National College of Ireland

Benchmarking for MySQL and MongoDB

Data Storage and Management

Anish Shah
X18124135

MSc Data Analytics {2019/20}

Submitted to: **Prof. Muhammad Iqbal**

MySQL vs MongoDB

Anish Shah
X18124135

April 22, 2019

Abstract

MongoDB and MySQL have gained their name across all over the world in terms of relational and non- relational databases. Technology has emerged so drastically in the past few years that everything has become online now. Due to rapid increase in information, the data available now is larger than it ever was and is increasing drastically. The relational database has held its leadership for decades and has been used to support many enterprise applications. But in today's world there has been a rapid increase in new types of applications which require more diverse and scalable database such as MongoDB. This research helps us to evaluate which database has better performance. This paper will try to evaluate the performance of a relational and a non-relational database. Yahoo cloud serving Benchmark (YCSB) will help us to analyze these two databases. We would be running various heavy workloads on these two databases and will take into consideration various factors for evaluating their performance.

1) Introduction

We know that MySQL will be the most popular name among all the relational databases around the world. But due to its replacement alternative databases, it has loose many supporters. Today the world needs a better way to store and manage their data due to different reason. Might be to attract various customers, or to change the user experience or might be that we have to beat our competitors. As a result of such assumptions or reasons, relational databases are losing their value [1].

No SQL databases have really gained their popularity in recent time and have been increased drastically. If we compare No SQL database with a relational database, No SQL can manage the data as compared to MySQL more efficiently due to its properties of flexibility and scalable structure. It provides better data storage and management solution with its key-value store. **(Shiyong Liu, 2018).**

There is a need for such type of No SQL database because it can manage a massive increase in new, rapidly changing data types. These changes are due to structure, semi-structured and polymorphic data generated by new classes of the web, mobile, social and IOT application.

This was the reason that non-tabular type database such has MongoDB has been introduced in order to fulfill the requirement of the different new emerging application and this can also help to modernize the existing workloads. MongoDB is an open source NoSQL database. It is a non-

relational database that has some features of a relational database like, sorting, indexing and query range. **(Elif Dede, 2013)**
In this paper, we would be discussing the key features of MySQL and MongoDB databases with their key feature followed by this we would be discussing the basis architecture about each database in detail.
To find out among a set of databases that which of the database will serve the application better. It is difficult to decide that for our application which database is better and it is indeed difficult to decide because might be the system differs and behave differently.

This could be achieved by using the Yahoo cloud serving Benchmark (YCSB) which can help us to evaluate and compare different databases and their performance. In this section, we have used YCSB to test MySQL and MongoDB databases and check their performances such as read operation, write operation and their throughput.

In the later part of we would be discussing both the databases their key characteristics, their architecture, performance test plan which would help us to conclude that which databases performance is better.

2) Key Characteristic:

2.1) MySQL:

This is a relational database management system which is freely available and which uses Structure Query language. It is the most popular RDBMS software brand name and it uses a server-client relationship. It is basically used for accessing, adding and managing CONTENT IN A DATABASE. It is a fast, easy to use, it works on different operating systems such as PHP, PERL, C, C++, JAVA, etc.
It is very powerful in its own way and it works well with large dataset sets.

Below are the important characteristic of MySQL database [2]

1) **Relational database system:** As we know that MySQL is a relational database system
2) **Client /Server Architecture:** MySQL system follows a client-server system, where one is SQL server and clients are the applicant programmer. The client communicates with the server by executing the query, and save the changes. Both client and server can be present on the same system.
3) **SQL compatibility:** MySQL supports as its database language.
4) **Stored Procedures:** They are generally SQL code which is used to simplify a certain amount of steps such as inserting or deleting the record. This means that the programmer does not have to execute the entire table directly, but they can rely on store Procedure. Basically, SPs increase efficiency. And they are supported by MySQL since version 5.0.
5) **Triggers:** They are normal SQL commands which are executed automatically by the server during execution of INSERT, UPDATE and DELETE commands.
6) **ODBC:** MySQL support ODBC interface Connector/ODBC: MySQL support many programming languages such as Delphi, Visual basis, etc. They can also be implemented under UNIX.

7) **It is scalable:** It can handle up to any amount of data, up to 50 million rows or more. The default size can be up to 4 GB however we can increase this number up to 8 TB of data.
8) **High Performance:** MySQL is faster, more reliable and cheaper because of its unique storage engine architecture.
9) **High Flexibility:** MySQL supports a large number of embedded applications which makes MySQL very flexible.
10) **High Productivity:** MySQL uses Triggers, Stored procedures and views which allows the developer to give higher productivity.

2.2) MongoDB

MongoDB was first release in 2009 and was developed and supported by Logen. It is an open source document-oriented database where the data is stored in JSON format. In short, it is a computer program design for storing managing, retrieving document-oriented information such as semi-structured data. It is scalable and has a high performance.

In MongoDB, all the data are accessed by MongoDB query language. They have their own unique structure and any new field can be added at any time of any data type.

Below are the characteristics of MongoDB [3]

1) **Ad hoc queries:** As we know it is a document oriented DB, so we can search fields by their own name, range query or by any regular expression.
2) **Indexing:** Indexing is one of the main important function present in MongoDB which can help to access the data faster. We can index any field in document.
3) **Replication:** We often think that what would happen if system get crashed and we have no backup. In MongoDB, it supports Master Slave replication. Here a master can perform read and write operations and a slave copies data from the master and can only be used for reads and backups.
4) **Load Balancing:** It has automatic load balancing configuration because MongoDB scale horizontally using sharding.
5) It is a schema-less database written in C++
6) Provide High performance.
7) **File storage:** MongoDB as a database can also be used as a file system called as gridFS. This file system has features such as load balancing, data replication over multiple machines for storing files.

3) Database Architecture:

3.1) MySQL Architecture:

The below diagram shows MySQL architecture and different layers present in them:

MySQL architecture are divided into 3 layers [4]

a) Application layer,
b) Logical Layer ,

c) Physical layer

a) Application layer: This is the layer where the users and clients interact with MySQL RDBMS. Here all the services that are needed for connection, authentication, and security are present here. As we can see in the below diagram there are three main part of the application layer.

- Administrators
- Clients
- Query User

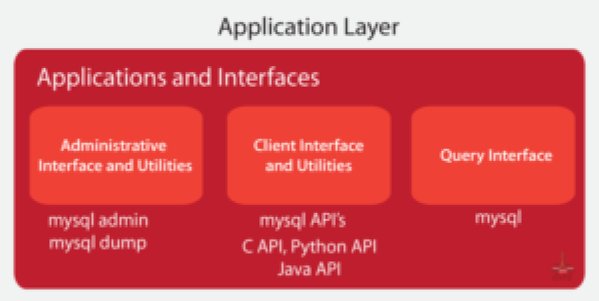
Application Layer
Applications and Interfaces
Administrative Interface and Utilities
Client Interface and Utilities
Query Interface
mysql admin
mysql dump
mysql API's
C API, Python API
Java API
mysql

Figure: Application layer

- **Administrators:** these are the ones which used various interface and utilities like mysqladmin which indirectly helps them to restart the server, or stop the server. They are also used to do work such as dropping the database or taking backups of the databases.
- **Client:** Client uses various API's to interact with MySQL. These APIs send a different query to the server sent by the client for execution.
- **Query user:** these type of user interact with MySQL via query interface that is MySQL.

b) Logical Layer:

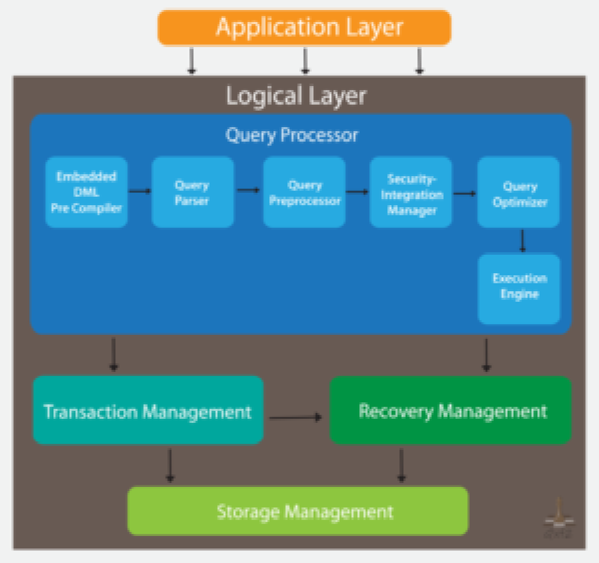

Figure: Logical layer

The main part of the logical layer is that it takes data from the application layer and send it to the logical layer for processing the data. All the functionality such as stored procedure, triggers and views reside in the layer. The logical layer is divided into the Query Processor, Transaction Management, Recovery Management, and Storage Management. All the system inside the Logical layer works together and output from each part be the input to another.

The query process helps the SQL statement to pass to various part of query processor so that the statement could be transformed into the understandable SQL statement which will help it to process the query and get the resulted output. The out obtains from the query processor will serve as the input for the transaction management.

Transaction management helps for making sure that the transaction is logged and executed automatically. One of the best parts of transaction management is that it is also responsible for COMMIT and ROLLBACK SQL commands.

• Recovery Management:

As the name suggests Recovery, it stores all the logs operation executed in the database and also stores operational logs as MySQL commands. In case of any failure or system crashes these logs would help us to recover the system back to the last stable form.

• Storage Management:

The main purpose of storage management is to allocate memory resources with the help of buffer manager with is present inside storage management.

c) Physical Layer:

The third layer is the Physical Layer which contains the storage engines. They are responsible for storing and retrieving all data stored in MySQL

3.2) MongoDB Architecture:

The architecture of the MongoDB can be classified into three section:

- The Config Servers
- The Mongos and
- The Primary or Secondary Replica Sets:

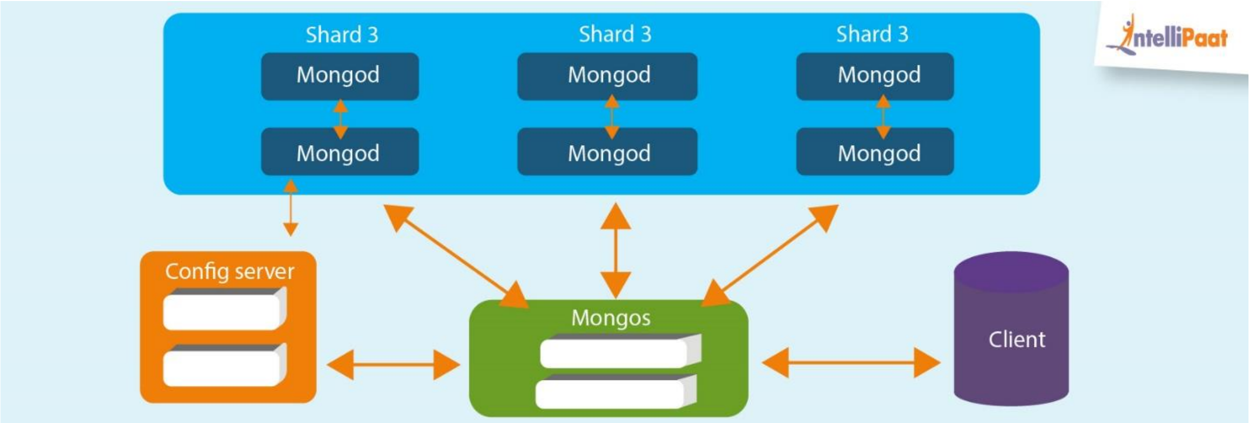

Figure: MongoDB Architecture

a) **The Config Servers:**
All the configuration files required for mongoDB for execution are stored in Config Server. In MongoDB read and write operation can be performed if the Config server get failed.

b) **The Mongos:**
These are schedulers that act as access passes for the client.

c) **Primary or Secondary Replica Sets:** these sets are the data storage component of MongoDB. The primary sets are responsible only for write operation. Later these set are replicated into secondary set for read operations. This replication will acts if there is any failure occur. If in case there is a failure of secondary sets than there will be always another secondary sets replica to maintain the availability.
Secondary sets act as primary set if primary sets gets failed.
So basically the client request are received by mongos which are then passed to Config server where shared information with correct data request are obtained.

4] Security:

4.1) MySQL Security:

Following factors need to be consider about security about MySQL [5]

a) **General Factors :**
These factors affect the security of MySQL which includes, choosing good passwords, not granting unnecessary privileges to different users, also we should ensure the security of the application by preventing the SQL injection and data corruption.
It is important for all the user to set a strong password and it is indeed important to set a new strong password for the root user. Most of the system does not have a root password set while installation. This means that any user can login into the MySQL machine and trigger MySQL client from localhost
➔ **mysql –u root**
So if the password is not present than anyone can have access to the entire database. So it is important to set a strong password for the root.
sudo mysqladmin password

b) **Security of Installation itself :**
As we know that all the data present under SQL server are more valuable. Hence to protect the data files, log files and all other application files should not be readable and writeable to unauthorized parties.

c) **Access Control and Security :**
SQL helps us to create account that helps the users to connect the server and access the data. But here different users need to be given different privileges to access the database. The main function of the privilege system is to authenticate the user and give respected privilege on the database such as SELECT, INSERT, UPDATE AND DELETE.

Access to the users can be created, granted and can be revoke. It means that the users can only perform the operations which are only permitted to them.

Security Plugins:

Several plugins are been offered by MySQL for security purpose. The Authentication plugins are used for authentication protocols which are used to authenticate the client to connect to MySQL.

The Password Validation Plugins is used for checking the strength of the password and accessing the strength of potential password.

The Keyring plugins is used to provide a secure storage of sensitive information.

MySQL Enterprise Firewall is used to monitor and login of different MySQL connection. This is used to permit and deny SQL statement against various SQL injection or attempt to exploit application from outside.

d) **The Network Security :**

This is required to grant specific permission to individual users, so that MySQL is restricted to used locally or on MySQL server host.

e) **Backup and Recovery :**

This is the most import security feature among all. If the data is not backup then all the important files on failure can be lost. If we have a backup than in any case we can again restore the backup file so that there would not be any loses. Time to time backups of all the data should be taken. And in case of any failure or data lost backups can be used to recover the lost data.

4.2) MongoDB Security:

a) **Enable Access Control and Enforce Authentication**

Authentication of MongoDB means that all the clients and servers are provided with valid credentials so that they are connected to the system with proper privileges. Authentication can be provided by default mongoDB or it can be provided by external framework.

b) **Configure Role Based Access control:**

As the name suggest role based access control, users or the clients are only granted the access limited to their work .Here only those role users are created who need access to the system.

c) **Encrypt Communication**

To have a proper communication from the outside world the connection should be secure and to secure the connection we should make use of TSL /SSL protocols. These protocols helps used to encrypt communication between the mongod and mongos components of mongoDB.

d) **Limit Network Exposure:**

We should check that our mongoDB is running in a trusted network environment. Only trusted users should be able to access the network and the port on which it is running.

e) **Audit System Activity :**

Audit Activities are used to record system events i.e. user operations and connection events. Which indirectly helps us to check and analyses and allow different privileges to different users.

f) **Run MongoDB with dedicated users :**
This means that only users should have a dedicated access no other access apart from which he has.

4. B] Transaction Management

B.1] MySQL Transaction:

It is a sequential group of operation which is performed as it is a single unit. In short it is group of operation divided into smaller portion of group and executed as a whole. It will only get completed once all the operations within the group are executed. [10]

Below are the important statements which need to be followed while executing the transaction;

- To start a transaction we need to use START TRANSACTION statement.
- If we need to commit the current transaction and make it a permanent change we need to COMMIT the statement.
- If we need to rollback the current transaction or undo the changes we need to use ROLLBACK statement.
- SET autocommit is used if we need to disable or enable the auto commit mode.

By default MySQL automatically commit the transaction. To force MySQL to not to commit the changes we need to set autocommit = 0 or OFF. If we need to enable it, we need to change the autocommit = 1 or ON

B.2] Transaction Management in MongoDB

In MongoDB the simple rule of transaction management is “do all or nothing”. The latest version of MongoDB now started supporting transaction management. Before version 4.0 it was very difficult to manage transaction management because there we no function such as commit and rollback. This feature was introduced in version 4.0 to make all the developers from this overhead.

Atomicity in mongoDB prevents to update the database to write partially, which can cause a greater problem rather than to reject them completely. As we know that mongoDB supports atomicity for single documents. Multiple documents atomicity is also been supported by MongoDB but to that extend. Because most of the companies are which used databases such has mongoDB do not want to write multiple documents at the same time.

5] Literature Review

According to the author **(Y.H. Gu, 2011)** had tried to explain about how we can optimize MySQL database. In his research, he has discussed how to optimize the performance of MySQL on different storage engine. As he has explained that MySQL is not good for the performance of very large dataset, so he tried to increase and optimize performance MySQL by testing it on different storage engine. The storage engine which he has explained is MyISAM Engine and InnoDB Engine and later he has compared them both. The comparison of such type of data engines was based on its format, integrity, the primary key of data, chances of MySQL data, availability. He has taken into consideration various properties of these engines such has transactional management, crash recovery, multiversioned concurrency which has helped to check the performance of these databases. Both the engines are good at their respective side. If we consider MyISAM than it can only be used and it can only be suitable for the application where there are less insert operation and more query. But there are some properties which MyISAM doesn't support such as transaction, foreign key, and write-ahead logging. Whereas InnoDB Engine is only suitable for those applications where there are a large concurrent write and query and it indeed supports ACID properties, foreign key, and row level locking.

Another paper which has led me to choose to do such type of research was based on a study of relational and a non-relational database. The author (Robert, 2015) has compared both the databases on the basis of their commands. By executing each and every command he has made a comparison on both the database and demonstrated his output. He said that Non-relational database MongoDB is much faster as compared to MySQL. The author has stated the reason for choosing which database is much more efficient for which application. It is stated that switching from a relational database to a non-relational database is a challenging task. MongoDB can only be used for application where data is intensive and stores many data and queries a lot of data. In the end, he has concluded that the application which was used can only support MongoDB due to thousands of users and each user has enabled its own customization. This cannot be achieved using relational databases.

From the above papers, I need to compare and check the workload of a non-relational database such as MongoDB and a relational database such as MySQL. Here in my research, both the databases will be tested from YCSB benchmark were an equal amount of load would be given on the database and compared them. Each and every time the load of the data would be increased. We have used two types of workloads A and B from workloada, workloadb, workloadc, workloadd, workloade, workloadf.

Below are the detailed explanation of each workload:

- Workload A: Update heavy workload
- Workload B: Read mostly workload
- Workload C: Read only
- Workload D: Read latest workload
- Workload E: Short ranges
- Workload F: Read-modify-write

As we know that today’s modern apps generate a lot of data and due to increase in technology RDBMS database is not enough to store such huge amount of data and processed it. So a comparatively study need to be adapted to check the performance of both the databases.

6] Performance Test Plan:

In this paper we are analyzing a comparative study to check the performance of MongoDB and MySQL databases using Yahoo cloud serving Benchmark (YCSB). It is a tool which is mostly used to check the performance of different databases. For this research we would be running different workloads on both relational and non-relational databases and compare their performances.

6.1] Methodology:

These test were executed on OpenStack platform setup at National College of Ireland. It is a virtual cloud platform where an instance was created and relational database such as MySQL and non-relational database such as MongoDB were installed. Yahoo Cloud Service Benchmark version 0.14.0 was installed to check the workload on both MySQL and MongoDB. The workload which were selected were A and D. The test were ran for workload of 20k, 30k ,40k ,50k and 60k .Below are the details of machine which was create on OpenStack .

Cloud	Instance Name	Operating System	Machine Type	RAM	Disk	CPU	IP Address
NCI OpenStack	X18124135-DSMProject	Ubuntu	m1.medium	4GB	40GB	2	192.168.101.186/ 87.44.4.163

6.2] Workload Details

a) **Workload A : Update Heavy workload**
This workloads has a mixture of 50 % read and 50 % write.

b) **Workload D : Read Latest Workload**
In this workload most recent workload are inserted and the records which are insert are the most popular once. In simple words the most recent recorded are read.

6.3] Software installed

a) **MySQL:** MySQL version 5.7.25-0ubuntu0.18.04.2 was downloaded and installed.
b) **YCSB :** ycsb-0.14.0 was downloaded and installed

c) **Testharness :**

- Testharness was downloaded and installed
- Runtest.sh was modified as per the requirement
- Opcounts.txt file was added with workload IOPS (20k,30k,40k,50k,60k)
- Testdbs.txt was updated with '**jdbc'** for MySQL and **'mongodb'** for mongoDB for test execution.

d) **MongoDB:** MongoDB 3.2.10 version was downloaded and installed.

7] Evaluation and Result

7.1] YCSB –Working

It is an open source and program suit which is basically used for specification and evaluating and retrieval and maintenance of computer program. [8] Also the main feature of YCSB is to compare the performance of NoSQL databases. It consist of YCSB client, workloads and command line parameters.
The Below diagram explain the architecture of YCSB.

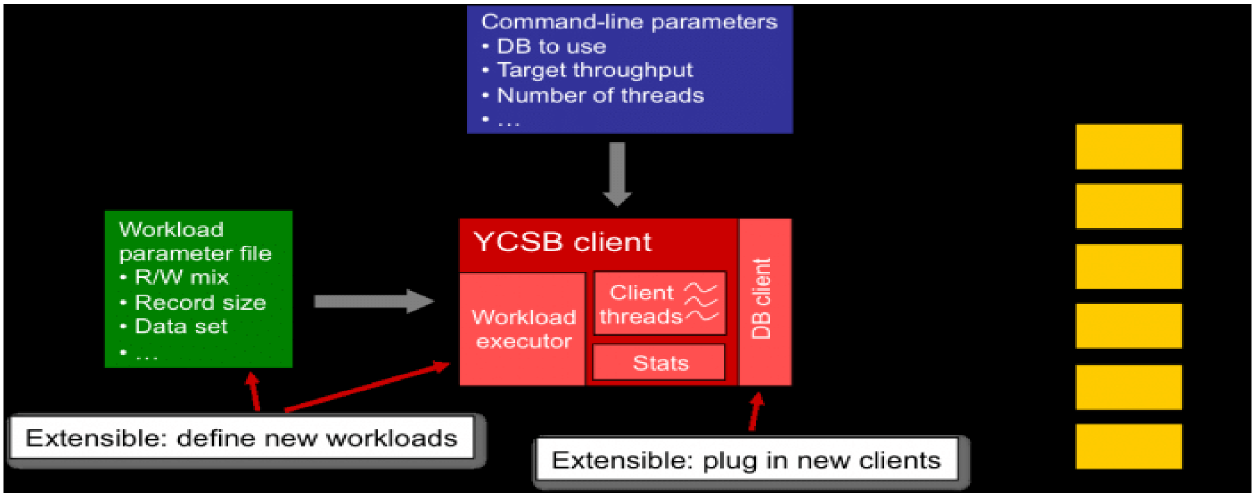

Figure: YCSB Architecture [9]

7.2] Result and analysis:

We have carried out experiments by running different workloads on each databases and we got the below results. Below graphs shows us the result of our experiments on MongoDB and MySQL. We have ran workload A and workload D on each database with 20k , 30k, 40k, 50k and 60k workload.

A. Workload A:

1) Operation Count vs Overall Throughput

The below figure explains the overall throughput (OPS/SEC) for mongoDB and MySQL database. From the output we can say that the overall throughput of mongodb is much greater compare to MySQL.

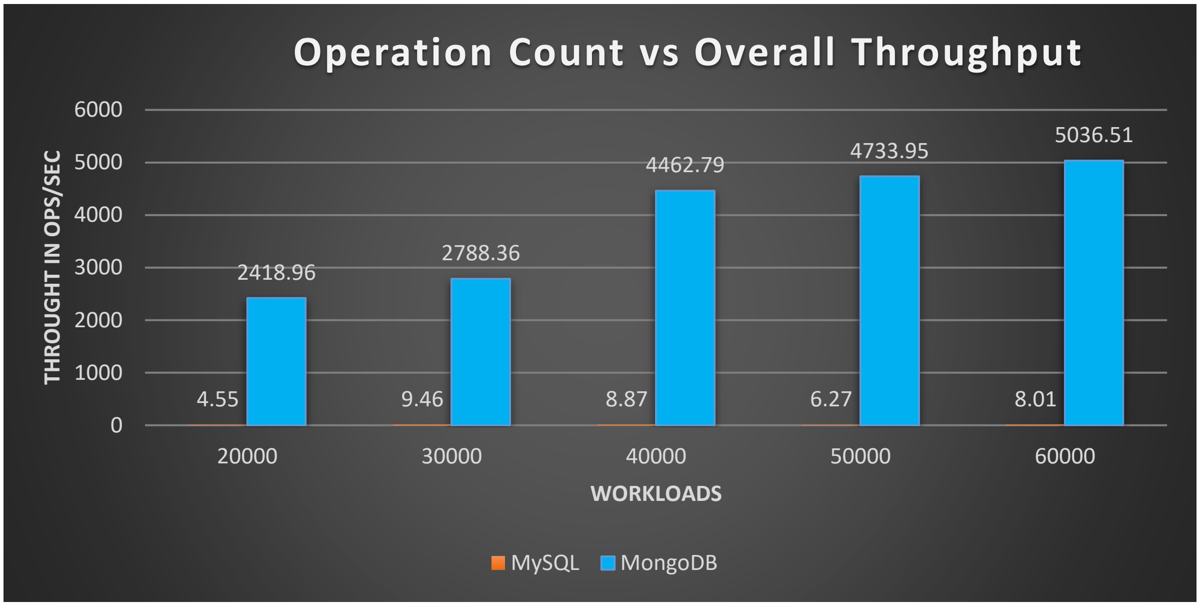

Figure 1: Operation Count (workload) vs Overall throughput

2) Insertion time vs Operation Count

It is clear from the below graph that insertion time taken in MongoDB is much faster as compare to MySQL. To insert a workload of 20000 mongoDB takes only 8.26 sec, compared to MySQL which takes 4329.89sec to insert. We can say that MongoDB is faster.

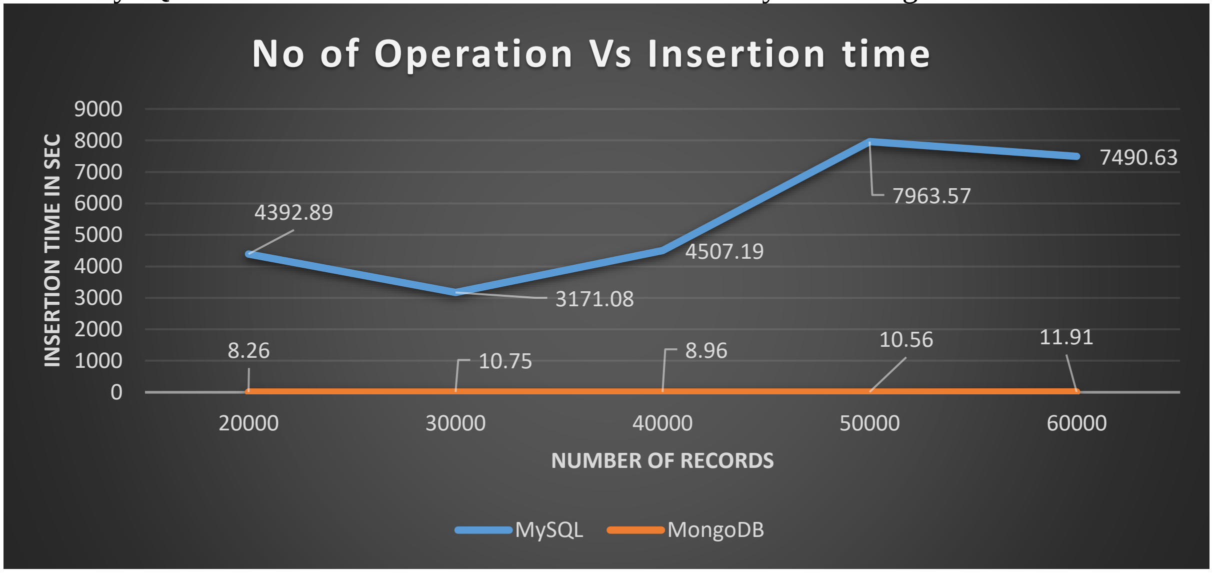

Figure: Number of Operation vs Insertion time in Sec

3) Read Operation vs Read Average Latency

As we know that, workload A is mixture of 50% read and 50% write so we considered read operation as 50% .From below graph, read average latency is pretty high for MySQL as compare to MongoDB for workload A .Therefore we can say that against increasing number of read operation, average latency rate of MySQL is much more higher as compare to MongoDB for workload A.

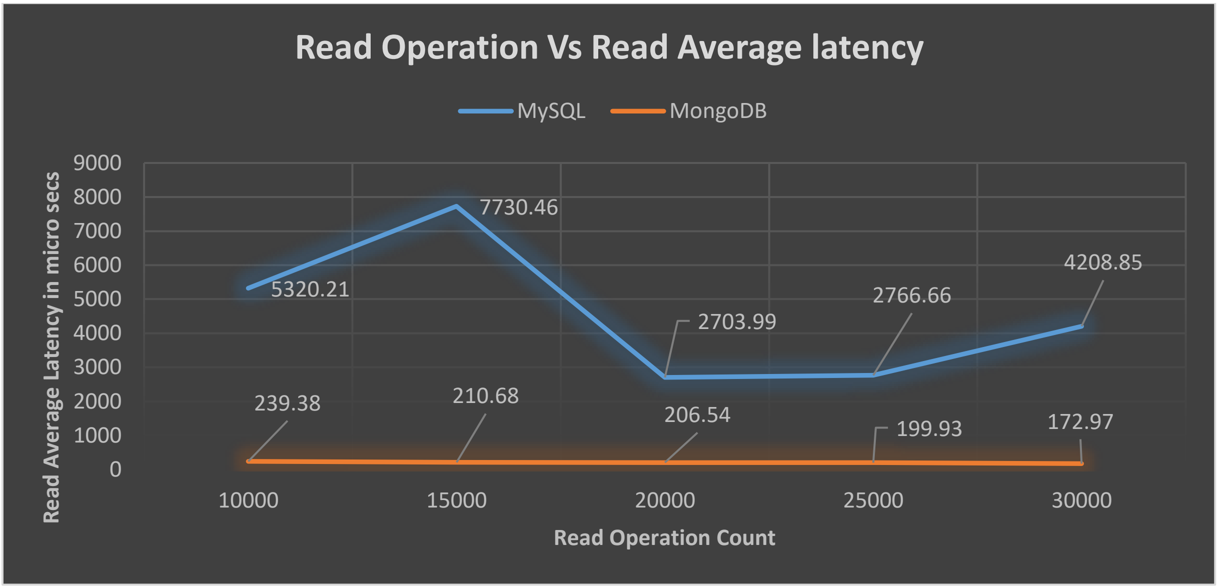

Figure: Read Operation VS Read Average Latency

4) Read 95 Percentile vs Operation Count

The read 95 percentile for read operation shows that the response time for MongoDB is lesser than that of MySQL. We can clearly observe from below diagram.

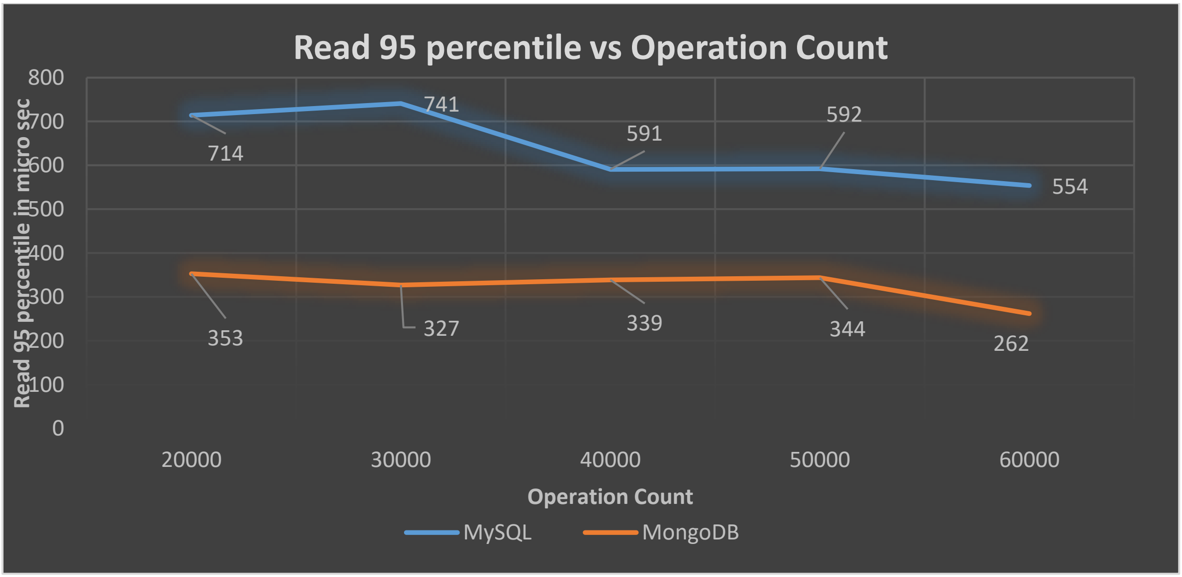

Figure: Read 95 Percentile vs Operation Count

B. Workload D

1) Operation Count vs Overall throughput

Below figure is following representation of workload D.
The graphs represent the overall throughput analysis of MongoDB and MySQL .We can say that the overall throughput for the workload D for mongoDB is much higher as compare to MySQL.

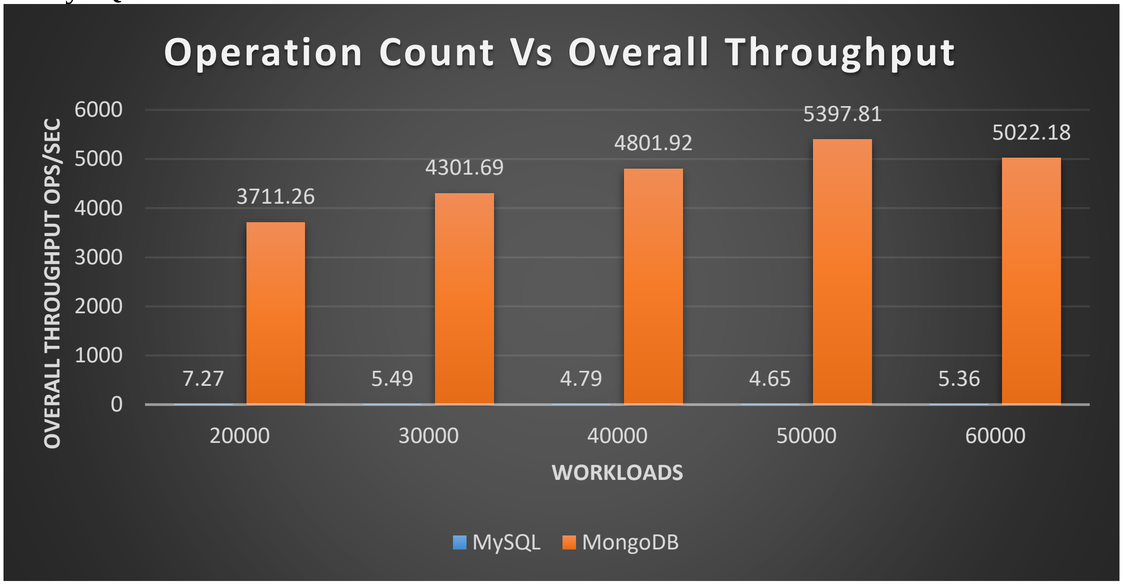

Figure: Overall throughput (ops/sec) vs Operational count

2) Insertion time vs Operation count

Inserting into both the databases is one of the most important things we need to consider as this depends directly on how the application will behave. From the below representation we can say that the time taken to insert the record for MySQL takes much time as compare to MongoDB.

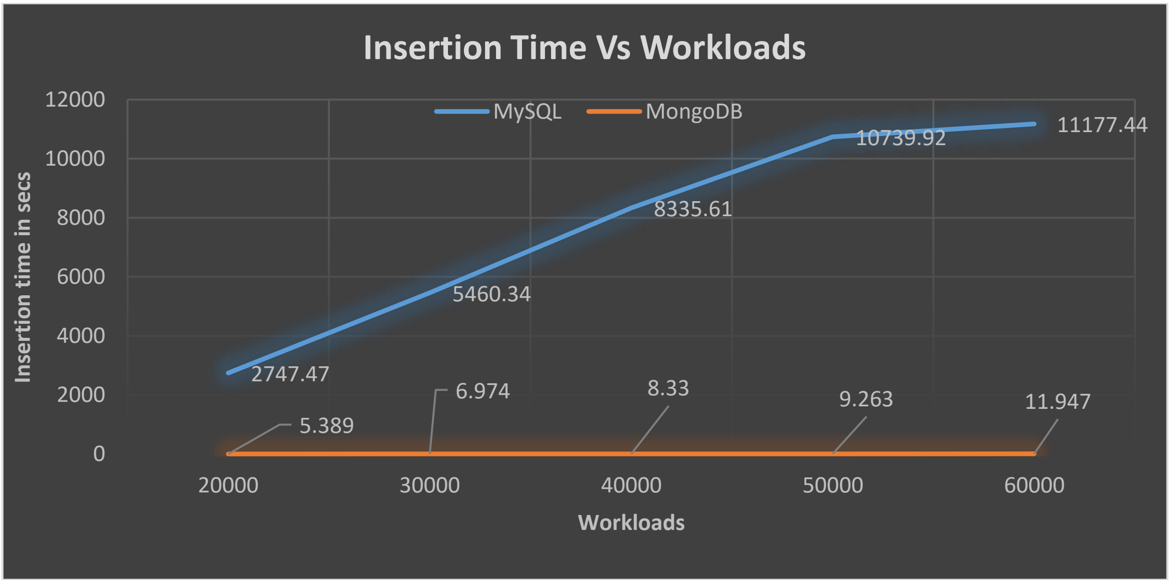

Figure: Insertion time vs Workloads

3) **Read Operation vs Read average latency**

We know that in Workload D most recent inserted records are most popular once. So comparing the read operation we can say that the read average latency for the MySQL is higher as compare to MongoDB.

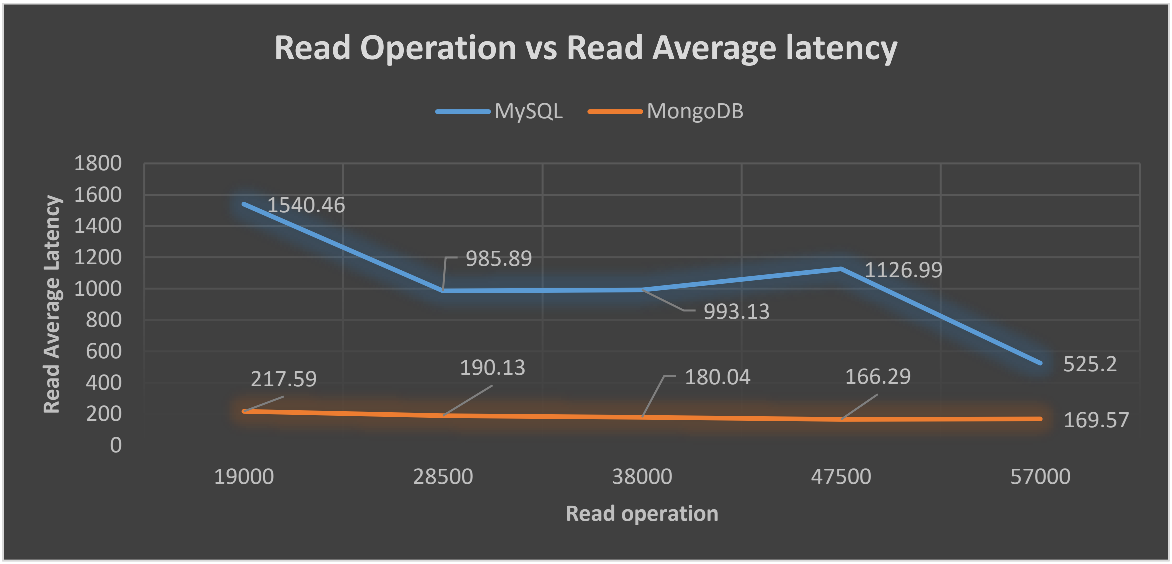

Figure: Read Operations vs Read average latency

4) **Read 95 Percentile**

The below graph represent that the read operation of MongoDB and MySQL and by looking the below graph we can say that there is not much difference between them. Here MySQL has higher value compare to MongoDB.

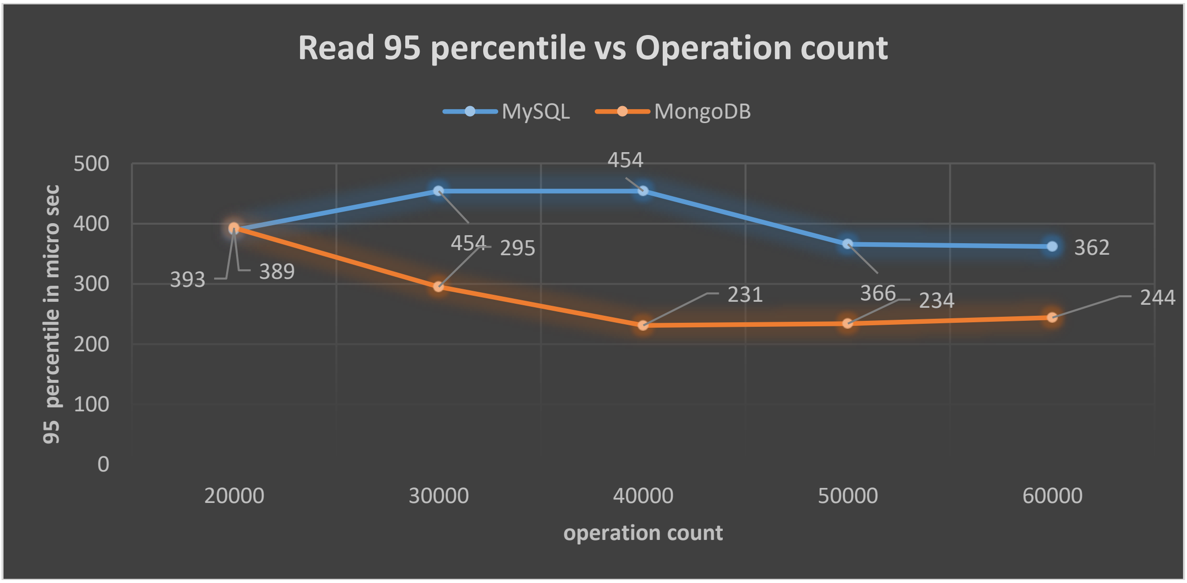

Figure: Read 95 percentile (micro sec) vs Operation count

8] Conclusion and discussion

Transition from traditional databases to the non-relational can be a very challenging task if we consider the architecture compatibility of a system. So there isn't a straight forward answer to the question 'which database is better' as both have their own pros and cons. So, a more detailed study is required in order to compare between these databases. Developers must decide which database is more suitable for a particular application as per their requirements.

We have used YCSB, which is a well-known for testing the performance of databases. On the basis of our findings, we came to a conclusion that MongoDB is much efficient and faster than MySQL.

As a part of our experiment, we loaded different workloads on both the databases to figure out which one gives the better performance in terms of time and it was observed that MongoDB was a clear winner amongst the two. We made use of Workload A and Workload D in order to compare the execution of both databases and our study revealed that when we consider parameters like throughput, insertion time, average latency and a latency of 95 percentile, MongoDB was much more efficient than MySQL. In addition to that, latency rate of MongoDB is much lesser than that of MySQL. So, we can conclude that with the increase in demand and application of unstructured data from various nodes, NoSQL databases has a bright future ahead in the market, though it cannot be denied that it still requires some necessary changes and removal of different limitations to be more application friendly.

After considering all our experiments and assumptions, it is also evident that the era of RDBMS is definitely not coming to an end in near future as it provides distinct set of unparalleled features like providing data integrity and scalability. As far as our experiment was concerned, we preferred MongoDB over MySQL as our application was more data intensive and required storage and processing of a huge chunk of data.

9] References:

1. Dede, Elif & Govindaraju, Madhusudan & Gunter, Dan & Canon, Shane & Ramakrishnan, Lavanya. (2013). A Performance Evaluation of a MongoDB and Hadoop Platform for Scientific Data Analysis. Proc 4th ACM Work Sci cloud Compute. 10.1145/2465848.2465849.

2. What Are the Top MySQL Features?" SearchITChannel,searchitchannel.techtarget.com/feature/What-are-the-top-MySQL-features-What-is-MySQL.

3. "MongoDB Features - Javatpoint." www.javatpoint.com, www.javatpoint.com/mongodb-features.

4. "MySQL Architecture and Layers." Qxf2 Blog, 3 Apr. 2018, qxf2.com/blog/mysql-architecture-and-layers/.

5. Linode, and Linode. "5 Essential Steps to Hardening Your MySQL Database." Medium, Linode Cube, 23 Aug. 2017, medium.com/linode-cube/5-essential-steps-to-hardening-your-mysql-database-591e477bbbd7.

6. Győrödi, Cornelia & Gyorodi, Robert & Pecherle, George & Olah, Andrada. (2015). A Comparative Study: MongoDB vs. MySQL. 10.13140/RG.2.1.1226.7685.

7. Pan, Xiaolong & Wu, Weiming & Gu, Y.H.. (2011). Study and Optimization Based on MySQL Storage Engine. Advances in Intelligent and Soft Computing. 129. 185-189. 10.1007/978-3-642-25986-9_28.

8. "YCSB." Wikipedia,Wikimedia Foundation, 4 Dec. 2018, en.wikipedia.org/wiki/YCSB.

9. Khazaei, Hamzeh & Fokaefs, Marios & Zareian, Saeed & Beigi, Nasim & Ramprasad, Brian & Shtern, Mark & Gaikwad, Purwa & Litoiu, Marin. (2015). How do I choose the right NoSQL solution? A comprehensive theoretical and experimental survey. Journal of Big Data and Information Analytics (BDIA). 2. 10.3934/bdia.2016004.

10. "MySQL Transaction: START TRANSACTION, COMMIT & ROLLBACK by Examples." MySQL Tutorial, www.mysqltutorial.org/mysql-transaction.aspx.

11. "Transactions"TransactionsMongoDBManual,docs.mongodb.com/manual/core/transactions/