



## A Comparative Analysis of MongoDB and NOSQL

*Ruchir Tanwar<sup>1</sup>, Dr. Vishal Shrivastava<sup>2</sup>, Dr. Akhil Pandey<sup>3</sup>, Prof. Mohit Mishra<sup>4</sup>*

<sup>1</sup>B.TECH. Scholar, <sup>2,3</sup>Professor, <sup>4</sup>Assistant Professor

Computer Science & Engineering, Arya College of Engineering & I.T. India, Jaipur

[tanwarrt114123@gmail.com](mailto:tanwarrt114123@gmail.com), [vishalshrivastava.cs@aryacollege.in](mailto:vishalshrivastava.cs@aryacollege.in), [akhil@aryacollege.in](mailto:akhil@aryacollege.in), [mohitmishra.cs@aryacollege.in](mailto:mohitmishra.cs@aryacollege.in)

### Introduction

Advanced data management has seen a rise in the use of No SQL databases. To aid in the choice you make, this demo compares MongoDB with various No SQL databases. We'll discuss the positive aspects and drawbacks of each choice. This is a No SQL database that uses JSON format to collect unstructured data. No SQL databases are relational, which means that they store and retrieve data without the usage of tabular structures. Big data can be efficiently stored with MongoDB thanks to its design. offers outstanding efficiency, availability, and scalability. The following is how MongoDB and SQL databases differ from one another: Data storage: Unstructured data is stored in MongoDB, whereas structured data is stored in SQL databases. Query: While SQL-based databases employ SQL query processing, MongoDB allows JSON queries. Structure: The table structure of a relational database is not the foundation of MongoDB. Scalability: Compared to relational database systems, No SQL databases are more scalable. The following are some of MongoDB's drawbacks: It requires a lot of RAM. The maximum file size is 16 MB. The maximum number of slots is 100 levels. There is a maximum of 20,000 connections allowed.

### MongoDB

MongoDB is an open No SQL database that is driven by different information. Big files are stored on it. Unlike relational databases, which utilize tables and rows, MongoDB uses lists and records. Key pairs is the basic building blocks of multiple data in it . Because MongoDB lacks a schema, it is capable of managing data without one. High performance is also provided by it, and it is scalable. Eliot Horowitz, Kevin Ryan, and Dwight Merriment started MongoDB in 2007. The group created and implemented multiple unique storefronts at Double Click, an online advertising company owned by Google. MongoDB Atlas is a cloud database solution available globally for contemporary applications.

### Capabilities of MongoDB:

**Based on data:** MongoDB saves important information in the least amount of data possible, as opposed to RDBMS, which enters into several relationships. For instance, all computer information is stored in a file named "Computer" better than in several relational structures like CPU, RAM, and hard drive. **Index:** In order to obtain the index, the database needs to duplicate every record in the collection. It won't assist to select those that match the question. As a outcome of it, indexing is essential for efficient searching, and it leverages this to quickly analyze massive amounts of information. **Scalability:** MongoDB scales horizontally by using sharding, which distributes data among several servers. Shared keys are used to split data into shards, which are akin to shards spread over several physical servers. Furthermore, additional systems may be added to the storage feature. **Replication and availability:** Data from several copies on several servers is arranged using Mon go DB. It provides redundancy, which shields data from hard drive failure. Data can be readily accessed from other servers that are still keeping data in the event that a server goes down. **Aggregation:** Gathering data and providing the computed outcomes is the process of aggregation. It is similar to the GR MySQL's OUPBY clause. Common phrases include min, max, average, total, and so forth. Where is MongoDB used?

RDBMS is inferior to MongoDB in the following scenarios:

**Big database:** Prioritize MongoDB over RDBMS databases if you have a lot of data saved in tables. Solutions for partitioning and data partitioning are offered by MongoDB. An unstable schema Whereas MongoDB lacks a schema, DBMS make it difficult to add additional columns. It is quite simple and has no effect on the previous DAT to add new fields. Since distributed data is saved in several copies across various servers, data recovery is quick and secure even in the event of hardware issues.

## NO SQL

It is called No SQL is intended to handle and keep a large amount of redundant data and semi-transactions. No SQL databases employ flexible databases that may be altered for data revisions and changes, in difference to traditional relational databases, which use tables with preset structures to store information. horizontal scaling done voluntarily to manage data accumulation. No SQL originally stood for "non-SQL" or "non relational" databases, but as No SQL databases grew to encompass a wide range of databases and data structures, the term eventually came to signify "non-SQL."

There are four general categories into which No SQL databases fall:

**Databases of documents:** These databases allow for database queries and store data in semi-structured formats like JSON or XML.

**Key Store:** This file, which is designed for read/write speed and user-friendliness, stores data as key values. **Column family storage:** A collection of rows handled as a single entity, a column family is how data is stored in this database. It is designed to efficiently and rapidly query massive volumes of data.

**Graph databases:** These databases are made to maintain relationships between data and store information as nodes and edges. No SQL databases are frequently utilized in applications like social networking, e-commerce, and gaming that need instantaneous processing and analysis of massive volumes of data. In addition, they can be applied to other tasks including customer, data, and content management. No SQL databases might not offer the same degree of data consistency and transaction guarantee as relational databases, hence they might not be appropriate for all applications. A database management system selection requires thorough consideration of your application's particular needs. Big data and real-time web applications both employ No SQL databases, and this usage is only growing. Because No SQL systems can enable SQL like queries, they are frequently referred to as Not SQL systems. No SQL databases offer more control, easier horizontal scaling across clusters of servers, and a more streamlined design. Certain operations in No SQL databases are faster because they use different data types than relational databases do by default. A No SQL database's applicability is determined by the issue it seeks to resolve. No SQL database, also known as "not just SQL" database, is a new database management system that has emerged in recent years. Unlike traditional relational databases, No SQL databases are designed to handle large or semi-structured data and can be modified to accommodate changes in the data structure. This makes No SQL databases ideal for modern web applications.

The data used by No SQL databases is sometimes considered to be more flexible than database contracts. Many No SQL vendors sacrifice consistency for availability, speed, and distributed integrity. Obstacles to widespread use of No SQL storage include the use of low-level queries, shortage of connection standards, over investment in existing database connections. The majority of No SQL stores as ACID (Atomicity, Consistency, Isolation, and Durability) operations; nevertheless, several databases, like Google Spanner (which is actually a New SQL database), Mark Logic, Aerospace, Fair Com c-tree Ace, Sassy LMD B, and Orient Db, have been employed. as the foundation for their production. Database queries do not always return fresh information right away or may result in stale reads, or incorrect data readings. This is because the majority of No SQL databases provides a idea of correspondence in which information updates are transmitted to all nodes. No SQL systems may also experience write loss and other data losses.

Re-write capabilities, for example, are provided by some No SQL systems to avoid data loss. Database is a basic illustration of a No SQL database.

The majority of No SQL stores lack full ACID (atomicity, consistency, isolation, and durability) operations; nevertheless, several databases, like Google Spanner (which is actually a New SQL database), Mark Logic, Aerospace, Fair Com c-tree Ace, Sassy LMD B, and Orient Db, have been employed. as the foundation for their production.

Database queries do not always return fresh data right away or may result in stale reads, or incorrect data readings. This is because the majority of No SQL databases offer a concept of consistency in which data updates are transmitted to all nodes. Certain No SQL systems may also experience write loss and other data losses. Re-write capabilities, for example, are provided by some No SQL systems to avoid data loss.

Database is a basic illustration of a No SQL database. They are made to withstand changes to the data structure and big amounts of unstructured or semi-structured data. Because of this, No SQL databases are perfect for big data processing, real-time analytic, and contemporary web applications. **Key Characteristics of No SQL Dynamic Schema:** No SQL databases are able to update data changes without requiring any alterations or modifications because they lack a set schema.

**Horizontal scalability:** No SQL databases are ideal for handling massive data sets and heavy traffic since they are designed to extend horizontally by adding nodes to the database cluster. **Data-based:** MongoDB and other No SQL databases employ database models, in which data is kept in a format like JSON or BSON.

**Key-value based:** Redis and other No SQL databases employ a key-value model in which information is kept as a set of key values.

**Column-based:** A column-based data model is used by some No SQL databases, including Cassandra, in which data is arranged in columns as opposed to rows.

**Availability and deployment:** No SQL databases are typically made to be very available, able to manage node loss, and capable of replicating data among several nodes in the database cluster.

**Flexibility:** No SQL databases enable developers to collect and keep information in a dynamic manner by supporting a variety of data kinds and evolving data models.

Performance: No SQL databases are ideal for big data and time-consuming applications because they are designed for high performance and can manage a lot of reads and writes.

Benefits of No-SQL There are many advantages of using No-SQL databases such as MongoDB and Cassandra. Its main advantages are high scalability and versatility.

High scalability: No-SQL databases use sharing for horizontal

scalability. Sharing information and placing it in multiple systems in a way that preserves the order of the information is called sharing. Vertical scaling means adding more resources to existing machines, while horizontal scaling means adding more machines to process data. Vertical scaling is not easy to implement but horizontal scaling is. Examples of horizontally scalable databases include MongoDB, Cassandra, and others. Because of scalability, No SQL can handle large data and as the data grows, No SQL automatically scales the data to be efficient. Flexibility: No-SQL databases are designed to manage unstructured or semi-structured data; This means they can adapt to changes in the data structure. This makes No-SQL databases ideal for applications that need to manipulate data on demand. High Availability: A No-SQL database's automated replication capability makes it extremely available as, in the event of a failure, the data replicates itself back in time.

Scalability: No-SQL databases can readily handle massive volumes of data and traffic because of their excellent scalability. They are therefore perfect for applications that have a lot of data or traffic to handle. Performance: Compared to traditional relational databases, No-SQL databases can offer greater performance because they are built to manage massive volumes of data and traffic. Because they are simpler and don't require pricey gear or software, they are typically more affordable. Quickness: Perfect for quick development.

Negative aspects of No-SQL: No-SQL has the following negative aspects.

No pattern: No-SQL databases come in a variety of forms, each with pros and cons of their own. Selecting the appropriate materials for a given application can be challenging in the absence of design.

Lack of full ACID compliance: No-SQL databases are unable to ensure the consistency, integrity, and longevity of the data since they lack full ACID compliance. For applications that need high data consistency assurances, this could be a disadvantage.

Narrow Focus: Although No-SQL databases offer many capabilities, their primary purpose is storage, which gives them a very narrow focus. Relational databases are a preferable option than No-SQL databases in business management.

Open Source: A database that uses No-SQL is open source. For No-SQL, there aren't any trustworthy standards available yet. That is, it could appear that the two database systems are incompatible. Lack of support for sophisticated queries: No-SQL databases are not made to manage sophisticated queries, hence they are not appropriate for applications that need sophisticated data or searches. to display.

Lack of maturity: Compared to conventional relational databases, No SQL databases are more recent and less developed. As a result, they will be less dependable and safe than conventional storage facilities.

Challenges in management: large data tools for the easiest possible large data management. Still, it's not that simple. Compared to relational databases, No-SQL data management is more complicated. In particular, No SQL is known to be challenging to set up and very difficult to maintain.

GUI not available: There are no instruments in the shop to modify the GUI mode used to access information.

Backups: One major flaw in several No-SQL databases, such as MongoDB, is their backup system. There is no viable system that allows MongoDB to backup its data.

Big data: JSON is the format used by several database systems, like Couch Db and MongoDB, to store data. This indicates that there is a lot of data (size, bandwidth, and speed of the network) and that critical descriptive labels might get destroyed as the data volume increases. Different No-SQL database types: The following is a list of database systems that fall under the category of No SQL databases, along with their types:

Databases on graphs: Examples are Neo4j and Amazon Neptune.

Stores of key values: Memcached, Redis, and consistency are a few examples. Column : Abase, Big Table, and Accumulator are a few examples. File-based databases include MongoDB, Couch DB, and Colourant.

### **Examples**

MongoDB and Other No SQL Databases: A Comparative Analysis Compared to other No-SQL databases, MongoDB There are also additional No-SQL databases available, including Couch base, Redis, and Cassandra. Every database has benefits and drawbacks, and the optimal option will rely on the particular requirements of your application. Applications requiring adaptable and high-performing data structures should consider MongoDB. Numerous additional No-SQL repositories are accessible, such as Couch base, Redis, and Cassandra. Every database has benefits and downsides of it, and it will rely on the individual requirements of your application. Applications requiring adaptable and fast data structures should consider MongoDB.

Choosing between No-SQL and MongoDB

Take into consideration elements like data structure, scalability, performance, interoperability, and durability while selecting MongoDB and other No SQL databases. It's crucial to take into account each repository's information availability and degree of community support. In the end, the optimal option is determined by the particular requirements of your application. Consider features like data structure, scalability, performance, consistency, and durability while selecting MongoDB and other No SQL databases. The amount of community support and documentation offered by each database should also be taken into account. In the end, the optimal decision depends on the particular requirements of your application.

---

## Conclusion

This is a very specific and direct conclusion to your MongoDB vs No-SQL comparison.

In our comparison of MongoDB and No-SQL, we found that MongoDB is a popular and powerful No-SQL database with many advantages such as flexible data structures, high performance and ease of use. However, it also has some disadvantages such as limited transaction support and data loss. When choosing MongoDB and other No SQL databases, consider the specific needs of your application. It is a popular and powerful No-SQL database with many advantages such as flexible data structure, high performance and ease of use. However, it also has some disadvantages such as limited transaction support and data loss. When choosing MongoDB and other No-SQL databases, consider the specific needs of your application and carefully weigh the pros and cons.

## References:

---

1. AngularJS "Angular JS" , Guide [online] available :<https://docs.angularjs.org/guide>
- 2.ReactJS "React JS" Guide[online] Available:<https://facebook.github.io/react/docs/thinking-inreact.html>
- 3.<https://www.ijltet.org/journal/148051944230.1245.pdf>
- 4.[https://www.academia.edu/35099335/COMPARATIVE\\_ANALYSIS\\_OF\\_NGULARJS\\_AND\\_REACTJS](https://www.academia.edu/35099335/COMPARATIVE_ANALYSIS_OF_NGULARJS_AND_REACTJS)
- 5.<https://ijcrt.org/papers/IJCRT2205332.pdf>
6. <https://www.rishabhsoft.com/blog/reactjs-vs-angularjs>