# **SMH - A QUERY OPTIMIZATION TOOL**

# Omkar Randive<sup>1</sup>, Anuj Bidkar<sup>2</sup>, Yash Mestry<sup>3</sup>, Bajrang Kharat<sup>4</sup>

<sup>1</sup>Student, Computer Engineering, Isb&m School of Technology, Maharashtra, India <sup>2</sup>Student, Computer Engineering, Isb&m School of Technology, Maharashtra, India <sup>3</sup>Student, Computer Engineering, Isb&m School of Technology, Maharashtra, India <sup>4</sup>Student, Computer Engineering, Isb&m School of Technology, Maharashtra, India

#### Abstract

There is tremendous growth in heterogeneous and unstructured data in last few years. Various SQL and NoSQL databases have been developed to store and query this humongous data. MongoDB is prevailing document oriented database among SQL and NoSQL databases. MongoDB has been chosen among other technologies because of its ability to work with variety of latest as well as conventional technologies. On the other hand Hadoop can be contagious. Its' implementation in one organization can lead to another one elsewhere. The ability to include HIVE in an EMR workflow is yet another splendid point. In this Project we are making a Migration-Tool. Using these three Technologies to Convert or integrate the queries. In consideration of the migration of relational database tables, data along with queries into SMH, there are certain issues like current migration systems being incapable of properly mapping table relationships. It could result in the necessity of multiple query statements, thereby decreasing query performance and data redundancy thus wasting space needlessly. Adding up to those complications are the existing query converters only being capable of converting select queries, the inability of query converters to support table joins and most of the tools requiring SQL, MongoDB command or HIVE knowledge. Hence, the focus of the intended system Sql Mongo Hive - 'SMH' is to facilitate users to easily transfer from MySOL tables along with proper relationship mapping opted from embedding and referencing, data as well as queries into MongoDB and HIVE or vice-versa without requiring any prior knowledge of MongoDB commands or HIVE.

Keywords: Sql, MongoDB, Hive, Database, Migration, Hadoop, Query Conversion, Map Reduce, Embedding, Relational Database. Database.

\*\*\*

# **1. INTRODUCTION**

In this project we are going to introduce a new tool called SMH.SMH STANDS FOR Sql, mongodb and hive tool. This Tool is nothing but the migration of SQL,MongoDB and Hive(Hadoop)Technologies. In consideration of the migration of relational database tables, data along with queries into SMH, there are certain issues like current migration systems being incapable of properly mapping table relationships. It could result in the necessity of multiple query statements, thereby decreasing query performance and data redundancy thus wasting space needlessly. Adding up to those complications are the existing query converters only being capable of converting select queries, the inability of query converters to support table joins and most of the tools requiring SQL, MongoDB command or HIVE knowledge.We are proposing multiple query conversion technology at only one platform. With the help of sql, mongdb, and hive database languages we are migrates all query at one platform and doing all the possible conversions amongst this languages. The query conversions from sql to mongodb, mongodb to hive, sql to hive. This conversion is possible through conversion algorithm which

helps to converts query from one database language to another. SMH tool doesn't need a particular DBA to handling this tool, any DBA and any user can handle it but user need a knowledge of basic query amongst sql,hive, mongodb. The user doent need to be aware of sql, mongodb and hive database, only one of the basic knowledge of database language is enough to handle SMH tool and perform the query conversion as in required format. this SMH tool is much easier to use and understand from user side of view. with the help of this tool user can studies all the database language with there queries.

#### 1.1 System Overview

A system design is mainly based on the two factors which are User and Database. where user have to enter the query for conversion and fetching records from the databases. Here we are going to convert three database languages vice versa.



# **1.2 Query Conversion**



The record table shown above is in Mysql database then user will enter query in search box then this SMH tool converts this query in required MongoDB or Hive.and output of this query fetched from the database and display to user.

# 2. ALGORITHM

- 1. Start:- A Query is entered by the user.
- 2. Check whether the given query is in SQL, MongoDB, Hadoop.
- 3. If given query is SQL then go to step 4.
- 4. Check the given condition whether the query is transferred into MongoDB or Hive and go to step 9.
- 5. If given query is in MongoDB then go to step 6.
- 6. Check condition whether the given query is transferred into Sql or Hive and then go to step 9.
- 7. If given query is in Hive then go to step 8.
- 8. Check condition whether the given query is transferred into Sql to MongoDB and then go to step 9.

- 9. For conversion a query we need first to tokenized a query then according to different tokens the query is mapped to corresponding query format also we use conversion algorithm.
- After this proper query conversion the required result is displayed by the tool. Exit

### **3. MATHEMATICAL MODEL**

- 1. Let," q" is a entered query by the user.
- 2. The entered query can be SQL, MONGODB or HIVE. So, let's considered: SQL = f1,

MONGODB= f2, HIVE=f3.

- 3. If "q" is f1 then, q=f1 Now conversion of "q" has possible into either f2 or f3 as, q=f2 || f3 q=o/p.
- 4. If "q" is f2 then, q=f2 Now conversion of "q" has possible into either f1 or f3 as, q=f1 || f3 q=o/p.
- 5. If "q" is f3 then,

q=f3 Now conversion of "q" has possible into either f1 or f2 as,

- $\begin{array}{l} q=f1 \parallel f2 \\ q=o/p. \end{array}$
- 6. Exit.

# 4. FLOWCHART



### **5. TIME COMPLEXITY**

SQL query needs 0.5 seconds to execute a query. Whereas 4 min. required to execute 10,000 rows in database. MongoDB takes 1 ms time to execute 100kb document. Hadoop requires speculative execution time. What is speculative exicution?

Sometimes you will notice that a Job which has 3 input plits executed 4 mappers and killed the 4th mapper. The job would still complete successfully but ever wondered why did it execute the 4th Mapper when there are only 3 input splits and kill it at the end

What you see is called Speculative Execution.

When Hadoop framework feels that a certain task is taking longer on average compared to the other tasks from the same job, then it clones the "long running" task and run it on another node. This is called Speculative Execution. Meaning Hadoop is speculating that something is wrong with the "long running" task and runs a clone task on the other node.

The slowness in the "long running" job could be due to a faulty hardware, network congestion, or the node could be simply busy etc. Most of the the time this is a false alarm and the task which was considered long running orproblematic completes successfully. In that case Hadoop will kill the cloned task and proceed with the results from the completed task.

#### 6. CONCLUSION

The SMH tool is a conversion tool for query languages of MySQL, Hive and MongoDB. Our conversion tool will help developers and employees who use database on a daily basis. All the database related languages are to be migrated in one tool and makes user friendly tool. All database language conversions will be done at one platform. So as the paper shows we will successfully convert data query conversion of MySQL, Hive and MongoDB. We can show all types of query conversions on one platform.

#### REFERENCES

[1]. Mital Potey, Megha Digrase, Gaurav Deshmukh, Minal Nerkar, "Database Migration from Structured Database to non Structured Database", ICRTAET 2015.

[2]. Diego Serrano, Dan Han and Eleni Stroulia, "From Relation Multi Dimensional Maps: Towards ASQL-to -HBase Transformation Methodology" IEEE International Conference2015.

[3]. Gayan Liyanaarachchi, Lakshan Kasun, Malki Nimesha, Kanishka Lahiru, Anuradha Karunasena "MigDB – Relational to NoSQL mapper", Base Paper.

[4]. Research Scholar, Computer Science and Engineering Department, "Proposed Architecture of MongoDB-Hive Integration", International Journal of Applied Engineering Research 2017.

#### BIOGRAPHIES



Omkar Randive, B.E Computer Engineering, ISB&M SOT Pune 412115



Anuj Bidkar, B.E Computer Engineering,

ISB&M SOT Pune 412115



Yash Mestry, B.E Computer Engineering, ISB&M SOT Pune 412115



Bajrang Kharat, B.E Computer Engineering, ISB&M SOT Pune 412115