

Migrating Relational Database Management to NoSQL MongoDB Database

Enugala Dhamoder, M.Tech. (CSE)
Faculty of Computer Science
Lal Bahadur Degree and PG College
Warangal, Telangana, India 506009
email: dhamoder@yahoo.co.in

S.V. Ramana Reddy, M.Sc.(Comp.)
Faculty of Computer Science
Lal Bahadur Degree and PG College
Warangal, Telangana, India 506009
email: sramana05@gmail.com

Abstract- Increasing the use of data in databases needs NoSQL databases that store different data types of data in document orientation without following the advanced definition of databases. The Relational Database Management System stores data in rows of schema definitions whereas NoSQL data is stored in collections of documents of JSON(Java Script Object Notation). JSON format is useful to machines for parsing data. NoSQL MongoDB technology provides schema-less, clear object definition that does not require complex joins in accessing its database. Existing relational database management systems can be converted to NoSQL MongoDB non-relational databases and utilizes its benefits such as big data storage, quick access and object oriented programming in scalable architecture efficiently. This paper describes the Data Migration Module, Data Mapping Module and Framework of SQL and NoSQL interaction.
Keywords: MongoDB, migration, mapping, Relational Database and metadata

I.INTRODUCTION

There are several organizations which are moving towards conversion from Relational DBMSs such as Oracle, MySQL, PostgreSQL to NoSQL MongoDB. MongoDB provides horizontal scaling and high availability that gives power to Big Data by redefining the databases. MongoDB has an advantageous framework that enables advanced analytics. In Business Analytics customers easily meet requirements for big data using MongoDB.

The following table shows the equivalent terminology of Relational Database Management and MongoDB NoSQL non-Relational Database

Relational DBMS	MongoDB
Database	Database
Tables	Collections
Rows	Documents
Columns	Fields

MongoDB document contains JSON format of data equivalent to rows in RDB as following

```
{
  Empname:"rajesh",
  Empno :201,
  Emppcity: "Warangal"
}
```

MongoDB is chosen as target database for conversion from Relational Database because of few advantages such as document-oriented, functioning with a centralized server, data can be replicated on multiple servers, dynamic schema and query performance with speed in data access. A Database Migration to another

database contains Extraction, Transformation and Loading processes of data.

II.DATA MIGRATION MODULE

This module contains the following steps

1. Getting complete metadata information from the source Relational DBM
2. Designing the database schemas for Target NoSQL database.
3. Loading data from source to targeted database.

The first step is the process of getting the name of the database, table names, column names, and constraints, etc. metadata information from the database. For example, using Java programming language metadata information is retrieved from the Relational DBM.

It provides java.sql package classes and interfaces that consists of several methods some of which are mentioned as following.

ResultSetMetaData Interface contains following methods.

```
public int getColumnCount ()
public String getColumnNme (int)
public String getColumnType (int)
public String getTableName()
DatabaseMetaData interface contains the following methods:
public String getDatabaseProductName ();
public String getDatabaseProductVersion ();
public String getDriverName ();
public String getDriverVersion ();
public String getDriverMajorVersion ();
public String getDriverMinorVersion ();
```

The second step, combine the objects into one schema in MongoDB design such that it could not arise joining in querying data. Schema design, Data can be duplicated because compute time is costly than storage space. Design the schema that requires joining in write operation but not in reading.

The third step, Programming Language java can be used to retrieve data from Relational database by using select * from table name from a source database and can analyze the target database and can use programming

instruction that uses insert SQL command to load data into target NoSQL MongoDB database.

III.DATA MAPPING MODULE

Mapping database is the technique to relate source database fields to target database fields. It defines attributes of a source relational database to load its data to the target MongoDB database. This approach is done through well defined algorithms. Algorithms deals with complex relationships, define the keys in migration, constraints and index migration for a fast query search.

Relational Database relations with relationships can be transformed into new well defined databases of MongoDB. Embedding is used to store data in the MongoDB database. Embedding databases is used to store related data in one document.

Source database relation that contains 1:1 relationship with another relation, it is preferable for the target MongoDB database to embed retrieved information of both relations into one table. For example Document employee related to only one address document. The following example demonstrates one to one relationship embedding in MongoDB.

```

db.artists.insert(
{
  empno:101,
  empname : "Rajender",
  address: {
    houseno:"5-22-222",
    village : "hanamkonda",
    district: "Warangal",
    state:"Telangana",
    country:"India",
  }
}
)
    
```

1:M relationship in source database model can be transformed into MongoDB by linking, embedding or bucketing strategies, in which embedding avoids complex joins to retrieve the information. For example, if many employees will work in one department then one to many relationships between department document and employee document sample code is demonstrated as follows.

M:N relationship of Relational Database consists of two 1:M relations with intermediate relation which contains primary keys that are two foreign keys of two different relations. M:N relationship uses either one way embedding or two way embedding. MongoDB Data model in this approach can create two documents differently for two 1:M relationships which could break

M:N relationship to document oriented denormalized databases.

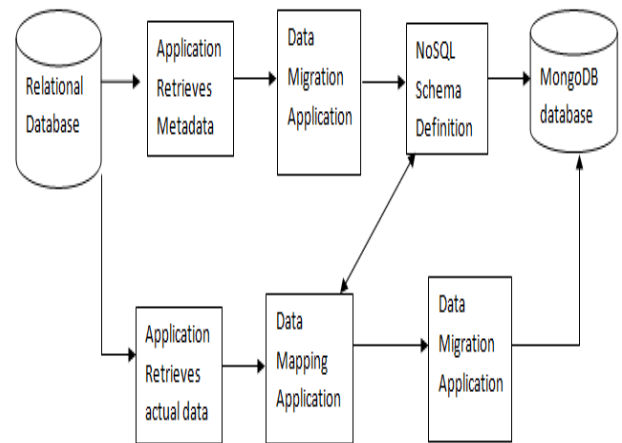
```

db.artists.insert(
{
  deptno:10,
  deptname : "Sales",
  employee: [
    {
      empno:101
      empname : "rajender,
      empsalary : 50000
    },
    {
      empno:102
      empname : "ramesh,
      empsalary : 40000
    }
  ]
}
)
    
```

IV.FRAMEWORK OF RDM AND NOSQL INTERACTION

The framework describes the complete process of the conversion from Relational Database to MongoDB NoSQL database. Relational Database Metadata is retrieved by an application that uses any programming language. In the next step, the Data Migration Application algorithms design the NoSQL schemas in MongoDB database. The Data Mapping Application that maps and loads the data from Relational Database using Migration Module.

Relational Database and NoSQL interaction in migration



V.CONCLUSION

Well defined algorithm concepts are required to define NoSQL Schema Definition while converting from Relational Database that Defines fields, constraints, and indexes into for consistency, reliability and efficient access of databases. A software package can be developed that automates the process of conversion of a database that can increase scalability and availability of MongoDB NoSQL database access.

REFERENCES

- [1] Automatic Mapping of MySQL Databases to NoSQL MongoDB by Liana Stanescu, Marius Brezovan and Marius Brezovan Proceedings of the Federated Conference on Computer Science and Information Systems pp. 837–840, ACSIS, Vol. 8. ISSN 2300-5963
- [2] Data Migration Methodology from Relation to NoSQL Databases by Mohammed Hanine, Abdesadik Bendarag and Omar Boutkhoul - World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering Vol:9, No:12, 2015
- [3] Approaches used in efficient migration from Relational Database to NoSQL database by Arati Koli and Swathi Shinde, Proceedings of the Second International Conference on Research in Intelligent and Computing in Engineering Vol. 10 ISSN 2300-5963
- [4] Relational to NoSQL Database Migration by Aparna Babu and Subu Surendran, International Journal of Innovative Research in Science, Engineering and Technology An ISO 3297: 2007 Certified Organization Volume 6, Special Issue 5, March 2017, Relational Database to NoSQL Conversion by
- [5] Schema Migration and Mapping by Neelima Kuderu and Dr. Vijaya Kumari, International Journal of Computer Engineering In Research Trends, Volume 3, Issue 9, September-2016, pp. 506-513