MONGODB VS MYSQL: A COMPARATIVE STUDY OF PERFORMANCE IN SUPER MARKET MANAGEMENT SYSTEM

Dipina Damodaran B, Shirin Salim and Surekha Marium Vargese

Department of Computer Engineering,M A College of Engineering, Kothamangalam, Kerala, India

ABSTRACT

A database is information collection that is organized in tables so that it can easily be accessed, managed, and updated. It is the collection of tables, schemas, queries, reports, views and other objects. The data are typically organized to model in a way that supports processes requiring information, such as modelling to find a hotel with availability of rooms, thus the people can easily locate the hotels with vacancies. There are many databases commonly, relational and non relational databases. Relational databases usually work with structured data and non relational databases are work with semi structured data. In this paper, the performance evaluation of MySQL and MongoDB is performed where MySQL is an example of relational database and MongoDB is an example of non relational databases. A relational database is a data structure that allows you to connect information from different 'tables', or different types of data buckets. Non-relational database stores data without explicit and structured mechanisms to link data from different buckets to one another. This paper discuss about the performance of MongoDB and MySQL in the field of Super Market Management System. A supermarket is a large form of the traditional grocery store also a self-service shop offering a wide variety of food and household products, organized in systematic manner. It is larger and has a open selection than a traditional grocery store.

KEYWORDS

Relational database, MySQL, MongoDB, Super Market Management System

1. INTRODUCTION

The relational database has been the foundation of enterprise applications for decades, and when MySQL is released in 1995 it has been popular and in expensive option. Due to the explosion of large volume and variety of data's in recent years, non-relational database technologies like MongoDB become useful to address the problems faced by traditional databases. MongoDB is very useful for new applications as well as to augment or replace existing relational infrastructure.

MySQL is a popular open-source relational database management system (RDBMS) that is distributed, developed, and supported by Oracle Corporation. The relational systems like, MySQL stores data in tabular form and uses structured query language (SQL) for accessing of data. In MySQL, the programmer should pre-define the schema based on requirements and set up rules to control the relationships between fields in the record. The related information may be stored in different tables, but they are associated by the use of joins. Thus, data duplication can be minimized.

MongoDB is an open-source database developed by MongoDB Inc. MongoDB stores data in JSON-like documents that can vary in structure. Related information can be stored together for fast query access through the MongoDB query language. MongoDB uses dynamic schemas, which helps to create records without first defining the structure, such as the attributes or the data types. It is possible to change the structure of records by simply adding new attributes or deleting existing fields. This model helps to represent hierarchical relationships, to store arrays, and other more complex structures very easily. Documents in a record need not have an identical set of fields. MongoDB is designed with high availability and scalability includes replication and auto-sharding. In this paper, we perform a comparison on both MySQL and MongoDB on the platform of supermarket.
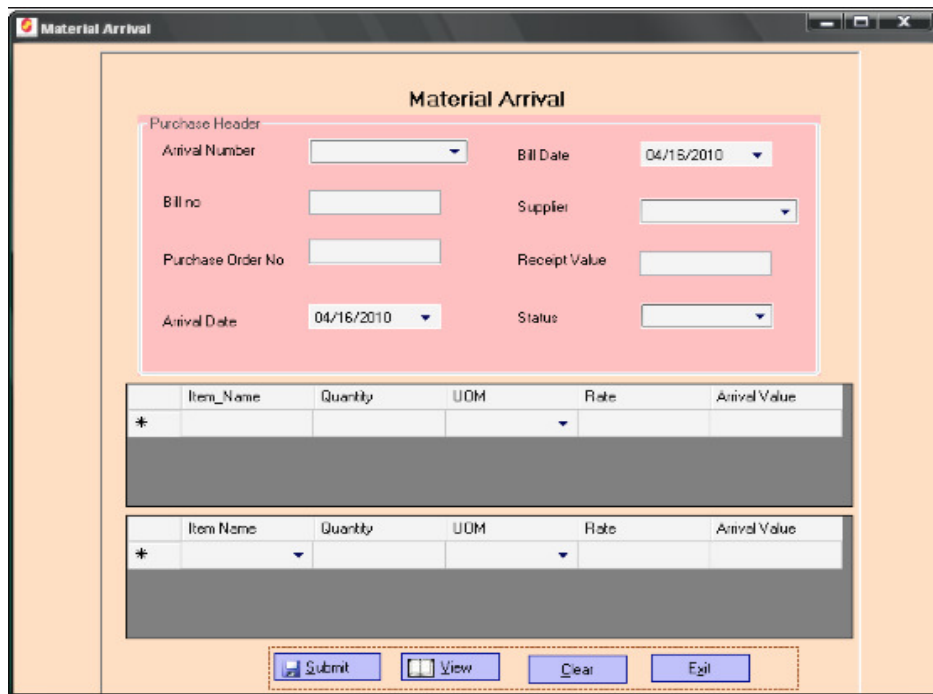

Figure 1.Example of Super Market Management System Home page

The "Supermarket Management System "which manages the sales activity in a supermarket, maintaining the records of stock details, maintaining the records of the sales done for a particular month/year etc. Thus users will consume less time for calculation and the sales activity can be completed within a fraction of seconds whereas manual system will make the employ to write it down the cost of each item and perform calculation which is a long procedure and it also needs a lot of time. By using this system paper work can be reduced and the user can spend extra time for monitoring the supermarket. MongoDB is more applicable to large databases but for the simplicity we take supermarket data.

2. PROBLEM DEFINITION

This section gives a brief definition on MySQL and MongoDB. Then evaluate the performance of both the databases on the application of hypermarket. When compared to MySQL it is observed that MongoDB is much better in query processing [9][12]. The MongoDB database consists of a set of databases in which each database contains multiple collections. Because MongoDB

operates with dynamic schemas, every collection might contain different types of datas. Every object also called as documents is represented by a JSON structure: a list of key value pairs. The value can be of mainly three types: a primitive value, an array of documents or a list of key-value-pairs. For to query these objects, the client can set the collections expressed as a list of key value pairs. It is also possible to query nested fields. The queries are also JSON like structured; hence a complex query can take much more space than the same query for the relational databases. If the built-in queries are too limited, it is possible to send JavaScript logic to the server for more complex queries.

MongoDB supports mainly two types of replication: master-slave and replica sets. In the master-slave replication, the master has control of full data access and which writes every change to its slaves. The slaves can only possible to read data. Replica sets works same as master-slave replications, but it is possible to select a new master if the original master become down. Another important feature that supported by MongoDB is automatic sharding. Using this feature data can be partitioned to different nodes. The administrator has to verify a sharding key for each collection which defines how to partition the documents. In such an environment, the clients connect to a special master node called mongos process which analyses and redirects the query to the appropriate node or nodes. To eliminate data losses, every logical node contain physical servers which act as a replica set. Using this infrastructure it is also possible to use Map/Reduce having a very good performance.

2.1 Architecture

MongoDB supports standalone or single instance operations. The replica sets provide high performance of replication with automated failure handling, while sharded clusters make it possible to divide large data sets over different machines which are transparent to the users. MongoDB users combine replica sets and sharded clusters to provide high levels of redundancy of data sets which are transparent for applications [7] . MongoDB scales horizontally by using sharding. The key called shard key is chosen by user, which determines how the data in a collection will be distributed. The data is split into several ranges (based on the shard key) and distributed across different shards. A shard is a master with one or more slaves. Alternatively, the shard key can be hashed to map to a shard enabling an even data distribution.
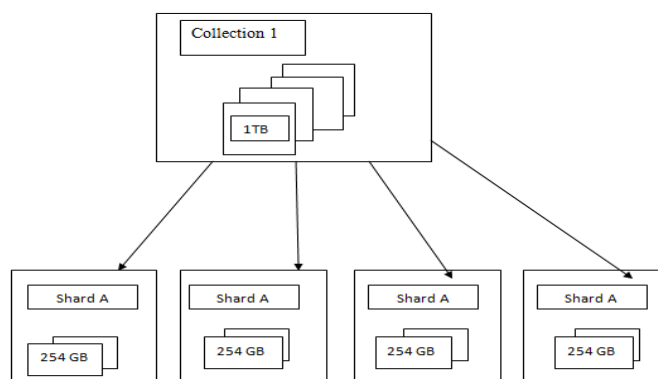

Figure 2.Deployment Architecture

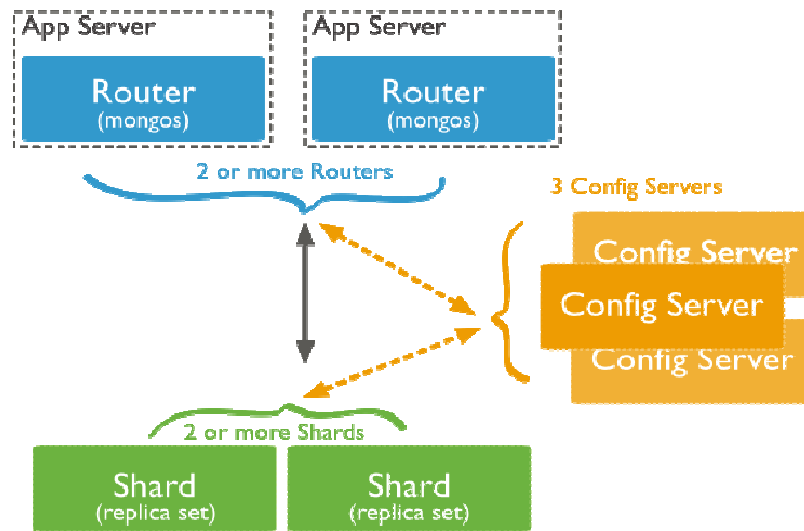MongoDB supports sharding through the configuration of a *sharded clusters*.

Figure 2.Sharding in Mongodb

- Sharded cluster has the following components: shards, query routers and config servers.
- **Shards** store the data. To provide high availability and data consistency, in a production sharded cluster, each shard is a replica set For more information on replica sets, see Replica Sets.
- **Query Routers** interface with client applications and direct operations to the appropriate shard or shards. The query router processes and targets operations to shards and then returns results to the clients. A sharded cluster can contain more than one query router to divide the client request load. A client sends requests to one query router. Most sharded clusters have many query routers.
- **Config servers** store the cluster's metadata. This data contains a mapping of the cluster's data set to the shards. The query router uses this metadata to target operations to specific shards. Production sharded clusters have *exactly* 3 config servers.

3. METHODOLOGY

Organizations of all sizes commonly adopting MongoDB because it enables them to build applications which are faster, handle highly diverse types of data's, and manage applications more efficiently at scale. MongoDB documents map naturally to modern, object-oriented programming languages. MongoDB removes the complex object-relational mapping (ORM) layer which translates the objects in code to relational tables. MongoDB's flexible data model helps the database schema can evolve with business requirements. For example to add a single new field to Craiglist's MySQL database, it would take months to execute. The Craigslist team moved to MongoDB because it helps to accommodate changes to the data model without costly schema migrations.

MongoDB can scale within and across multiple distributed data centers, providing new levels of scalability and availability which are unachievable with relational databases like MySQL. As your deployments grow in terms of data volume and throughput, MongoDB scales easily without much downtime, and without changing the application. But, to achieve scale with MySQL, it often requires significant, custom engineering work. While modern applications require a flexible and scalable system like MongoDB, there are use cases for which a relational database like

MySQL are better suited. MongoDB is not a drop-in replacement for legacy applications built around the relational data model and SQL.

A concrete example would be the booking of tickets behind a travel reservation system, which also involves complex transactions. While the core booking system might run on MySQL, those parts of the app that system with users – serving booking, integrating with social networks, managing sessions – would be better when placed in MongoDB. MongoDB came with the aim of giving the new way of data storage. Therefore database provide storage of document for the World Wide Web. Began in 2007, MongoDB is built to store data in a dynamic schema, instead of a tabular representation like SQL. The data in MongoDB is stored in the form of object notation based on the format of JSON (Java Script Object Notation). To data transfer over the network between the server and web application which use human readable format the standard method is using of JSON. Prior to JSON, the XML was used for that purpose. MongoDB modified the JSON format into its own BSON, which stores the object as a binary format. BSON stands for Binary JSON. Due to its binary format provide more reliable and efficient in the area of storage space and speed.

4. EXPERIMENTAL RESULTS

The results of experiments performed to test various aspects of the implementation employed in hypermarket are provided in this section i.e., using the insertion and search operations on databases for auditing purposes .The various operations are performed on the two databases and we obtain the below results.

Operations	No.of Records	Execution Time (in ms)	
		MongoDB	MySQL
INSERTION	100	0.01	0.01
	1000	0.5	1.25
	10000	1.2	2.2
	25000	2.25	3
SEARCH	100	0.05	0.152
	1000	0.12	1.52
	10000	0.55	4.47
	25000	1.25	5.21

Table 1 for insertion and searching operations

The performance of MongoDB while comparing with SQL by performing two operations, Insertion and Searching. A large no of records were taken and performed the operations in both databases. The graph plotted based on the performance is shown below.
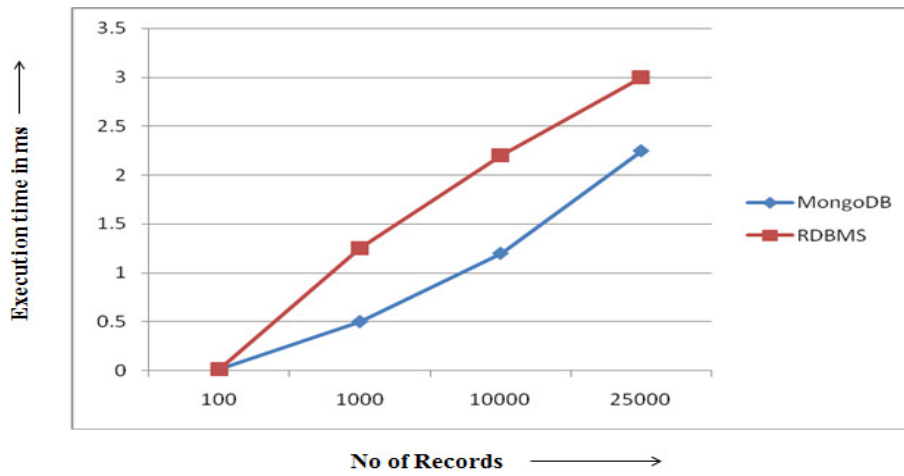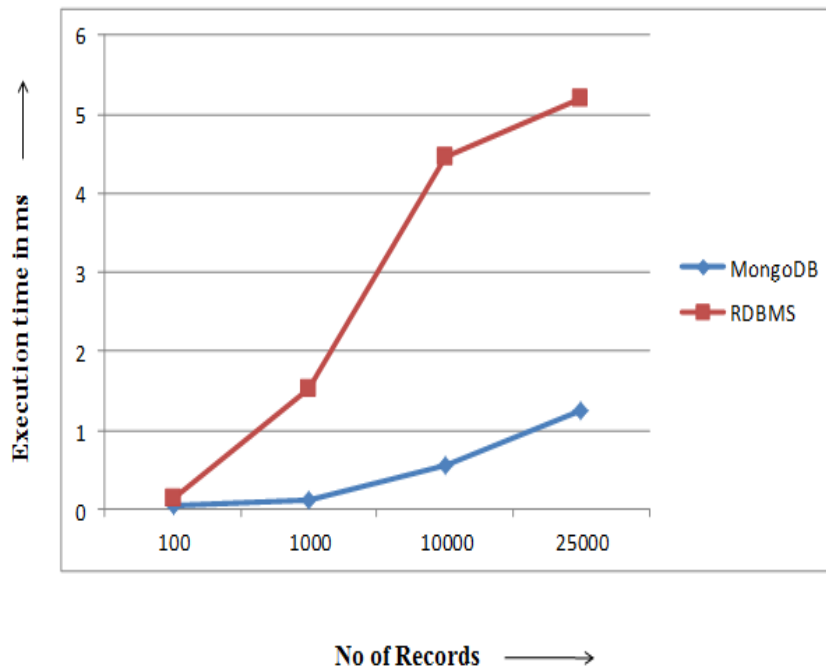
Figure 3. Insertion operation

Figure 4.Searching operation

5. PERFORMANCE EVALUATION

On analysing the performance of MySQL and MongoDB databases on hypermarket application, the performance of MongoDB is more when compared to that of MySQL. Organizations of all sizes are commonly adopting MongoDB because it enables them to build applications faster, handle highly diverse types of data, and manage applications more efficiently at large scale. Development is simplified because MongoDB documents map naturally to modern, object-oriented programming languages. Using MongoDB, it removes the complex object-relational mapping (ORM) layer that translates the objects in code to relational tables. MongoDB's flexible data model helps that the database schema can evolve with business requirements.

One of the most important drawbacks of relational databases is that each item can only contain single attribute. Consider a bank example; a customer's relationship with a bank is stored as different row items in separate tables. So each customer's master details are stored in one table, the account details of those customers are in another table, the loan details in yet another table, investment details are in a different table, and so on. But these tables are connected to each other by use of relations like primary keys and foreign keys. Non-relational databases, use key-value stores or key-value pairs, are different from this model. Key-value pairs provide possibility to store several related items in one "row" of data in the same table. For instance, in a non-relational table for the same bank example, each row can store the customer's details as well as their account details, loan and investment details. All data relating to one customer can conveniently stored as one record. This implies an obviously superior method for storing of data, but it has a major limitation: key-value pairs, unlike relational databases, it cannot use relationships between data items. In key-value databases, the customer details like (name, social security, address, account number, etc.) are stored in one data record (instead of stored in several tables, as in the relational model). The customer's transaction details (account withdrawals, account deposits, loan repayments, etc.) would also be stored as another single data record.

6. DISCUSSION

MongoDB is widely used in the field of large databases. One of the most important advantages is its scalability. MongoDB follows BASE transaction, Basically Available Soft State and Eventual consistency .Another important feature is handling of failures. For the simplicity we conduct an analysis based on super market. But MongoDB is more suitable for other applications having large volume of data where data need high security. Since it is schema less, it supports different types of data.

7. CONCLUSION

In this paper, we undergo performance evaluation between MySQL and MongoDB on hypermarket application. For evaluating its performance execution time is considered. We came to a conclusion that when number of records inserted or searched is smaller, there is no difference in the execution time taken for each of these operations to complete for both MongoDB and MySQL databases. However, when number of records is increased, MongoDB shows significant reduction in the time taken for execution compared to MySQL. Thus, when the number of records is higher, MongoDB takes less time compared to MySQL. MongoDB can be preferred for better performance.

So in summary, RDBMS's suffer from no horizontal scaling for high transaction loads (millions of read-writes), while NoSQL databases solve high transaction loads but at the cost of data integrity and joins.

REFERENCES

[1] K. Sanobar, M. Vanita, "SQL Support over MongoDB using Metadata", International Journal of Scientific and Research Publications, Volume 3, Issue 10, October 2013

[2] https://www.mongodb.org/about/introduction/

[3] S. Hoberman, "Data Modeling for MongoDB", Publisher by Technics Publications, LLC 2 Lindsley Road Basking Ridge, NJ 07920, USA, ISBN 978-1-935504-70-2, 2014.

[4] http://dwhlaureate.blogspot.in/2013/10/features-of-mongo-db.html

[5] R. P Padhy, M. R. Patra, S. C. Satapathy, "RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database's", International Journal of Advance Engineering Sciences and Technologies, Vol. 11, Issue No. 1, 015-030, 2011.

[6] https://en.wikipedia.org/wiki/MongoDB

[7] https://en.wikipedia.org/wiki/shard-(database-architecture)

[8] Z. Wei-Ping, LI Ming-Xin, H. Chen, "Using MongoDB to Implement Textbook Management System instead of MySQL", IEEE 3rd International Conference on Communication Software and Networks (ICCSN), ISSN 978-1-61284-486, 2011.

AUTHORS

Dipina Damodaran B completed her B.Tech degree from Malabar College of Engineering and Technology,Trissur in 2013 which is affiliated to Calicut University. She presented paper on National Level Conference. She is currently pursuing M.Tech in Computer Science in Computer Science and Engineering in Mar Athanasius College of Engineering. Her areas of research are Modern Databases, Data Structure and Data Mining.

Shirin Salim currently pursuing M.Tech in Computer Science and Engineering   in Mar Athanasius College of Engineering. She completed her B.Tech degree from Ilahia College of Engineering in 2014 which is affiliated to Mahatma Gandhi University. She presented paper in National Conference. Her areas of research are Modern Database, Data Mining and Machine Learning.

Surekha Mariam Varghese is currently heading the Department of Computer Science and Engineering, M.A. College of Engineering, Kothamangalam, Kerala, India. She received her B-Tech Degree in Computer Science and Engineering in 1990 from CET affiliated to Kerala University and M-Tech in Computer and Information Sciences from CUSAT, Kochi in 1996. She obtained Ph.D in Computer Security from CUSAT, Kochi in 2009. Her research interests include Network Security, Database Management, Data Structures and Algorithms, Operating Systems and Distributed Computing. She has published 17 papers in international journals.