relational and non-relational models, are increasingly deployed in cloud-native applications to balance consistency, scalability, and complexity. Organizations must evaluate workload characteristics before selecting a data model to ensure optimal performance and reliability.

8. Conclusion and Future Scope

The ongoing evolution of data storage technologies suggests that no single model can address all future needs. Organizations must adopt flexible strategies that align database selection with business objectives, workload types, and technological trends.

Future research could focus on multi-model databases that integrate SQL and NoSQL features under one engine, serverless database technologies, and further exploration into AI-augmented database management. Moreover, tighter integration of databases with real-time analytics and machine learning pipelines will define the next decade of innovation in high-volume data environments.

References

[1] Jatana, N., et al. "A survey and comparison of relational and non-relational database." *International Journal of Soft Computing and Engineering*, 2012.

[2] Sheta, S.V. (2021). Artificial Intelligence Applications in Behavioral Analysis for Advancing User Experience Design. International Journal of Artificial Intelligence (ISCSITR-IJAI), 2(1), 1–16.

[3] Ordonez, C., et al. "Relational versus non-relational database systems." *ACM Workshop on Data Warehousing*, 2010.

[4] Mohamed, M. A., et al. "Relational vs. NoSQL databases: A survey." *International Journal of Computer Applications*, 2014.

[5] Sheta, S.V. (2019). The Role and Benefits of Version Control Systems in Collaborative Software Development. Journal of Population Therapeutics and Clinical Pharmacology, 26(3), 61–76. https://doi.org/10.53555/hxn1xq28

[6] Du Toit, P. "An Evaluation of Non-relational Database Management Systems." *CORE*, 2014.

[7] Györödi, C., et al. "A comparative study of relational and non-relational databases." *International Journal of Advanced Computer Science*, 2015.

[8] Sheta, S.V. (2021). Security Vulnerabilities in Cloud Environments. Webology, 18(6), 10043–10063.

[9] Salehnia, A. "Comparisons of relational databases with big data." *ASEE Zone 3 Conference*, 2015.

[10] Addal, W. H. "A Comparative Analysis of Relational and Non-Relational Databases." *Near East University*, 2019.

[11] Kolonko, K. "Performance comparison of the most popular database systems." *DiVA Portal*, 2018.

[12] Sheta, S.V. (2022). A Study on Blockchain Interoperability Protocols for Multi-Cloud Ecosystems. International Journal of Information Technology and Electrical Engineering, 11(1), 1–11. https://ssrn.com/abstract=5034149

[13] Kashyap, N. K., et al. "A review of leading databases." *i-Manager's Journal*, 2016.

[14] Castelle, M. "Relational and non-relational models in bureaucracy." *Computational Culture*, 2013.

[15] Gyorodi, R., et al. "Comparative analysis of MongoDB and MSSQL." *Semantic Scholar*, 2015.

[16] Sheta, S.V. (2022). An Overview of Object-Oriented Programming (OOP) and Its Impact on Software Design. Educational Administration: Theory and Practice, 28(4), 409–419.

[17] Skublewska-Paszkowska, M. "Performance of relational and non-relational DBs." *EDULEARN18 Proceedings*, 2018.

[18] Tamane, S. "Non-Relational Databases in Big Data." *ACM ICACCI*, 2016.

[19] Atzeni, P., et al. "Uniform access to non-relational systems." *Springer CAiSE*, 2012.

[20] Patel, T., et al. "Relational database vs NoSQL." *JMEST*, 2015.