MIND (Multimedia Artificial Intelligent Networking Database) Journal

MIND Journal | ISSN (p): 2528-0015 | ISSN (e): 2528-0902 | Vol. 8 | No. 2 | Halaman 217 - 229
DOI : https://doi.org/10.26760/mindjournal.v8i2.217-229 Desember 2023

Exploring the Effectiveness, Features, and Compatibility of MongoDB and MySQL: A Comprehensive Comparison of NoSQL and Relational Databases

CHELSEA MARCHELLE SOWANDI, IVANA LEONITA, STEVEN HARTANTO PANJI ARISAPUTRA, DAVID DAVID

Computer Science Department, School of Computer Science Bina Nusantara University

Email : chelsea.marchelle@binus.ac.id

Received 24 Juni 2023 **|** *Revised* 05 Oktober 2023 **|** *Accepted* 17 Oktober 2023

ABSTRAK

Di era digital ini, data menjadi aset yang sangat berharga bagi bisnis dan organisasi. Terdapat jenis database yang umum digunakan, yaitu Database Relasional dan Database NoSQL. Perbandingkan antara kedua jenis database tersebut dibahas melalui analisis perbedaan dan dampak praktisnya. Evaluasi terhadap kedua jenis database ini dilakukan untuk menentukan platform database terbaik yang sesuai dengan kebutuhan organisasi. Penelitian ini membahas perbandingan MongoDB dengan MySQL. Analisis, kelebihan, dan keterbatasan dari masing-masing jenis database ini, serta hal yang mempengaruhi pemilihan platform database dibahas pada penelitian ini. Dengan membandingkan fitur dan kasus penggunaan kedua database ini, diharapkan dapat memberikan wawasan berharga yang dapat membantu organisasi membuat keputusan yang tepat saat memilih platform database.

Kata kunci*: database, Database NoSQL, MongoDB, MySQL, platform database*

ABSTRACT

In today's digital age, data has become an essential asset for businesses and organizations. Databases are used to store, manage, and update information, providing reliable and organized data that can be accessed and analyzed as needed. There are several types of databases, including Relational Databases and NoSQL Databases. This paper compares the two types of databases, analyzing their differences and practical implications. We examine the aspects of both type of database to determine the best database platform for specific organizational needs. Specifically, we compare MongoDB, a popular NoSQL Database, with MySQL, a widely used Relational Database. Our analysis considers the strengths and limitations of each type of database, and how they impact the selection of a database platform. By comparing the features and use cases of these databases, we provide valuable insights that can help organizations make informed decisions when selecting a database platform.

Keywords*: databases, NoSQL Databases, MongoDB, MySQL, database platform*

1. INTRODUCTION

In today's digital age, data is an essential resource for businesses and organizations. The amount of data generated continues to grow at an unprecedented rate, making effective management and organization of this data increasingly important. Databases are critical tools for managing and storing this data, providing efficient and organized access to information that can be analyzed and used to make informed decisions.

Relational databases and NoSQL databases are two commonly used types of databases, each with its strengths and weaknesses. Relational databases have been the traditional choice for many organizations, providing a structured and organized approach to data storage. These databases use a tablebased approach to store data, with each table consisting of rows and columns. The relationship between these tables is defined by primary and foreign keys, ensuring that data is organized in a logical and structured manner. Relational databases are ideal for handling structured data, such as financial records, customer data, and inventory management.

On the other hand, NoSQL databases are designed to handle unstructured data, providing flexibility and scalability that is especially useful for handling large volumes of data. These databases are highly scalable, allowing organizations to store and process large amounts of data quickly and efficiently. NoSQL databases are ideal for handling unstructured data, such as social media feeds, sensor data, and multimedia content.

Despite their differences, both types of databases are used extensively across various industries, including e-commerce, healthcare, finance, and more. Organizations must carefully consider their specific needs when selecting a database platform, taking into account the type of data they need to store, scalability required, transaction features needed, security considerations, cost, and compatibility with existing systems.

Organizations that require high levels of data integrity often choose Relational Databases. These databases include advanced features such as referential integrity, which ensures that all data entered into the database conforms to predefined rules. Additionally, these databases provide ACID (Atomicity, Consistency, Isolation, and Durability) transactions, which ensure that all data is consistent and accurate, even in the event of a system failure. However, these features come at the cost of reduced scalability and flexibility.

NoSQL databases, on the other hand, are highly scalable and flexible but may lack some of the advanced features provided by Relational Databases. These databases come in various types, including Document Databases, Key-Value Databases, Graph Databases, and Column-Family Databases. Each type of NoSQL database has its unique features and use cases, making it important for organizations to choose the right database type for their specific needs.

In this paper, we aim to provide a comprehensive comparison of two database platform which is MongoDB as a NoSQL database and MySQL as Relational database, examining their practical implications and identifying the best database platform for specific organizational needs as these two database platform represent two different approaches to data storage and management. In section 2, we provide an overview of MySQL as a Relational Databases, including their history, features, and limitations. In section 3, we examine NoSQL databases which is MongoDB, including their various types, features, and use cases. In section 4, we compare the two types of databases, examining their practical implications and identifying the best database platform for specific organizational needs. Finally, in section 5, we provide a conclusion and recommendations for selecting a database platform based on the specific needs of an organization.

We hope that this paper will serve as a valuable resource for organizations seeking to select the best database platform for their specific needs. By examining the differences between Relational Databases with MySQL and NoSQL Databases with MongoDB, organizations can make informed decisions that will help them to manage and analyze their data effectively.

2. RESEARCH METHODOLOGY

3.1.Literature Review

Relational databases have been widely used in various industries for decades. But as big data has grown in importance, non-relational databases, also referred to as NoSQL databases, have become more well-liked because of their capacity to manage unstructured and semi-structured data. MongoDB and MySQL are the two NoSQL databases that are most frequently used. In order to compare the performance of MongoDB vs MySQL, there is now a lot of study in this area.

Based on a research Bolcea **(Boicea et al., 2012)** MongoDB outperforms Oracle databases that assessed the two databases' ability to handle unstructured data. The query performance of relational and non-relational databases was examined in a different study by Ceresnak and Kvet **(Čerešňák & Kvet, 2019)** which they demonstrated that in terms of query performance, the non-relational database MongoDB performed better than the relational database PostgreSQL.

Similar to this, Jose and Abraham **(Anju & Swathi, 2017)** carried out a study using MongoDB and MySQL to compare the effectiveness of NoSQL and relational databases. According to the findings, relational databases are better at handling structured data than NoSQL databases are at handling unstructured data. And according to D. D. B. et al.'s **(Damodaran B et al., 2016)** comparison of the two databases' performance, MongoDB surpassed MySQL in terms of query performance for unstructured data.

A study by Palanisamy and Vani **(Palanisamy & Suvithavani, 2020)**, they compared MySQL and MongoDB in order to assess the differences between RDBMS and NoSQL databases. According to the survey, MySQL is better suited to handle structured data, whereas MongoDB is better suited to handle unstructured and semi-structured data.

R and P **(Győrödi et al., 2022)** conducted a comparison between MongoDB and Oracle databases using classification algorithms. According to the findings, MongoDB performs better in handling unstructured data, while Oracle performs better in handling structured data. A similar explanation by Gupta et al. **(Gupta et al., 2018)** conducted a critical analysis and comparison of NoSQL databases and concluded that MongoDB is more suitable for handling unstructured and semi-structured data.

An explanation overview of SQL and NoSQL explained by Dave and Sharma (Sharma & Dave, 2012). According to Ilić et al. **(Ilic et al., 2021)** Microsoft SQL Server and Oracle databases were compared and discovered that both perform similarly in terms of query execution time. However, Oracle has better performance in handling complex queries.

According to Islam et al. **(Islam et al., 2017)**, who did a comparison analysis and review of SQL Server 2016, Oracle 12c, and MySQL 5.7 in scope for personal computers, the comparative study concluded that MySQL performs better in terms of cost-effectiveness, while SQL Server and Oracle perform better in terms of scalability and performance.

Ceresnak and Kvet **(Čerešňák & Kvet, 2019)** conducted another study comparing the query performance of relational and non-relational databases. The results showed that NoSQL databases have better performance in handling unstructured data, while relational databases have better performance in handling structured data.

Faraj et al. **(Faraj et al., 2013)** compared relational and nonrelational databases using Oracle and MongoDB. The study concluded that MongoDB handles unstructured data better, while Oracle handles structured data better.

Singh and Yadav **(Győrödi et al., 2022)** compared MongoDB and document-based MySQL for data management in big data applications. The study concluded that MongoDB handles unstructured data better, while document-based MySQL handles semi-structured data better.

Hamad and Al-Dhuraibi **(Mumtahana, 2022)** optimised transaction database design with MySQL and MongoDB and came up with a result that both databases have comparable query execution time. However, MongoDB handles unstructured data better.

Le et al. (Keshavarz, 2021a) discovered the performance of MySQL and MongoDB databases. The study compared the two databases' performance in terms of response time and throughput using four separate benchmarks. The results showed that MySQL performed better than MongoDB in terms of read-only workloads, while MongoDB performed better in write-heavy workloads.

In a similar study, Mohamed et al. **(Eyada et al., 2020)** had evaluated the performance of IoT data management using MongoDB and MySQL databases in different cloud environments. The research was studying which database can perform some cases to measure in terms of response time and output from a lot of different scenarios. The results showed that MongoDB outperformed MySQL in terms of both response time and throughput, especially in scenarios with high write loads.

A conversion of data from a relational MySQL, had been presented by Nugraha and Winarno **(Susanti et al., 1994)** to a NoSQL MongoDB database for a Small and Medium Enterprises Information System. These studies compare the performance of the two databases in terms of data insertion and retrieval. The results showed that MongoDB outperformed MySQL in terms of data insertion, while MySQL performed better in data retrieval.

Overall, the studies show that the workload and context of a task have an impact on the performance of MySQL and MongoDB databases. MongoDB beats MySQL in write-heavy workloads , while MySQL performs better in read-heavy workloads. The choice of database ultimately depends on the specific requirements of the application. However, for big data applications and real-time processing, NoSQL databases like MongoDB are better suited due to their scalability and performance capabilities.

3.2 Objective

The goal of this study is to conduct a comprehensive comparison of MongoDB and MySQL, focusing on their various merits as relational and NoSQL databases in terms of effectiveness, features, and compatibility. The study aims to provide insights into their performance characteristics, data management capabilities, and overall suitability for different application scenarios.

3.2. Study Design

This research follows a secondary research method, utilizing existing literature and observations from relevant sources. A systematic approach is employed to gather, analyze, and synthesize information from selected articles, conference papers, and scholarly publications.

3.3. Data Sources and Selection Criteria

Databases such as IEEE Xplore, ScienceDirect, ACM Digital Library, and Google Scholar are used to conduct a literature search. For the literature review, the following selection criteria were used:

- Relevance: Articles emphasizing the effectiveness, functionality, and compability of MongoDB and MySQL in comparison to each other **(Gupta et al., 2018) (Mumtahana, 2022) (Damodaran B et al., 2016) (Eyada et al., 2020) (Jose & Abraham, 2019) (Palanisamy & Suvithavani, 2020) (Anju & Swathi, 2017) (Keshavarz, 2021b) (Győrödi et al., 2022) (Sharma & Dave, 2012) (Keshavarz, 2021a) (Faraj et al., 2013) (Čerešňák & Kvet, 2019) (Islam et al., 2017) (Ilic et al., 2021);(Susanti et al., 1994)(Boicea et al., 2012).**
- Recency: works published over the past ten years to reflect the most recent developments and updates **(Gupta et al., 2018)(Mumtahana, 2022) (Damodaran B et al., 2016) (Eyada et al., 2020) (Jose & Abraham, 2019) (Palanisamy & Suvithavani, 2020) (Anju & Swathi, 2017) (Keshavarz, 2021b) (Győrödi et al., 2022) (Sharma & Dave, 2012) (Keshavarz, 2021a) (Faraj et al., 2013) (Čerešňák & Kvet, 2019) (Islam et al., 2017) (Ilic et al., 2021)(Susanti et al., 1994)(Boicea et al., 2012).**
- Credibility: Sources with reputable academic to ensure the reliability of the information **(Gupta et al., 2018)(Mumtahana, 2022)(Damodaran B et al., 2016) (Eyada et al., 2020)(Jose & Abraham, 2019) (Palanisamy & Suvithavani, 2020) (Anju & Swathi, 2017) (Keshavarz, 2021b) (Győrödi et al., 2022) (Sharma & Dave, 2012) (Keshavarz, 2021a) (Faraj et al., 2013) (Čerešňák & Kvet, 2019) (Islam et al., 2017) (Ilic et al., 2021) (Susanti et al., 1994) (Boicea et al., 2012)**.
- Performance Metrics: Articles providing specific performance metrics and benchmarks for MongoDB and MySQL **(Eyada et al., 2020)(Anju & Swathi, 2017)(Keshavarz, 2021b) (Győrödi et al., 2022)(Sharma & Dave, 2012)(Keshavarz, 2021a) (Čerešňák & Kvet, 2019) (Islam et al., 2017)(Susanti et al., 1994)(Boicea et al., 2012)**.

3.4. Data Collection and Extraction

The identified articles are carefully reviewed to extract relevant data and observations. Key aspects considered during data collection include:

- Effectiveness: Comparative analysis of query performance, scalability, and efficiency of MongoDB and MySQL **(Damodaran B et al., 2016) (Eyada et al., 2020) (Jose & Abraham, 2019) (Anju & Swathi, 2017) (Keshavarz, 2021b) (Győrödi et al., 2022) (Sharma & Dave, 2012)(Keshavarz, 2021a) (Faraj et al., 2013) (Čerešňák & Kvet, 2019) (Ilic et al., 2021) (Boicea et al., 2012)**.
- Features: Evaluation of data modeling capabilities, transaction support, indexing mechanisms, and other relevant features **(Jose & Abraham, 2019)(Anju & Swathi,**

2017) (Sharma & Dave, 2012) (Islam et al., 2017) (Faraj et al., 2013) (Győrödi et al., 2022).

- Compatibility: Analysis of ecosystem integration, compability with current frameworks, and tools **(Palanisamy & Suvithavani, 2020) (Gupta et al., 2018) (Islam et al., 2017) (Faraj et al., 2013)(Mumtahana, 2022).**

3.5. Data Analysis

The collected data is qualitatively analyzed to identify the similarities, differences, strengths, and limitations of MongoDB and MySQL. To allow an extensive comparison, the research results from the examined literature are summarized and presented in a systemic way.

3.6. Limitations

This research methodology has inherent limitations:

- Reliance on secondary data: The study is based on previously published research, which could bring biases and limitations from the original studies.
- Methodological variability: The evaluated literature may use different methodologies and experimental setups, which might cause findings to vary.
- Scope limitation: The analysis is limited to the factors that are specifically discussed in the selected literature.

3.7. Future Research

Based on the findings and limitations, suggestions for future research areas are provided. These may include:

- Conducting primary research studies to validate and supplement the findings of the secondary research.
- Considering more relational and NoSQL databases for comparison in order to conduct a more thorough analysis.
- More thorough investigation of particular performance factors, such as query optimization or data processing capabilities.

By employing this research methodology, we hope to provide a thorough and informative comparison of MongoDB and MySQL, providing valuable guidance for researchers, practitioners, and organizations in selecting the most suitable database solution for their specific requirements.

3. RESULT AND DISSCUSSION

This section summarizes the experimental research done to evaluate the query analysis and performance of MongoDB, and MySQL databases. The experiment attempts to evaluate these database systems' strengths and weaknesses, offering information on their applicability for various application scenarios. A number of cited research, including conference papers, journal articles, and surveys, released between 2019 and 2021, had an impact on the experimental design and analysis.

The chosen references provide insightful information about the functionality of MongoDB and MySQL databases in terms of performance, query handling, and data management. These database systems are assessed using a variety of experimental approaches and performance indicators in diverse scenarios. By adding more performance and query analysis parameters, this experimental study enhances their research and broadens its focus.

The experiment is intended to present a comprehensive analysis of the contrasted databases, taking into account both their strengths and weaknesses. It includes data management evaluations, query execution testing, and performance measures. We hope to add to the body of information about the performance traits and query capabilities of MongoDB, Oracle, and MySQL databases by undertaking a careful and scientific experiment.

This section presents the performance analysis of MySQL and MongoDB databases based on the comparative study titled "Comparison of query performance in relational and non-relational databases" by Ceresnak and Kvet **(Čerešňák & Kvet, 2019)**, a comprehensive investigation into the query performance of relational and non-relational databases is presented.

The researchers performed tests on four types of queries: Insert, Update, Delete, and Select. These queries were executed on each type of database, using a dataset consisting of 10,000 records. The measured time for query execution was recorded in milliseconds. Which the result is written in Table 1.

Table 1. Query Performance of Databases With 10 000 Records in Milliseconds

Type	Oracle	MySql	MsSql	Mongo	Redis	Graph	Cassandra
Insert	0,076	0,093	0,093	0,005	0,009	0,008	0,011
Update	0,077	0,058	0,073	0,008	0,013	0,007	0,013
Delete	0,059	0,025	0,093	0,01	0,021	0,017	0,018
Select	0,025	0,093	0,062	0,009	0,016	0,010	0,014

The purpose of these tests was to compare the performance of different types of databases and evaluate their efficiency in handling various types of queries. By examining the execution times for each query type across the different databases, the researchers aimed to identify the strengths and weaknesses of each database system in terms of query performance.

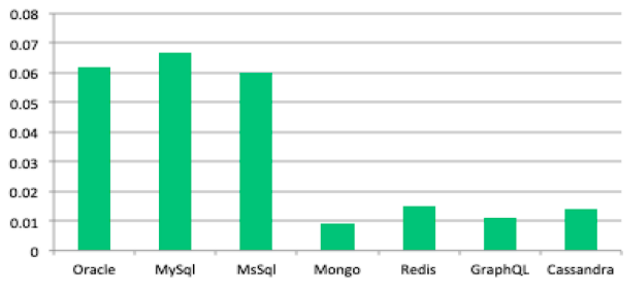

Figure 1. Query Performance in Miliseconds

According to Table 1 and Figure 1, the results for the select query times in relational databases (Oracle, MySQL, and MS SQL) and non-relational databases (MongoDB, Redis, Cassandra, and GraphQL) are shown. The results indicate that the performance differences between these types of databases are as expected.

In NoSQL databases such as Redis, MongoDB, Cassandra, and GraphQL, data is stored in the form of flat collections. This means that the data is duplicated, and a single piece of data is rarely partitioned off but stored as an entity. As a result, reading or writing operations to a single entity are more accessible and faster.

On the other hand, in relational databases, data needs to be broken into several small logical tables to avoid duplicity and redundancy. This is achieved through normalization, which helps manage data correctly and efficiently. However, dividing the data into related tables as part of the normalization process can hinder the performance of data processing in relational databases using SQL. Consequently, the time needed to retrieve values from relational databases is higher compared to non-relational databases, as depicted in figure 2.

The reference **(Celesti et al., 2018)** by Celesti, M. Fazio, A. Romano, A. Bramanti, P. Bramanti, and M. Villari titled "An OAIS-Based Hospital Information System on the Cloud: Analysis of a NoSQL Column-Oriented Approach" provides further insights into the topic The comparison of query performance between the relational database Oracle and the non-relational database MongoDB shows that the ratio for data retrieval is almost 1:3 for select performance, 1:6 for the delete operation, 1:9 for the update operation, and 1:15 for the insert operation. This implies that non-relational databases generally outperform relational databases in terms of query performance for these operations.

It's important to note that these performance ratios may vary depending on the specific use case, database configuration, and dataset characteristics. Therefore, it's crucial to evaluate the trade-offs and suitability of different database types based on the specific requirements and performance needs of the application or system being developed.

In the comparative study conducted by Faraj, Rashid, et al. **(Győrödi et al., 2022)**, the performance analysis of Oracle and MongoDB databases is presented. They examined various aspects, including select, count, sum, and average operations, as shown in Figures 1 to 4. Another study by Ceresnak and Kvet **(Čerešňák & Kvet, 2019)** investigated the query performance of relational and non-relational databases, and Figure 5 illustrates their findings.

The paper titled "Optimization of Transaction Database Design with MySQL and MongoDB" by Hamad and Al-Dhuraibi **(Mumtahana, 2022)** focuses on the performance analysis of transaction processing in two popular database systems, namely MySQL and MongoDB. The study aims to investigate and optimize the transactional database design to enhance the performance of these systems such as covering insert, update, delete, and search operations, as represented by Figures 6 to 9 below.

Table 2. MySQL and MongoDB Performance Test Result

Type	**MySQL**				**Mongo DB**			
Data	**25**	**50**	**75**	**100**	**25**	**50**	**75**	**100**
Insert	0.045	0,080	0,096	0,112	0,005	0.013	0,0027	0,045
Update	0.015	0,037	0,059	0,076	0,002	0.005	0,008	0,014
Delete	0.013	0,025	0,047	0,077	0,003	0.005	0,007	0,009
Select	0.035	0,055	0,077	0,096	0,012	0.022	0,045	0,063

Table 2 provides a comparison of the run time differences between MySQL and MongoDB for the insert, update, delete, and search processes. The tests were conducted using varying numbers of data, including multiples of the initial 25 data, 50 data, 75 data, and 100 data. These tests aimed to evaluate the performance of both MySQL and MongoDB in terms of their execution time for these essential database operations. For a visual representation of the test results and a more detailed analysis, the comparison of test results is described in the figure 2, Figure 3, Figure 4, and Figure 5.

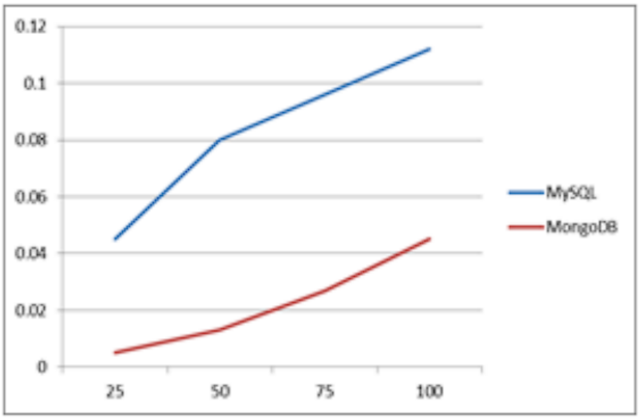

Figure 2. Comparison of Insert

In Figure 2, the paper is comparing the performance of MySQL and MongoDB for the insert test data operation.

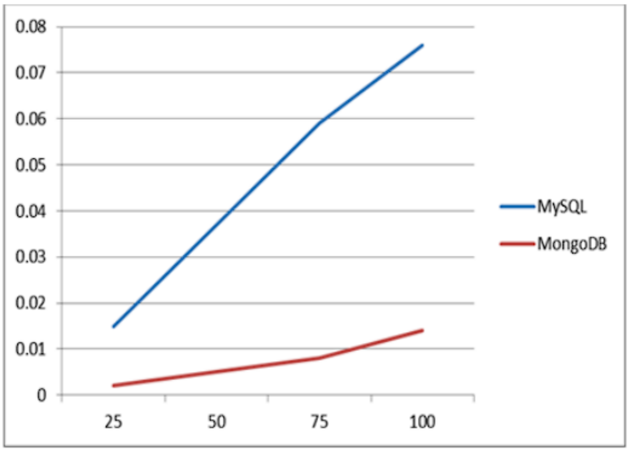

Figure 3: Comparison of Update

Figure 3 presents the results of a comparison between MySQL and MongoDB for the data update command.

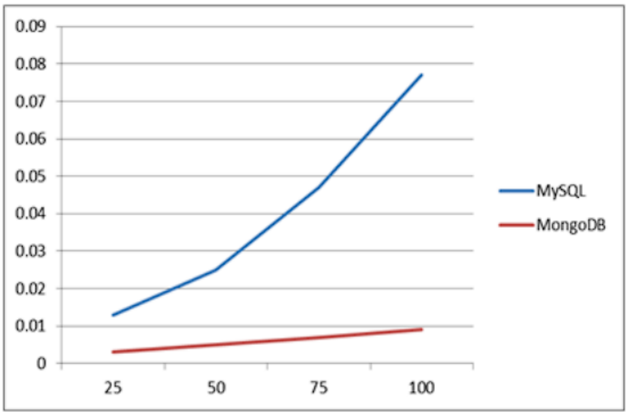

Figure 4. Comparison of Delete

Figure 4 displays the results of a comparison between MySQL and MongoDB for the delete command.

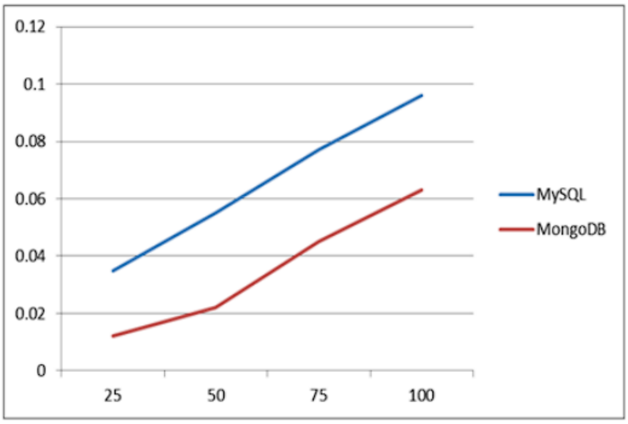

Figure 5. Comparison of Search

Lastly, Figure 5 illustrates the comparison of test results for the search data command. It covers three scenarios: simple data search, data search with JOIN (likely involving relational database features), and data search involving transactions using aggregation functions.

The test results presented in the paper "Optimization of Transaction Database Design with MySQL and MongoDB" by Hamad and Al-Dhuraibi **(Mumtahana, 2022)** indicate that MongoDB exhibits superior performance compared to MySQL. Specifically, in the context of the insert command, MongoDB outperforms MySQL with a time difference of 0.005625 seconds. Similarly, for the update command, MongoDB demonstrates a time difference of 0.001688 seconds, while the delete command exhibits a time difference of 0.00075 seconds. In terms of the search operation, MongoDB showcases a time difference of 0.006875 seconds compared to MySQL.

In the field of database management systems, several journals have contributed to the understanding of NoSQL databases, with a particular focus on MongoDB, through performance analysis, comparative studies, and critical assessments.

For instance, a study by B. Jose and S. Abraham **(Jose & Abraham, 2019)** compared the performance of MongoDB and MySQL, highlighting MongoDB's better response time and scalability. A critical analysis and comparison of various NoSQL databases, including MongoDB, was conducted by A. Gypta et al. **(Gupta et al., 2018)**, emphasizing the advantages of NoSQL databases in terms of horizontal scalability, flexible data models, and high performance. P. R. Singh and R. K. Yadav **(Győrödi et al., 2022)** compared MongoDB and document-based MySQL for big data application data management, concluding that MongoDB outperformed document-based MySQL in terms of performance and scalability.

Other studies, such as the one by Le et al. **(Keshavarz, 2021b)**, demonstrated that MongoDB exhibited faster query execution time and lower memory consumption than MySQL. Mohamed et al. **(Eyada et al., 2020)** evaluated the performance of MongoDB and MySQL for IoT data management in different cloud environments, finding that MongoDB provided better scalability and response time for IoT data management tasks. Celesti et al. **(Celesti et al., 2018)** discussed the use of MongoDB's column-oriented approach in an OAIS-based hospital information system, emphasizing its benefits in managing complex and heterogeneous healthcare data..

4. CONCLUSION

In conclusion, the experiments conducted in these studies highlight the strengths and limitations of both NoSQL and relational databases. MongoDB, as a representative of NoSQL databases, demonstrated superior performance in terms of data insertion, update, delete, and search operations compared to MySQL. Its ability to store data in the form of flat collections allows for faster access and processing of individual entities. On the other hand, relational databases such as Oracle, MySQL, and MS SQL provide the advantage of data normalization, which ensures data integrity and efficient management through the division of data into logical tables.

Organizations that prioritize scalability, flexibility, and high-speed data access may find NoSQL databases like MongoDB, Redis, Cassandra, or GraphQL suitable for their needs. These databases excel in scenarios where large volumes of data need to be handled, and data structures are subject to frequent changes. They are well-suited for applications that require quick and efficient read and write operations, such as real-time analytics, content management systems, and high-traffic websites.
On the other hand, organizations that require strict data consistency, complex relationships, and advanced query capabilities may benefit from relational databases. Oracle, MySQL, and MS SQL offer powerful data management capabilities, ensuring data integrity through normalization and providing robust transaction support. These databases are commonly used in applications that involve financial systems, e-commerce platforms, and enterprise resource planning (ERP) systems.

However, it is important to note that the choice of database should be driven by specific requirements and considerations. Factors such as data structure, query patterns, scalability needs, data integrity requirements, and development resources should be taken into account. Additionally, organizations should consider the potential trade-offs between performance and data consistency when selecting between NoSQL and relational databases.

In summary, while NoSQL databases offer advantages in terms of scalability and speed, relational databases provide strong data integrity and advanced query capabilities. Organizations should carefully evaluate their requirements and align them with the strengths of each database type to make an informed decision that best suits their specific application needs.

REFERENCES

Anju, R., & Swathi, B. P. (2017). MongoDB vs. Oracle: A comparative study using classification algorithms. *2016 IEEE International Conference on Computational Intelligence and Computing Research, ICCIC 2016*, 2–5. https://doi.org/10.1109/ICCIC.2016.7919628

Boicea, A., Radulescu, F., & Agapin, L. I. (2012). MongoDB vs Oracle - Database comparison. *Proceedings - 3rd International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2012*, *August*, 330–335. https://doi.org/10.1109/EIDWT.2012.32

Celesti, A., Fazio, M., Romano, A., Bramanti, A., Bramanti, P., & Villari, M. (2018). An OAIS-based hospital information system on the cloud: Analysis of a NoSQL column-oriented approach. *IEEE Journal of Biomedical and Health Informatics*, *22*(3), 1–7.

https://doi.org/10.1109/JBHI.2017.2681126

Čerešňák, R., & Kvet, M. (2019). Comparison of query performance in relational a non-relation databases. *Transportation Research Procedia*, *40*, 170–177. https://doi.org/10.1016/j.trpro.2019.07.027

Damodaran B, D., Salim, S., & Vargese, S. M. (2016). Performance Evaluation of MySQL and MongoDB Databases. *International Journal on Cybernetics & Informatics*, *5*(2), 387–394. https://doi.org/10.5121/ijci.2016.5241

Eyada, M. M., Saber, W., El Genidy, M. M., & Amer, F. (2020). Performance Evaluation of IoT Data Management Using MongoDB Versus MySQL Databases in Different Cloud Environments. *IEEE Access*, *8*, 110656–110668. https://doi.org/10.1109/ACCESS.2020.3002164

Faraj, A., Rashid, B., & Shareef, T. (2013). Comparative Study of Relational and Non-Relational Database Performances Using Oracle and MongoDB Systems. *International Journal of Computer Engineering and Technology (IJCET)*, *June 2016*, 323–329.

Gupta, A., Tyagi, S., Panwar, N., Sachdeva, S., & Saxena, U. (2018). NoSQL databases: Critical analysis and comparison. *2017 International Conference on Computing and Communication Technologies for Smart Nation, IC3TSN 2017*, *2017-Octob*(October 2017), 293–299. https://doi.org/10.1109/IC3TSN.2017.8284494

Győrödi, C. A., Dumşe-Burescu, D. V., Zmaranda, D. R., & Győrödi, R. (2022). A Comparative Study of MongoDB and Document-Based MySQL for Big Data Application Data Management. *Big Data and Cognitive Computing*, *6*(2). https://doi.org/10.3390/bdcc6020049

Ilic, M., Lazar Kopanja, Zlatkovic, D., Trajkovic, M., & Curguz, D. (2021). Microsoft SQL Server and Oracle : Comparative Performance Analysis. *The 7th International Conference Knowledge Management and Informatics Vrnjačka*, *1*(June), 33–40.

Islam, K., Ahsan, K., Bari, S. A. K., Saeed, M., & Ali, S. A. (2017). Huge and Real-Time Database Systems: A Comparative Study and Review for SQL Server 2016, Oracle 12c & MySQL 5.7 for Personal Computer. *Journal of Basic & Applied Sciences*, *13*(21), 481–490.

Jose, B., & Abraham, S. (2019). Performance analysis of NoSQL and relational databases with MongoDB and MySQL. *Materials Today: Proceedings*, *24*, 2036–2043. https://doi.org/10.1016/j.matpr.2020.03.634

Keshavarz, S. (2021a). *Analyzing Performance Differences Between MySQL and MongoDB*. *March*, 46. https://www.researchgate.net/publication/349764039_Analyzing_Performance_Differen ces_Between_MySQL_and_MongoDB

Keshavarz, S. (2021b). *Analyzing Performance Differences Between MySQL and MongoDB* (Issue March). https://www.researchgate.net/publication/349764039_Analyzing_Performance_Differences_Between_MySQL_and_MongoDB

Mumtahana, H. A. (2022). Optimization of Transaction Database Design with MySQL and MongoDB. *SinkrOn*, *7*(3), 883–890. https://doi.org/10.33395/sinkron.v7i3.11528

Palanisamy, S., & Suvithavani, P. (2020). A survey on RDBMS and NoSQL Databases MySQL vs MongoDB. *2020 International Conference on Computer Communication and Informatics, ICCCI 2020*. https://doi.org/10.1109/ICCCI48352.2020.9104047

Sharma, V., & Dave, M. (2012). SQL and NoSQL Databases. *International Journal of Advanced Research in Computer Science and Software Engineering Research*, *II*(8), 19–27.

Susanti, E., Sholeh, M., Yuliana, R., & Rachmawati. (1994). Konversi Data dari Database Relational MySQL ke Database NoSQL MongoDB (Studi Kasus pada Sistem Informasi UMKM). *Jurnal Gaung Informatika*, *135*(4).