IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol-6, Special Issue-4, May 2018 E-ISSN: 2347-2693

Query Performance Analysis in NoSQL and Relational Dtabases: MongoDB Vs MySQL

Benymol Jose[1*], Sajimon Abraham[2], Praveen Kumar V S [3]

[1*] School of Computer Sciences, Mahatma Gandhi University, Kottayam, India
[2] School of Management and Business Studies, Mahatma Gandhi University, Kottayam, India
[3] Dept. of Computer Applications & Sciences, SAS SNDP Yogam College, Konni, Kerala, India

[*]*Corresponding Author: benymol.jose@mariancollege.org, Tel.: +00-9995237344*

Available online at: www.ijcseonline.org

***Abstract*—** To improve the data processing of the unstructured data generated, a NoSQL framework can be used to achieve better distribution of storage and analysis work of the collected big data. NoSQL is particularly helpful when a venture needs to get to and investigate huge measure of unstructured information or information that is put away on numerous virtual servers. MongoDB uses an extensive variety of methods to solve the huge information execution issues that ordinary databases were not intended to solve. Relational databases like MySQL are storing data in structured format in tables as rows and columns. This paper concentrates on the advantages of NoSQL databases over relational databases in the analysis of the big data. It mainly uses MongoDB which is one of the boosting technology of NoSQL databases and makes a performance comparison of a particular query in MySQL and MongoDB and justifies why MongoDB is preferred over MySQL**.**

***Keywords*—** *Unstructured Data, NoSQL Databases, MongoDB, Relational Databases, MySQL*

I. INTRODUCTION

Big data refers to information with immense volume which is having exponential development in growth. This data arrives in various structures and with increased velocity. This sudden growth in volume of data has introduced new data storage, organization, processing and analysis methods. This caused the need of new architectures and query languages to handle the massive data generated in every second [1]. The big data generated contributes to a much larger variety of data types. Storing and processing this data with the conventional relational databases is not easy. NoSQL systems and Big Data Analytics hold significant promise for improving the storage and processing problems with this data. This analysis of huge amounts of varying data, allows companies to understand their customers and helps better administration of business. Big Data Analytics facilitates enhanced decision making, increased visibility and overall greater value [2].

MySQL is a popular relational database management system (RDBMS) supported by the Oracle Corporation. It is an open source software available and in it data is kept in tables and retrieval of the data is performed using a structured query language. In RDBMS systems the structure of the table, which is also known as the schema has to be defined in light of prerequisites. We can also set up constraints to manage the communications between different fields in a record. In MySQL, related data might be put away in various tables. However, they can be related with the help of table joins [3].

MongoDB is a document-based database developed by MongoDB, Inc, which is available as an open source. These systems can be utilized to defeat the storage problems associated with managing large volumes of unstructured data with the ordinary existing database management systems. The important storage components in document databases like MongoDB are collections, rather than the tables used in relational databases. Similar or different JSON, BSON based documents or sub documents are the primary component of these collections in MongoDB.

The main aim of this paper is to do the performance evaluation of a query with MongoDB Studio3T and MySQL Workbench. The paper is organized as follows: section I contains the introduction and section II, depicts few related studies. In section III, document-based NoSQL databases and furthermore some of its points of interest are discussed. Section IV, compares MongoDB and its features with the relational databases. In section V, a performance test is done for the simple select query and the description of the dataset being used is also done. Finally, in section VI, the evaluation graph is shown and the paper is concluded with the comparison of query performance in MySQL Workbench and MongoDB Studio3T.

II. RELATED WORK

In one of the current studies a new technique was proposed to coordinate MySQL and MongoDB by including a middleware amongst application and database layers. The middleware includes metadata which contains assorted sorts of packages [4].

Another study has conducted a set of experiments with different types of operations such as read, write, delete, and select from different aspects in the two databases and on the same data for a typical e-commerce schema. The results show that MongoDB performs better for most operations excluding some aggregate functions [5].

In another work, attempt is made to use NoSQL database to replace the relational database, applied to traditional information management systems, compare the two database technologies, give the key code of NoSQL implementation, and finally list the performance comparison of the two schemes [6].

Another paper is trying to evaluate the performance of five NoSQL clusters namely Redis, MongoDB, Couchbase, Cassandra, HBase by using a measurement tool - YCSB which is Yahoo Cloud Serving Benchmark [7].

III. DOCUMENT DATABASES AND ADVANTAGES OF NOSQL DATABASES

A. Document databases

Document databases contain documents as their basic components inside. In document databases, data is often stored in XML, JSON, or BSON formats. Document databases store all data related with a given object. It is kept as a single instance in the database. Due to this document stores are used widely for web related applications. They accommodate embedded documents, which are hierarchical tree structures that consists of maps, collections, and scalar data. Document databases like MongoDB has effective and strong query formats, with which an easier move from conventional databases is possible. Some of the common document databases are MongoDB, CouchDB, Terrastore, OrientDB and RavenDB [7].

B. Advantages of NoSQL

1) Easy scaling

Database administrators are regularly purchasing greater servers to get scaling together as the volume of data increments. Use of numerous hosts is really unrealistic to assign the huge load created. The expenditure related with the central purposes of scaling out on product equipment can be improved by using commodity hardwares. The speed with which transactions happen is also a factor requiring huge storage. A similar issue happens when databases move into the cloud or onto virtualized circumstances. RDBMS systems require costly servers as they cannot be scale out on commodity hardware components. NoSQL databases can be utilized in these circumstances which can expand easily with commodity servers. Low cost commodity servers can be easily used for constructing NoSQL databases.

2) Big data

The volume of information oversaw today has expanded heavily. It is as an outcome of expanded number of exchanges in the present situation. Despite the fact that the RDBMS limit has been expanded to join these new necessities, the functional administration of the framework is getting to be troublesome. Today just the NoSQL frameworks can deal with the tremendous volumes of enormous information which is produced quickly.

3) No Use of DBAs

Regardless of the various sensibility changes ensured by RDBMS merchants consistently, first class RDBMS systems can be kept up just with the assistance of expensive, incredibly arranged DBAs. DBAs are actually required in the blueprint, foundation, and persistent tuning of first class RDBMS structures. NoSQL databases are generally made from the earliest starting point to require less organization, customized repair, data scattering, and more direct data models provoking lower association and tuning necessities.

4) Cost effectiveness

NoSQL databases ordinarily utilize numerous servers to store the enormous information utilizing distinctive methods. RDBMS regularly utilize high performing servers for this reason. This builds the cost included immensely on account of RDBMS frameworks while NoSQL can be made do with ordinary expenses.

5) Flexibility in data models

Administrating change is a major issue with RDBMS systems. In reality, even minor changes to the data model of a RDBMS must be definitely managed and might require huge processing time or can cause poor administration strategy. Key value stores and document databases are good solutions in this scenario [8].

IV. COMPARISON OF MONGODB WITH RELATIONAL DATABASES

There are a lot of differences in the concepts used in relational database systems and NoSQL databases. For the comparison, MongoDB, which is a document-based NoSQL database and MySQL are considered. Few operations which can be compared with these databases are create, select, insert, delete and update. And the table, Table I lists and compares the syntax of few operations used in MySQL and in MongoDB with examples.

Table I: Queries Used in Two Different Databases

Query	Relational Database	MongoDB Database
Create Command	CREATE TABLE EMP (eno INT(5),salary INT(7), eaddress CHAR(12))	No Schema
Insert Command	INSERT INTO EMP (eno,eaddress) VALUES(10,"100/abc")	db.emp.insert ({"eno":"10","eaddress":"100/abc"})
Delete Command	DELETE FROM EMP WHERE eno=10;	db.emp.remove ({"eno":"10"})
Select command	SELECT *from EMP where eno=10;	db.emp.find({"eno"=10})

V. PERFORMANCE TEST

A comparison between the databases is done by performing a simple query which includes only a SELECT statement in MySQL and the equivalent in MongoDB and is displayed in the following table, Table II.

The dataset used was downloaded from the URL https://data.cityofchicago.org/browse?category=Buildings. It consists of the details of permits issued by the Department of Buildings in the City of Chicago from 2006 to the present. The dataset for each year contains more than 65,000 records. And the dataset taken for the experiment contains around 1.5 lakhs of records. Also, the time taken for the execution of the query is noted in the table, Table III.

The query execution is performed for different number of records for both the databases and the time taken for query execution in each case is noted. The performance comparison is made with MongoDB Studio3T and MySQL Workbench6.3.

Table II: Select Queries used in case of MySQL and MongoDB studio 3T

Number of Rows	Queries Used
3000	MySQL : select *from buildingpermit LIMIT 0,3000; Studio3T:db.buidingpermit.find().limit(3000).explain("executionStats")
5000	MySQL : select *from buildingpermit LIMIT 0,5000; Studio3T:db.buidingpermit.find().limit(5000).explain("executionStats")
10000	MySQL : select *from buildingpermit LIMIT 0,10000; Studio3T:db.buidingpermit.find().limit(1000).explain("executionStats")
25000	MySQL : select *from buildingpermit LIMIT 0,25000; Studio3T:db.buidingpermit.find().limit(25000).explain("executionStats")
50000	MySQL : select *from buildingpermit LIMIT 0,50000 Studio3T:db.buidingpermit.find().limit(50000).explain("executionStats")
75000	MySQL: select *from buildingpermit LIMIT 0,75000 Studio3T:db.buidingpermit.find().limit(75000).explain("executionStats")
100000	MySQL: select *from buildingpermit LIMIT 0,100000; Studio3T:db.buidingpermit.find().limit(100000).explain("executionStats")
125000	MySQL: select *from buildingpermit LIMIT 0,125000; Studio3T:db.buidingpermit.find().limit(125000).explain("executionStats")
147429	MySQL: select *from buildingpermit LIMIT 0,147429 Studio3T:db.buidingpermit.find().limit(147429)

Table III:Query Execution time in MySQL and MongoDB

No. of Records	Time in Milli Seconds	
	MySQL	MongoDB
3000	93	1
5000	141	2
10000	250	5
25000	563	14
50000	1078	21
75000	1594	41
100000	2172	57
125000	2718	71
147429	3594	77

VI. ANALYSIS AND EVALUATIONS

The performance of MongoDB Studio 3T is compared with MySQL Workbench 6.3 by executing a simple query operation by varying the number of records. A large number of records are taken and the operation is performed by increasing the number of records in successive steps. A graph is plotted based on the query execution time and is shown in the following figure, Figure1.

On analyzing and comparing, it is seen that, the performance of MongoDB is increasingly when contrasted with that of MySQL. When the number of records looked at is small, there is not much distinction in the execution time taken for the activities to finish for both MongoDB and MySQL databases. Be that as it may, when number of records is expanded, MongoDB demonstrates great improvement by taking less time for the completion of queries compared to MySQL.

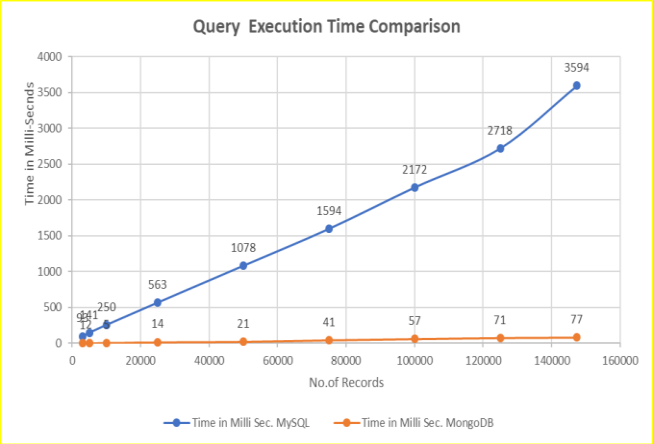

Figure 1. Performance comparison of MySQL with MongoDB

REFERENCES

[1] Yesha Mehta, and Sanjay Buch, "Big Data Mining and Semantic Technologies: Challenges and Opportunities", Int. J. on Recent and Innovation Trends in Computing and Communications(ISSN 2321-8169) ,Vol.3, Issue.7,pp.4907-4913, 2015.

[2] Lidong Wang ,and Cheryl Ann Alexander, "Big Data Driven Supply Chain Management and Business Administration", American Journal of Economics and Business Administration,Vol.7, Issue.2, pp.60-67, 2015.

[3] Dipina Damodaran B , Shirin Salim, Surekha Marium Vargese ,"Performance Evaluation of MySQL and MongoDB databases", Int.J of Cybernetics and Informatics,Vol.5, Issue.2,pp.387-394, 2016.

[4] Sanobar Khan, Vanita Mane, "SQL support over MongoDB using metadata". International Journal of Scientific and Research publications, Vol.3,Issue.10,pp.1-5,2013.

[5] Seyyed Hamid Aboutorabi, Mehdi Rezapour, Milad Moradi, Nasser Ghadiri, "Performance evaluation of SQL and MongoDB databases for big e-commerce data", International Symposium on Computer Science and Software Engineering (CSSE),Tabriz, Iran,pp.72-78, 2015.

[6] Zhu Wei-ping, Li Ming-xin, Chen Huan," Using MongoDB to implement textbook management system instead of MySQL", IEEE 3rd International Conference on Communication Software and Networks, Xi'an, China, pp.828-830, 2011.

[7] Enqing Tang, Yushun Fan, "Performance Comparison between Five NoSQL Databases", 7th International Conference on Cloud Computing and Big Data (CCBD), Macau, China, pp.105-109, 2016.

[8] Azhi Faraj,Bilal Rashid,Twana Shareef, "Comparative Study of Relational and Non Relations Database Performance using ORACLE and MongoDB Systems", International Journal of Computer Engineering & Technology,Vol.5,Issue.11,pp.11-22, 2014.

[9] Nancy Bhardwah, Gurvinder Kaur, "Greedy based privacy preserving in Data Mining using NoSQL", International Journal of Engineering Development and Research, Vol.5, Issue.1, pp.9-17, 2017

Authors Profile

Benymol Jose pursed Masters in Computer Science from Bharathidasan University, Thiruchirappally, Tamil Nadu in 1999 and M.Phil in Computer Science from Madurai Kamaraj University, Madurai, Tamil Nadu in year 2009. She is currently pursuing Ph.D. and currently working as Assistant Professor in Department of Computer Applications, Marian College, Kuttikkanam, Idukki, Kerala since 1999. She had published research papers in conferences including IEEE and it's also available online. Her main research work focuses on Unstructured data and NoSQL databases, Big Data Analytics, Data Mining. She has 18 years of teaching experience and 3 years of Research Experience.

Sajimon Abraham. M.Sc. Mathematics, M.C.A, M.B.A, Ph.D. in Computer Science. He has been working as Assistant Professor in Computer & IT, School of Management and Business Studies, Mahatma Gandhi University, Kottayam.

Praveen Kumar V S, pursed Masters in Computer Science from Bharathidasan University, Thiruchirappally, Tamil Nadu in 1998 and currently working as Assistant Professor in Dept of Computer Applications and Sciences, SAS SNDP Yogam College, Konni, Pathanamthitta, Kerala since November 1999. His area of interest is in Spatial Data Mining and now holding the post as Member, Syndicate of Mahatma Gandhi University, Kottayam

