

# A Comparative Study: MariaDB Vs MongoDB

S. Lakshmi Devi<sup>1</sup>, P. S. Vijayalakshmi<sup>2</sup>

Assistant Professor, Department of Computer Science, Rathinam College of arts and science, Coimbatore-21

**Abstract:** *The volume and variety of data, recently non-relational database technologies like MongoDB have emerged to address the needs of new applications. MongoDB is used for new applications however replace existing relational database structure. In this paper we will try to show case a comparative study of non-relational databases and relational databases. We focuses on our presentation on comparison of the NoSQL database technology, known as MongoDB, and make a comparison with another application of relational databases, known as MariaDB, and thus justifying why MongoDB is more efficient and how it is used are compared with this MariaDB. We will also present the importance of using a non-relational database and a relational database. A comparison criterion includes theoretical differences, characteristics, limitation, integrity, distribution, system requirements, and architecture, query and insertion times.*

**Keywords:** MariaDB, MongoDB, NoSQL, RDBMS

## 1. Introduction

**MariaDB** is considered as the fork(replica) of MySQL database. This database was developed by the developer as that of MySQL except for the fact MariaDB offer much more additional functionality to the MySQL engine [2].

**MongoDB** is a cross-platform document-oriented database. It's a specialised DB build on non-relational document store architecture similar to JSON and support any type of file or elements. It offers higher speed of processing data and less response time for some specified application [2].

To handle a large volume of data like internet, multimedia and social media the use of traditional relational databases is ineffective. To overcome this problem the "NO SQL" term was introduced. The NoSQL term was used by Carlo Strozzi in year 1998 and refers to non relational databases, term which was later reintroduced in 2009 by Eric Evans. The primary benefit of a NoSQL database is that, unlike a relational database it is able to handle unstructured data such as documents, email, multimedia and social media efficiently. Non relational databases do not use the RDBMS principles (Relational Database Management System) and don't store data in tables, schema isn't fixed and have very simple data model. Instead, they use identification keys and data can be found from the keys assigned.

## 2. Overview: MongoDB

A non-relational database is any database that does not follow the relational model provided by traditional relational database management systems. This type of databases, also referred to as NoSQL databases, has seen rapidly adoption growth in recent years with the rise of Big Data applications.

Non-relational databases have grown in popularity because they were designed to overcome the limitations of relational databases in dealing with Big Data demands. Big Data refers to data that is growing and moving too fast, and is too diverse in structure for conventional technologies to handle.

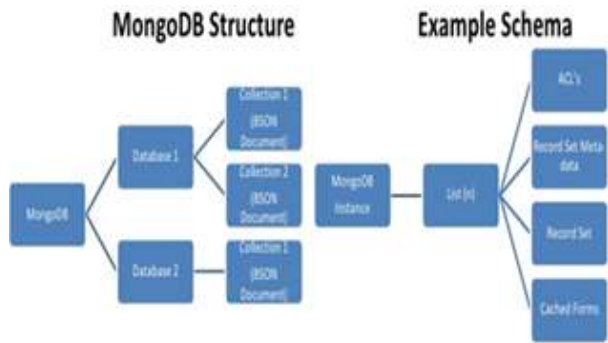
There are four strategies for storing data in a non-relational database, as shown in, and they are as follows:

- 1) **Key-Value** -A **key-value** store, or **key-value database**, is a data storage paradigm designed for storing, retrieving, and managing associative arrays, a data structure more commonly known today as a dictionary or hash.
- 2) **Document** -a document-oriented database contains *documents*, which are records that describe the data in the document, as well as the actual data. Documents can be as complex as you choose; you can use nested data to provide A **document-oriented** database, is called **document** store.
- 3) **Column store**—or, wide-column store, which stores data tables as columns rather than rows. It's more than just an inverted table—sectioning out columns allows for excellent scalability and high performance. *Examples: HBase, BigTable, HyperTable.*
- 4) **Graph-Oriented**-Graph databases handle fine-grained networks of information providing **any perspective** on your data that fits your use-cases. The relational systems, transactional guarantees protect updates of that connected data conforming to ACID standards.

MongoDB has its own query language named Mongo Query Language. To get certain documents from a db collection, a query document is created containing the fields that the desired documents must match. For example,

### Insert Command:

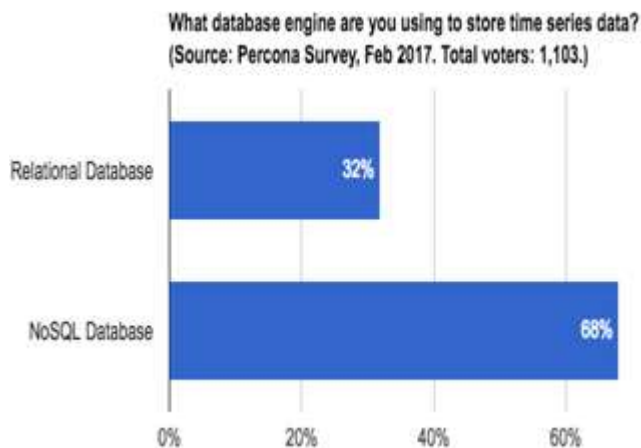
```
db.users.insert ({ user id:"xyz123", age: 34, status:"X"})  
Select Command db.users.find ({ status:"X", age: 34})  
Delete Command db.users.remove ({ status:"X"})  
Drop Command db.users.drop ()
```



### 3. Comparison of MariaDB & MongoDB

While these NoSQL technologies vary greatly, these databases are typically more scalable and flexible than their relational counterparts. Non-relational databases have evolved from relational technology in these ways:

- **Data models:** Unlike relational models which require predefined schema, NoSQL databases offer flexible schema design that make it much easier to update the database to handle changing application requirements.
- **Data structure:** Non-relational databases are designed to handle unstructured data that doesn't fit neatly into rows and columns. This matters as most of the data generated today is unstructured.
- **Scaling:** You can scale your system horizontally by taking advantage of cheap, commodity servers.
- **Development model:** NoSQL databases are typically open source which means you don't have to pay any software licensing fees upfront.



MariaDB	MongoDB
<b>Select Query:</b> Select * From Students where id="200A20"	<b>Select Query:</b> db.Students.find( { },{t_id:1 })
<b>Insertion Query:</b> Insert into students (s_id,name, course, branch ,email) values("200A20","Lokesh","M.Tech","CSE","lkba lkbansal1993@gmail.com")	<b>Insertion Query:</b> db.students.insert({s_id:"200A20",name:"Loke sh",course:"M.Tech",branch:"C SE",email:lkba nsal1993@gmail.com})
<b>Create Query:</b> Create table students (s_idchar,name varchar(50),course varchar(100),branch varchar(100),email varchar(50));	<b>Create Query:</b> db.students.insert({s_id:"200A20",name:"Loke sh",course:"M.Tech",branch:"C SE",email:lkba nsal1993@gmail.com})
<b>Drop Query:</b> Drop table students;	<b>Drop Query:</b> db.students.drop();
<b>Delete Query:</b> Delete From students where s_id="200A20"	<b>Delete Query:</b> Db.students.delete( {s_id:"200A20"} )

Learn about why MongoDB is the most widely-used non-relational database by

1) **Relational databases**, which can also be called relational database management systems (RDBMS) or SQL databases. The most popular of these are Microsoft SQL Server, Oracle Database, MySQL, and IBM DB2. These RDBMS's are mostly used in large enterprise scenarios, with the exception of MySQL, which is mostly used to store data for web applications, typically as part of the popular LAMP stack (Linux, Apache, MySQL, PHP/ Python/ Perl).

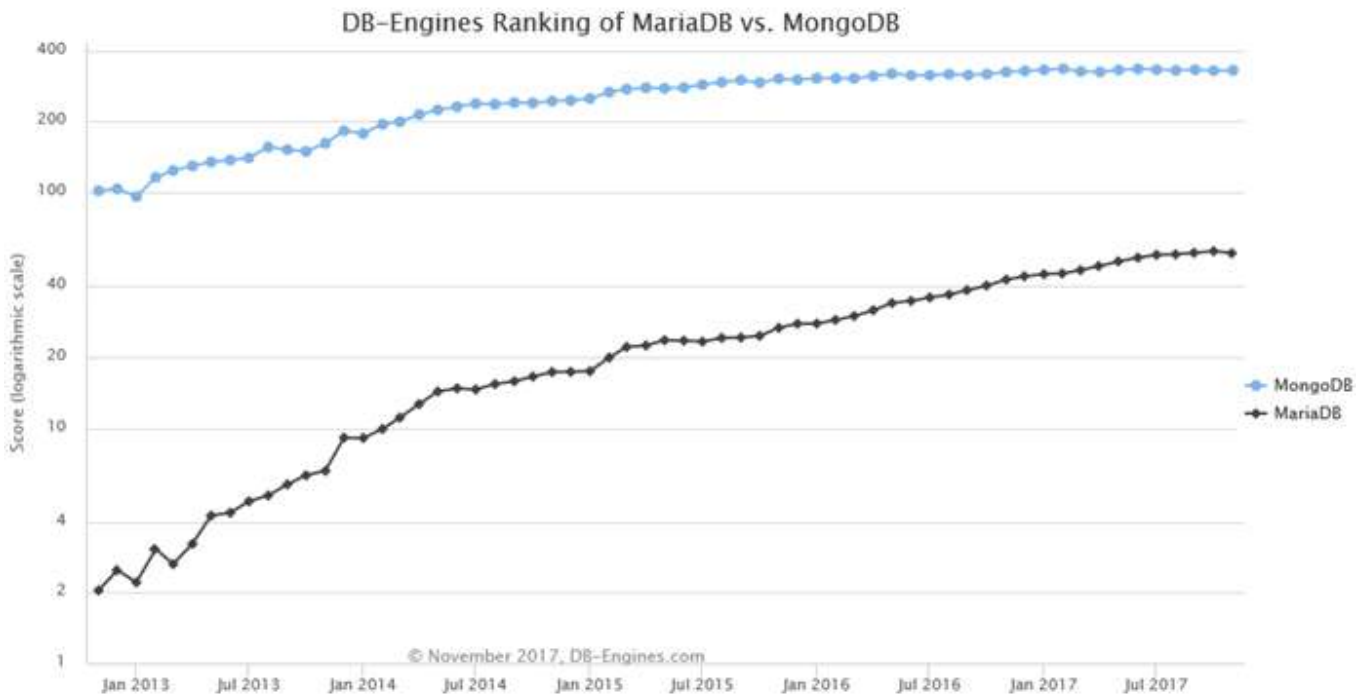
2) **Non-relational databases**, also called NoSQL databases, the most popular being MongoDB, DocumentDB, Cassandra, Couchbase, HBase, Redis, and Neo4j. These databases are usually grouped into four categories: Key-value stores, Graph stores, Column stores, and Document stores .

Relational databases usually work with structured data, while non-relational databases usually work with semi-structured data (i.e. XML, JSON).non-relational databases like MongoDB represent data in collections of JSON documents. The Mongo import utility can import JSON, CSV and TSV file formats. Mongo query targets of data are technically represented as BSON (binary JASON).

MariaDB terms/concept	MongoDB terms/concept
Table	Collections
Row/Records, Columns/Types	Key-Value Pairs,Documents
Index	index
table joins	Embedded Documents and Linking
fixed schema	schema less
primary key (explicitly)	primary key (implicitly)

Name	MariaDB	MongoDB
Description	MySQL application compatible open source RDBMS, enhanced with high availability, security, interoperability and performance capabilities	One of the most popular document stores
Primary database model	Relational DBMS	Document store
Additional database models	Document store Key-value store	
Website	mariadb.com mariadb.org	www.mongodb.com
Technical documentation	mariadb.com/kb/en/library	docs.mongodb.com/manual
Developer	MariaDB Corporation Ab (MariaDB Enterprise), MariaDB Foundation (community MariaDB Server)	MongoDB, Inc
Initial release	2009	2009
Current release	10.2.9, September 2017	3.4.9, September 2017
License	Open Source	Open Source
Cloud-based	no	no
Implementation language	C and C++	C++
Server operating systems	FreeBSD Linux Solaris Windows	Linux OS X Solaris Windows
Data scheme	yes	schema-free
Typing	yes	yes
XML support	yes	no
Secondary indexes	yes	yes
SQL	yes	no
APIs and other access methods	ADO.NET JDBC ODBC	proprietary protocol using JSON
Supported programming languages	Ada C C# C++ D Eiffel Erlang Go Haskell Java JavaScript (Node.js) Objective-C OCaml Perl PHP Python Ruby Scheme Tcl	Actionscript C C# C++ Clojure ColdFusion D Dart Delphi Erlang Go Groovy Haskell Java JavaScript Lisp Lua MatLab Perl PHP PowerShell Prolog Python R Ruby Scala Smalltalk
Server-side scripts	yes	JavaScript
Triggers	yes	no
Partitioning methods	Horizontal partitioning, sharding with Spider storage engine or Galera cluster	Sharding
Replication methods	Master-master replication Master-slave replication	Master-slave replication

MapReduce	no	yes
Foreign keys	yes	no
Transaction concepts	ACID	no
Concurrency	yes	yes
Durability	yes	yes
In-memory capabilities	yes	yes
User concepts	Users with fine-grained authorization concept, user roles and pluggable authentication	Access rights for users and roles



#### 4. Conclusion

MongoDB and MariaDB(SQL and NoSQL) are great inventions in the past to keep the data storage and optimized and speedy recovery. Criticized one of them will not help the cause. If a NoSQL buzz these days, does not mean that it is a silver bullet for all your needs. Both technologies are better at what they do. It's a developer to make better use in different situations and needs. NoSQL databases are now an integral part of the database landscape, and their practical benefits they could be a real game-changer in the enterprise arena. Lower cost, easier scalability, and open source NoSQL features make this an attractive option for many companies that want to integrate Big Data. NoSQL but still offers a relatively young technology without standard SQL databases such as MySQL. As with any major business decisions, IT leaders must weigh their options and determine which features are most important to them in a database.

- [4] MongoDB Books [http://mongodb.org/display/doc/ 6. mongodb with php](http://mongodb.org/display/doc/6-mongodb-with-php)
- [5] [Http://www.tutorialspoint.com/mongodb/mongodb\\_tutorial.pdf](http://www.tutorialspoint.com/mongodb/mongodb_tutorial.pdf)
- [6] <https://www.freelancinggig.com/blog/2017/05/20/sql-vs-nosql-differences-nutshell/>

#### References

- [1] <https://www.mongodb.com/scale/what-is-a-non-relational-database>
- [2] <http://www.jamesserra.com/archive/2015/08/relational-databases-vs-non-relational-databases/>
- [3] <https://www.supportpro.com/blog/comparison-between-mariadb-postgresql-and-mongodb/>