IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 18, Issue 3, Ver. IV (May-Jun. 2016), PP 13-17
www.iosrjournals.org

Migrate and Map: A Framework to Access Data from Mysql, Mongodb or Hbase Using Mysql Queries

Piyush Nikam[1], Tushar Patil[2], Gayatri Hungund[3], Ankit Pagar[4],
Aditi Talegaonkar[5], Ms. Soudamini Pawar[6]
[1,2,3,4,5,6]*(Department Of Computer Engineering, D. Y. P. C. O. E., Savitribai Phule Pune University, India)*

Abstract: *Due to ever-increasing amount of data, scalability factor of the databases becomes a major constraint. Moreover, traditional relational databases fix the user's perspective to view data in tabular format. Parallel databases emerged to be one of the solutions to tackle the scalability constraint which was later replaced by more scalable and efficient solution called NoSQL databases. By using the advantages of NoSQL we proposed a framework to migrate data from MySQL to NoSQL databases that is HBase and MongoDB. The framework will be used to map the MySQL query to HBase and MongoDB query format and fetch the data faster by introducing a decision maker that selects the database from which data can be fetched at a relatively faster rate, thus elevating the system performance.*

Keywords: *HBase, NoSQL, MapReduce, MongoDB, MySQL.*

I. Introduction

In the early years accessing Relational Database Management System (RDBMS) for data manipulation was the most popular way as it followed the traditional Atomicity – Consistency – Isolation – Durability (ACID) properties and provides support for storing substantial amount of data. Another advantage of having RDBMS is referential integrity which is a property of data which, when satisfied, requires every value of one attribute (column) of a relation (table) to exist as a value of another attribute in a different table. Also, advantage of using traditional RDBMS is having fine grain locking mechanism. But considering the scenario where the data is changing dynamically, data manipulation and handling becomes a difficult task when it comes to usage of RDBMS. NoSQL on the other hand, is a more efficient and scalable solution as compared to RDBMS. NoSQL basically provides an advantage of being schema less for processing of structured and unstructured data. It also provides MapReduce facility and robust operational tools. Considering all these advantages of using NoSQL this paper proposes a hybrid framework that migrates all the data stored in relational databases such as MySQL into a corresponding NoSQL database format. It also uses popular NoSQL databases such as MongoDB and HBase. The framework will firstly migrate the data stored in MySQL database to MongoDB and HBase. Further, the MySQL queries entered by user are mapped to MongoDB and HBase formatand datais retrieved from the MySQL or NoSQL database selected by the decision maker.

The Technologies Used Are Highlighted As Follows:

1.1. Mongodb:

MongoDB is one of the newest databases introduced by 10gen. It is open source and document type database in which the data is stored in the form of key and value pairs. MongoDB is a schema less database and it provides support to advanced features such as MapReduce and aggregation tools. The biggest advantage of using MongoDB is that it can store files of any size without complicating memory stack.

1.2. Hbase:

HBase is a NoSQL database that provides real time access to big data and efficient big data analytics. It is flexible as it can combine a wide variety of different structures and schemas. HBase is popularly used due to the speed of accessing the data. HBase is used for real time data analysis as it is user friendly and also most importantly, HBase is fault tolerant.

1.3. Javafx:

JavaFX is a set of media packages that is used to create rich client application acrossvarious platforms. It is a library which is written as Java API. JavaFX applicationcan use APIs from any Java libraries and can use these Java libraries to access nativesystem capabilities. The JavaFX APIs are fully integrated features of Java SE RuntimeEnvironment and Java Development Kit (JDK). It is cross-platform compatible.FXML or JavaFX Scene Builder are used to interactively design graphical user interface.Existing Swing applications can be updated with JavaFX features. JavaFXprovides all major UI controls that are required to develop rich applications.

II. Need To Switch From SQL To Nosql

Scalability of NoSQL is higher than that of relational database systems. Large amount of data is stored by horizontal scaling. NoSQL databases can handle data more efficiently as compared to relational database system. This is done by storing the data in distributed manner. NoSQL databases do not have fixed structure to store data. As there is no structure, different types of data can be stored. Caching facility is provided by NoSQL databases to increase the retrieval speed and efficiency of the system.

III. General mapping techniques

IV. The entered MySQL query is converted to parse tree and then this tree is mapped to NoSQL interface.

V. Entered MySQL query is checked by query graph model. S-H and H-H rules are applied by query rewriter to query generated model. Directed acyclic graph is obtained from this query based model with the help of Query compiler. Plan evaluator is used to select optimum Directed Acyclic Graph (DAG) from obtained DAGs.

VI. Using object relational mapping a data mapping module can be created which maps MySQL queries to NoSQL queries.

IV. Related work

Wu Chung Chung et. al. [1] have proposed a framework Jackhare that uses Druid API for implementation of scanner, parser and code generator. JDBC is used for connecting to databases. After entering the query, WHERE clause in the query is verified first and then, the query is checked for names of the tables and rows for which the WHERE clause is used. Finally, the data fields are sent as output in the SELECT statement. Hadoop Map Reduce jobs are used, in which, output of the Map phase is Key-value pairs and Reducer combines the output according to keys.

Julian Rith et. al. [2] have proposed a middleware implemented using c# and is based on .NET framework for query mapping from MySQL to NoSQL in which the queries are divided into two categories which are Data Definition Language(DDL) statements and **CRUD** operations. The query given as input is converted into intermediate parse tree representation and further, this parse tree is mapped to NoSQL format. ANTLR is used for building parse tree.

Xu et. al. [3] proposed a tool which increases probability of optimization of mapping rules using query rewriting to equivalent HiveQL queries. It uses cost estimation to select best method from the different HiveQL query versions which is based on data between mapper and reducer in network. Hive optimizer looks for chances to merge jobs after which QMapper estimator estimates cost of MapReduce DAGs. Further, it selects the method with minimum cost by measuring intermediate data between Maps and Reduces.

Rupali Arora et. al. [4] have proposed an algorithm for migrating data from RDBMS to MongoDB by establishing the connection among the databases after which the database is selected by the user from existing relational databases and corresponding collection is created in MongoDB. The table having its embedded document created in MongoDB is chosen from the selected list of tables. Pentoho tool is used to load data into MongoDB.

Leonardo Rocha et.al. [5] have proposed a NoSQL layer containing a mapping module to which user enters a query as input in MySQL format to retrieve data from NoSQL databases. Mapping is done according to types of statements. Each statement has separate java code which maps MySQL query to MongoDB. The results are stored and converted to string format. The mapped query is executed on MongoDB and result is forwarded to application.

V. Comparative study

Table1: Comparative study between MySQL, MongoDB and HBase

Sr. No.	Factor	MySQL	MongoDB	HBase
1.	Schema	Structured	Dynamic	Dynamic
2.	Scalability	Vertical	Horizontal	Horizontal
3.	Transactions	Complex	Simple	Simple
4.	Data Locality	No	Yes	Yes
5.	MapReduce Support	No	Yes	Yes
6.	Replication	No	Master-Slave replication	Scalable replication

VI. Architecture

To utilize the advantage of horizontal scaling also known as sharding, and processing BigData, the Migrate and Map framework consists of a front end GUI and back end to migrate data stored in existing MySQL database to MongoDB and HBase and a decision maker. The user enters a query in MySQL format and this query is provided as input to decision maker which is programmed to predict the database on which the query can be executed more efficiently which in turn will increase the overall efficiency of the system. Once the database is selected by decision maker, the query is mapped to corresponding database query format and the data is retrieved from the specific NoSQL database.

a. Gui

The GUI is designed using JavaFX which is a set of graphics packages that provides various classes which in turn provide various methods written in native Java for creation of rich looking graphical user interfaces. The WebKitHTML technology provided by JavaFX can be used for embedding web pages within JavaFX application. A designer can code in FXML or use JavaFX Scene Builder to interactively design the GUI. FXML is an XML based directive markup language for constructing a JavaFX application user interface.

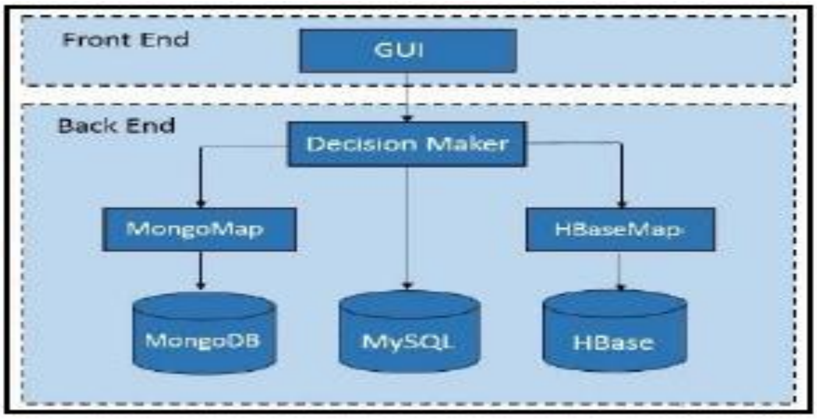

Figure1: architecture diagram – mapping module

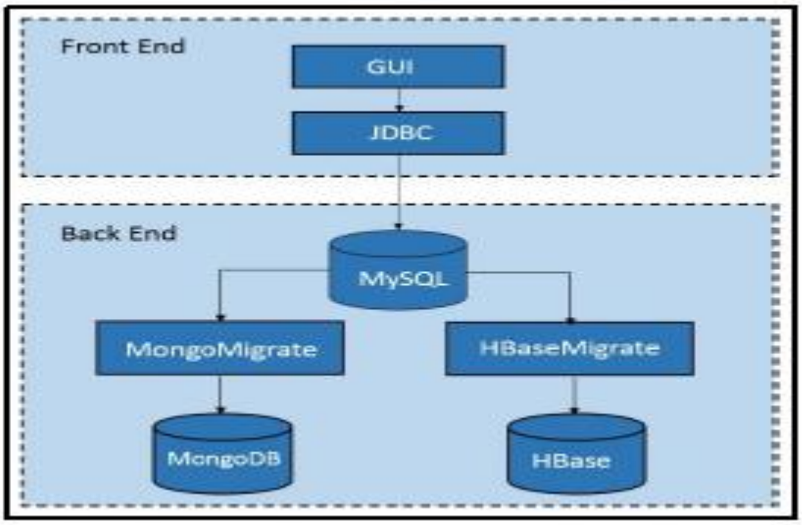

Figure 2: architecture diagram – migrate module

b. Jdbc

Java database connectivity is achieved using JDBC driver which is open source application programming interface and is used to establish connectivity amongst databases to perform data manipulation operations. JDBC allows the existence of multiple implementations which can be used by same application. JDBC is used in the proposed framework to establish connectivity between Java and MySQL. It is also used to fetch the table and database metadata which is required for converting tables from MySQL format to intermediate form which is csv format.

c. Mongomigrate

The MongoMigrate module is programmed to auto-detect all the databases in MySQL and corresponding tables that are to be migrated to MongoDB. Database metadata is extracted to make an ArrayList of available databases which are selected one at a time. Tables are enlisted and added to ArrayList of tables. The selected table is converted to csv file which is then migrated to MongoDB one table at a time. One table in MySQL corresponds to one collection in MongoDB. The implementation uses multithreading for concurrent execution of making csv files and migration of data to MongoDB.

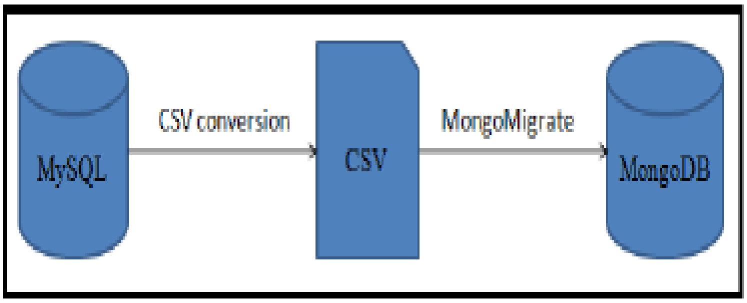

Figure 3: migration from MySQL to MongoDB

d. Mongo Map

In migrate module (MongoDB) we have migrated data from MySQL to MongoDB. The tables migrated from MySQL are stored as a collection in MongoDB. In this module query (Select) fired in MySQL format is tokenized. As there can be many parameters to select query, "," is used as a delimiter. The query mapper takes these tokens as a input and retrieves the data from MongoDB.

Exaple1: select * from student;

As the mapper will encounter a "*", it means that there is nothing to tokenize and hence it will display the data stored entire collection.

Ecample2: select name, rollno from student;

In this query when the mapper will encounter a "," query gets tokenized. Thus only name and rollno. will be retrieved.

Example3: select name, rollno from student where rollno>2;

It will display name and rollno whose rollno is greater than 2. Here not only "," is checked but also operators from "where clause" is checked. When the operator "=", ">" ,"<", ">=", "<=" is found it tokenized the where clause and gives the result accordingly.

e. Hbasemigrate

The HBase migrate module migrates data from MySQL to HBase. This module is also programmed to auto detect and migrate all the available MySQL databases and corresponding MySQL tables. To import data into HBase, the data is first converted into csv format and then imported to HBase.

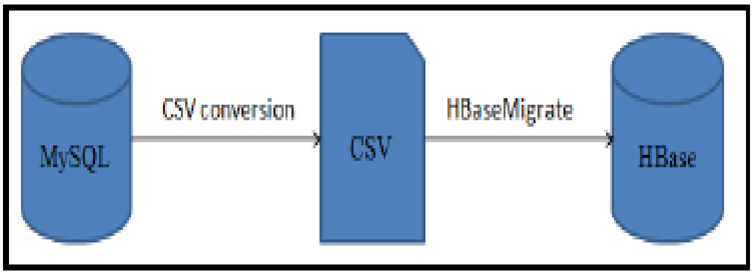

Figure 4: migration from MySQL to HBase

f. Hbasemap

This module maps the query from MySQL to HBase query format. The query is first tokenized then according to different tokens the query is mapped to corresponding HBase query format and fired on HBase to retrieve the data from HBase

VII. Observations

Table 2: Query mapping time

Sr. No.	Query Type	Mapping time (ms)
1.	select * from <table name>	13
2.	select <column name(s)> from <table name>	16
3.	select * from <table name> where <field = value>	50

4.	select <column name(s)> from <table name> where <field = value>	21
5.	select * from <table name> where <field > value>	45
6.	select <column name(s)> from <table name> where <field > value>	38
7.	select * from <table name> where <field < value>	24
8.	select <column name(s)> from <table name> where <field < value>	37
9.	select * form <table name> where <field >= value>	39
10.	select <column name(s)> from <table name> where <field >= value>	41
11.	select * from <table name> where <field <= value>	28
12.	select <column name(s)> from <table name> where <field <= value>	35
13.	select * from <table name> where <field1 operator value1 and field2 operator value2>	21
14.	select <column name(s)> from <table name> where <field1operator value1 and field2 operator value2>	32
15.	select * from <table name> where <field1 operator value1 or field2 operator value2>	27
16.	select <column name(s)> from <table name> where <field1 operator value1 or field2 operator value2>	27
17.	select avg(column) from <table name>	1
18.	select avg(column name) from <table name> group by <column name>	1
19.	select max(column) from <table name>	1
20.	select max(column name) from <table name> group by <column name>	1
21.	select min(column) from <table name>	1
22.	select min(column name) from <table name> group by <column name>	1
23.	select count(column) from <table name>	2

The Table 2 denotes the time taken by the proposed framework to map the MySQL queries to MongoDB and HBase query format.

The time required for mapping is in milliseconds which denotes that in comparison to executing NoSQL queries, MySQL queries can be mapped to NoSQL query format and fired on NoSQL databases. The proposed framework is advantageous as the query execution time is more by just fraction of milliseconds which encourages the use of NoSQL databases and on the other hand, reduces the learning curve of user.

Another advantage of using the proposed framework is that the query mapping time remains constant irrespective of size of data.

VIII. Conclusion

The paper proposes a hybrid architecture in which the data is migrated from traditional relational database that is MySQL, to MongoDB and HBase databases to address the scalability issue in existing relational database systems. The framework processes and serves the queries fired by the user in MySQL format to retrieve required data from MySQL or NoSQL databases. The data retrieval is done according to the decision making mechanism handled by the decision maker.

References

[1]. Chung, W. C., Lin, H. P., Chen, S. C., Jiang, M. F., & Chung, Y. C. (2014). JackHare: a framework for SQL to NoSQL translation using MapReduce. Automated Software Engineering, 21(4), 489-508;

[2]. Rith, J., Lehmayr, P. S., & Meyer-Wegener, K. (2014, March). Speaking in toungues: SQL access to NoSQL systems. In Proceedings of the 29th Annual ACM Symposium on Applied Computing (pp. 855-857). ACM.

[3]. Xu, Y., & Hu, S. (2013, May). Qmapper: a tool for sql optimization on hive using query rewriting. In Proceedings of the 22nd international conference on World Wide Web companion (pp. 211-212). International World Wide Web Conferences Steering Committee.

[4]. Arora, R., & Aggarwal, R. R. (2013). An Algorithm for Transformation of Data from MySQL to NoSQL (MongoDB). International Journal of Advanced Studies in Computer Science and Engineering (IJASCSE), 2(1).

[5]. Rocha, L., Vale, F., Cirilo, E., Barbosa, D., & Mourao, F. (2015). A Framework for migrating Relational Datasets to NoSQL. Procedia Computer Science, 51, 2593-2602.