IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 7, July 2021

DOI 10.17148/IJARCCE.2021.10705

A MECHANISM FOR MAPPING RELATIONAL DATABASE (SQL) SCHEMA TO NOSQL SCHEMA AND QUERY REWRITING OF SQL TO NOSQL QUERIES.

Chinwendu. Amadi[1], Bennett. O. E.[2], Nwiabu. N. D[3]

Department of Computer Science,

Rivers State University, Port Harcourt, Nigeria

Abstract: The huge amount of streaming data available nowadays are due to massive use of new technologies. Such tremendous amounts of data add a great deal of challenges to the traditional relational database paradigm. The challenges are: performance in reading data and scalability (that is the ability of a database to handle changing demands by adding/removing resources). However, the use of NoSQL database in new enterprises is not a major issue because the new application design will be based on NoSQL database. But the problem appears when the existing systems that relay on relational database are restructuring their systems to implement NoSQL database. They need to reanalyze the system requirements to build up the new database schema. This study aims at developing a mechanism for mapping relational database schema to NoSQL schema and query rewriting of SQL to NoSQL queries. Conceptual mapping has been used which involved the concepts of a relational structure (Table, Simple Attribute, Primary Key (PK), Foreign Key (FK)), and NoSQL concept (Document Collection, Document, Field, Embedded Field, Field List, ObjectId, DBRef). Relationships were realized by defining object references between objects belonging to documents. Mapping algorithm used the metadata stored in the MySQL system tables. The system was implemented in PHP. Experiments were conducted on MySQL and MongoDB based on migration speed and query speed. The read speed of each database was tested, four queries were conducted to pull the number of rows from the product table. The average of these five queries were taken for each implementation. MongoDB proved the faster database management system with a time of 1.78 seconds as compared to 3.35 seconds for MySQL.

1. INTRODUCTION

Database Management System (DBMS) is software designed to assist in maintaining and utilizing large collections of data. Database management systems have been around since the early 1960s. In the 1980s, the relational model consolidated its position as the dominant DBMS paradigm, and database systems continued to gain widespread use. The Structured Query Language (SQL) query language for relational databases, developed as part of International Business Machines (IBM's) System R project, is now the standard query language. Advances have been made in numerous areas of database systems. Large vendors began extending their database systems at this time so they could store new data types into their systems such as written text and images, and also gain access to more complex queries [1]. Relational Database (RDB) stores data into tables in a normalized and structured form, which has become a limitation for RDB with a fast evolution of applications. It was invented by Edgar F. Codd in 1970, the relational database arranges data into different rows and columns by associating a specific key for each row. Almost all relational database systems use Structured Query Language (SQL) and are remarkably complex. They are traditionally more rigid or controlled systems and have a limited or restricted ability to translate complex data such as unstructured data. That said, SQL systems are still used extensively and are quite useful for maintaining accurate transactional records, legacy data sources, and numerous other use cases within organizations of all sizes. In the mid-1990s, the internet gained extreme popularity, and relational databases simply could not keep up with the flow of information demanded by users, as well as the larger variety of data types that occurred from this evolution. This led to the development of non-relational databases, often referred to as Not Only SQL (NoSQL). NoSQL databases can translate strange data quickly and avoid the rigidity of SQL by replacing "organized" storage with more flexibility. Relational Database (RDB) cannot deal with un-normalized data and massive size, which makes companies like Google, Facebook, and Amazon choose NoSQL database as the option of their data storage [2]. Relational databases are designed to scale up. This means that the scaling ability is limited for relational databases to 10 terabytes or 100 terabytes. NoSQL databases can reach up to 1000 terabytes by just adding more boxes parallel to each other.

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

Vol. 10, Issue 7, July 2021

DOI 10.17148/IJARCCE.2021.10705

2. RELATED WORK

Serrano et al. [3] proposed relations to multi-dimensional maps. They used four steps for mapping the one-to-one and one-to-many relationships from RDB into HBase. Initially, the data is converted to a de-normalized level, followed by merging adjacent tables, then a row key is determined to be optimal for various accessing patterns, and finally the indexing on the HBase tables are maintained. Freitas et al. [4] presented the transformation of RDB to several NoSQL families namely keyvalue, column, document, and graph. The authors identified the concepts of each database using defined tuples. Following that, algorithms to conduct the transformation are presented, and a case study is used as a proof of concept. This work is complete in the sense that it covers all NoSQL families. However, it is not clear that all relationship types in RDB are included. Liao et al. [5] presented a data adapter used to query and map between SQL and NoSQL databases. The adapter enables query from application and deals with database transformation at the same time. While this work implements the data adapter, it does not provide clear transformation rules between two different databases. Liyanaarachchi et al. [6] used a standalone application named MigDB that analyzes tables in RDB, creates a JSON file based on the tables, and then passing the JSON file to a neural network. Moreover, the network made a decision on the most appropriate structure to map the JSON file, whether it will be an embedding or referencing structure. This work was done for mapping association relationships only. Chen and Lee [7] proposed data conversion from RDB to HBase. They mapped the one-to-one, one-to-many and many-to-many association relationships in RDB by two steps. In the first step, the row key for column families is chosen based on the expected entry pattern of users. Following that, in the second step the table that had a relationship with another table becomes merged into one super column family. Wardani and Kiing [8] mapped the one-to-many relationship from RDB to graph-based NoSQL. The starting node in the graph is the many side and the primary key of the one side is inserted into the many side by keeping the primary key as an edge property. The join table in RDB is not used by keeping the information as a relationship property. While mapping the ternary relationship, the join table and the foreign keys of other tables were deleted, but the attributes of the relationship were kept as the relationship property between nodes on the graph.

3. DESIGN METHODOLOGY

Architecture specifies the structure, views and action of a system. The proposed system maps relational database to non-relational database. Figure 3.2 shows the SQL and Mongo mapping mechanism.

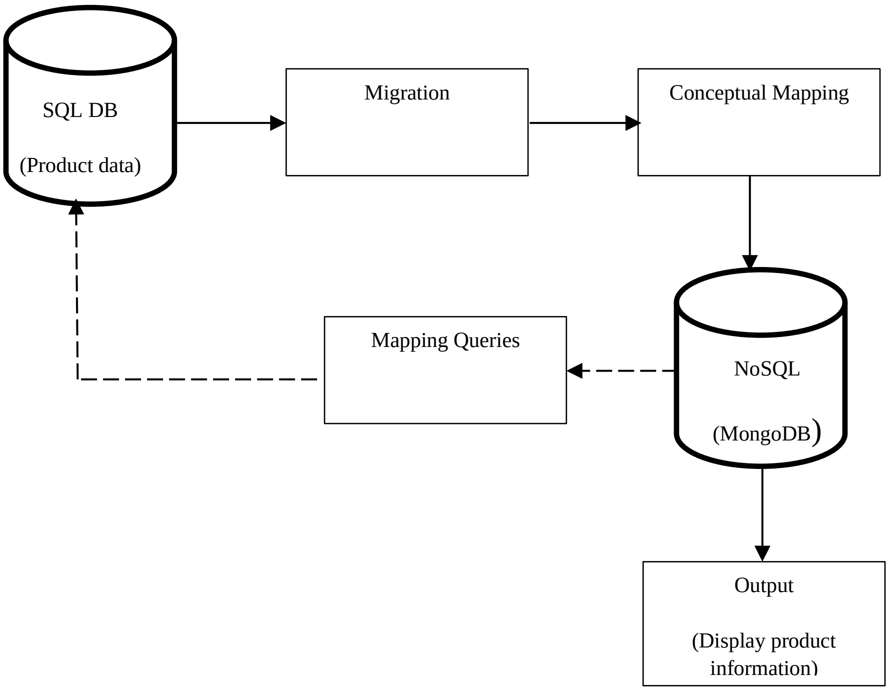

Figure 3.2: Architecture of the Proposed System

The Mapping framework consists of a front end GUI and back end to move data stored in existing MySQL to MongoDB. SQL database comprises of table, views and indexes. CSV file is an external database made up of columns and rows called table. Conceptual mappings enhance the data conversion process. The MongoDB consists of document shape, content and embedded text. The user enters a query in MySQL format, the data is converted to CSV and the query is mapped to MongoDB query format.

SQL: SQL is an example of relational database, which stores data into tables in a normalized and structured form.

Migration: The data migration is responsible for identifying automatically all elements from the original database (example tables, attributes, relationships, indexes).

Conceptual Mapping: This is the process of transforming requests and results between the conceptual or external levels.

NoSQL MongoDB: MongoDB is an example of document-based NoSQL. Data in document-based NoSQL is stored as document, each of which is identified by a particular key. The document is grouped into collections that are sequentially stored, with new documents able to be appended into the collection at any time.

Database Design

The various database tables that will be used in implementing the system are described Table 3.7 to Table 3.9. Any organized system of information storage and retrieval is technically a database. Whether that data is on computer or file cabinet. But for this study, the major concern is to develop a computerized relational database and map into Non-relational database. The term database also implies to a series of related properties which include: data sharing, data integrity, data security, data abstraction and data independence.

MySQL Database Schema

MySQL Database System will be used to store information about the ecommerce system.

Table 3.1: Customer Details in MySQL

S/N	Field Name	Data Type	Description
1	Customer ID	int	Primary key
2	Customer name	varchar	
3	Password	varchar	

4. RESULT AND DISCUSSION

Here we seek to do an experiment that will compare performance differences between MongoDB and MySQL. The categories for comparison are both migration speeds for database schema between the MySQL and MongoDB and their read-speeds. Firstly, with the migration, the speeds are measured as the schema's data is passed through CSV files from one MySQL to MongoDB. Then an experiment is conducted to see how fast each database management system can read data in the schema and return it to the user. It is expected that these experiments show the strengths of each database and how MongoDB can lead to an improvement in performance over the current MySQL implementation for ecommerce data. While implementations of MongoDB often have multiple nodes or clusters, this experiment was run using a single MongoDB node to compare direct performance to MySQL. The migration portion of our experiment consisted of migrating a total of 5 tables from MySQL to MongoDB. Table 4.1 shows a list of tables included in the migration process and the size of each table, the total size of MySQL tables is 36.352MB.

Table 4.1: Product Database

Tables	Size of Data (Kb)
Product	22000
Purchase	9000
Online_retail I	4000
Online_retail II	872
Buyer	480

.

DOI 10.17148/IJARCCE.2021.10705

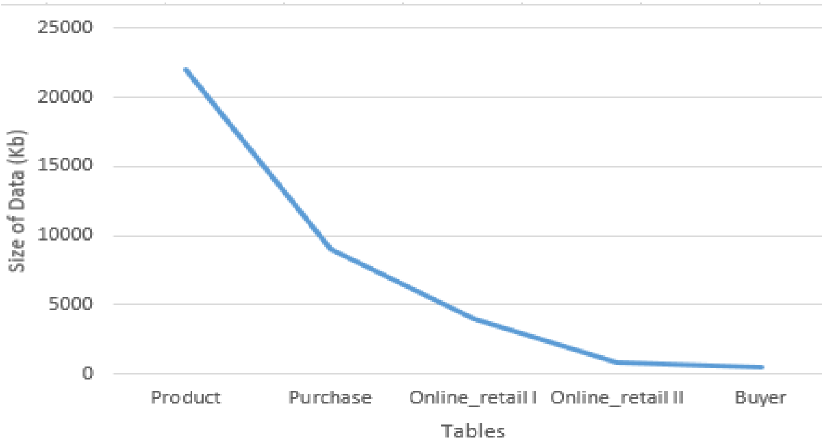

Figure 4.1: Graphical Representation of Table sizes in Database

Table 4.2: Migration Speed

Database	**Speed**
MySQL	6
MongoDB	1.1

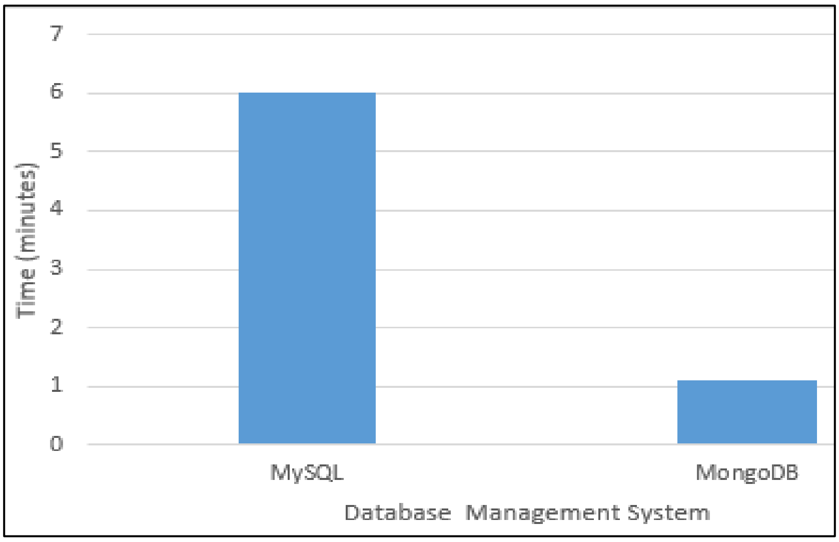

Figure 4.2: Graphical Representation of MySQL and MongoDB migration Speed

Table 4.3: Query Speed

Database	**Query Speed (Second)**
MySQL	1.78
MongoDB	2.35

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 7, July 2021

DOI 10.17148/IJARCCE.2021.10705

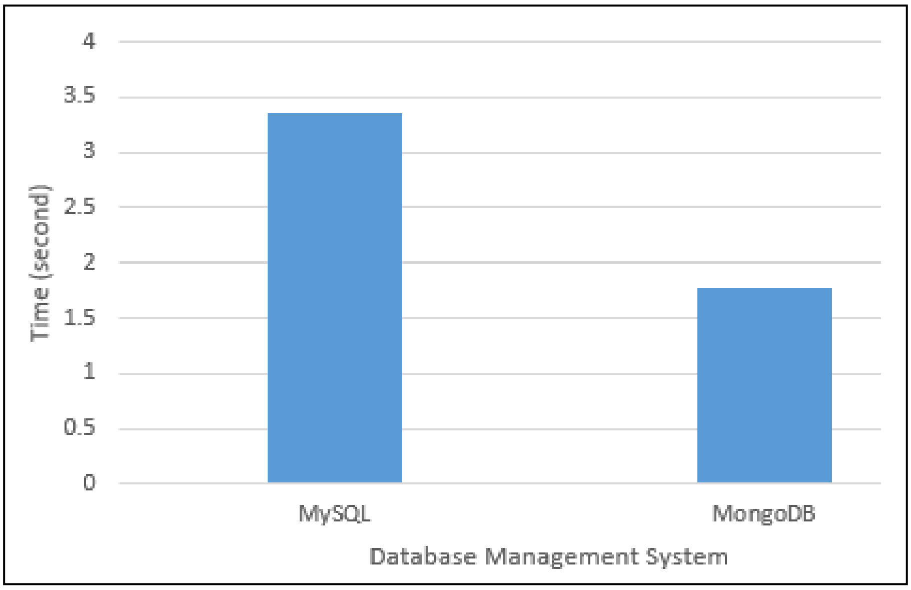

Figure 4.3: Graphical Representation of MySQL and MongoDB Query Speed

CONCLUSION AND FUTURE SCOPE

To meet the growing demands of currently used application of managing a huge amount of data in an efficient manner has emerged the necessity of schema-less NoSQL databases which are capable of handling large amount of data and makes the access to this data easy and efficient. All the data resides mostly on the traditional databases in the form of tables which is not suitable for real-time applications and social media sites etc. and this data needs to be migrated from Relational to NoSQL databases. A number of models, framework and layers have been proposed in recent years to migrate the existing data which is stored in relational databases and queries in MySQL to be appropriately executed without emerging the need of changes in the underlying application code. This study gives a mechanism for mapping relational database schema to NoSQL schema and query rewriting of SQL to NoSQL queries. Conceptual mapping was used which involves the concepts of a relational structure (Table, Simple Attribute, Primary Key (PK), Foreign Key (FK)), and NoSQL concept (Document Collection, Document, Field, Embedded Field, Field List, ObjectId, DBRef). Relationships are realized by defining object references between objects belonging to documents. Thereby, queries may take into account these references to get related objects information. The algorithm uses the metadata stored in the MySQL system tables. It takes into consideration the concepts from Entity-Relationship (ER) model: entity type represented by a relation in the Relational Model. The output of the algorithm that was tested on small size databases (5 tables with many relationships and above 10000 records).We have researched both relational and NoSQL databases and compared their migration and query speeds. We have worked alongside with ecommerce data with migrating a total of 5 tables containing more than 10000 records. This study only gives mapping output for MongoDB so the work can be enhanced for other NoSQL databases such as experiments on Cassandra database.

REFERENCES

[1] Bazar, C. and Iosif, C. S. (2014). The Transition from RDBMS to NoSQL, A Comparative Analysis of Three Popular Non-Relational Solutions: Cassandra, MongoDB and Couchbase. *Database Systems Journal*, 5(2), 49–59.

[2] Rocha, L., Vale, F., Cirilo, E., Barbosa, D. and Mourão, F. (2015). A Framework for Migrating Relational Datasets to NoSQL. *Procedia Computer Science*, 51, 2593–2602.

[3] Serrano, D., Han, D. and Stroulia, E. (2015) From relations to multi-dimensional maps: Towards an SQL-to HBase transformation methodology. In *Proceedings of the IEEE 8th International Conference on Cloud Computing (CLOUD)*, 81–89

[4] Freitas, M. C., Souza, D. Y. and Salgado, A. C. (2016) Conceptual Mappings to Convert Relational into NoSQL Databases. In *Proceedings of the 21st International Conference on Enterprise Information Systems (ICEIS)*, 174–181.

[5] Liao, Y. T., Zhou, J., Lu, C. H., Chen, S. C., Hsu, C. H., Chen, W., Jiang, M. F. and Chung, Y. C. (2016). Data adapter for querying and transformation between SQL and NoSQL database. *Future Generational Computer System*, 65, 111–121.

[6] Liyanaarachchi, G., Kasun, L., Nimesha, M., Lahiru, K. and Karunasena, A. (2016). MigDB-relational to NoSQL mapper. *In Proceedings of the IEEE International Conference on Information and Automation for Sustainability (ICIAfS)*, 1–6

[7] Chen, J. K. and Lee, W. Z. (2017). Data conversion from RDB to HBase. *In Proceedings of the IEEE 8th International Conference on Awareness Science and Technology (iCAST)*, 170–175.

[8] Wardani, D. W. and Kiing, J. (2014) Semantic mapping relational to graph model. In *Proceedings of the IEEE International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*, 160–165.