Implementation of Cross-platform Language between SQL and NoSQL Database Systems

Kenechukwu K. Okeke
Nnamdi Azikiwe University, Awka.
Kenechu.okeke@gmail.com

Virginia E. Ejiofor
Nnamdi Azikiwe University, Awka.
virguche2004@yahoo.com

ABSTRACT

This work proposes to look into the construct of SQL (structured Query Language) and NoSQL (Not only SQL) systems. This will help proffer a framework that guarantees interoperability across board for SQL and NoSQL platforms. This research is being carried out to find a lasting solution to the problem of data management and analysis. Various mathematical/statistical laws will be looked into to help create a grounded study which would be used as set down formulae for the proposed work and the final outcome. A theoretical construct will be proposed which will fill up this gap in the field of database research.

CCS Concepts

• Information systems →Database designs and models

1. PROBLEM STATEMENT

Big data management poses a serious obstacle in the IT research and industry domain. Increasing amount of data produced has rendered the traditional (SQL) systems incapable of handling these volumes of data both in speed and structure. New systems have been developed to combat this major issue but the problem yet persists. Availability of data is at forefront of the NoSQL management system which leaves consistency of data a secondary consideration to this architecture. SQL stands for consistency and integrity of data but still lacks the ability of horizontal partitioning/scalability as it partitions vertically. These various systems; the SQL and different NoSQL systems have their strong attributes which if properly integrated could avail the research community and the industry with a system which could be relied on to parse data across the different platforms and provide high consistency and throughput.

2. RELEVANCY

Paper-based management system is still predominantly the method used for data storage at the Nnamdi Azikiwe University Teaching Hospital, Nigeria. The choice of the hospital is for a test-case scenario as the research is not entirely based on the hospital but on the integration of a cross-platform language between SQL and NoSQL systems. The Relational model is still very feasible but the everyday growth in data has called for systems that can offer mass storage of information that can be scaled out and which can be easily handled. The system proposed will offer interoperability between SQL and NoSQL systems.

CoRI'16, Sept 7–9, 2016, Ibadan, Nigeria.

3. BACKGROUND AND RELATED WORK

Liu and Vitolo [3] worked on the concept of 'Graph Cube', a design which integrates graphs with tables. This concept serves as a prototype which is the basis of a graph data warehouse. Some DML (select, insert, update and delete) and DDL (create, drop and update) in SQL is synchronized with the graph data model to give GDML (Graph Data Manipulation Language) and GDDL (Graph Data Definition Language). This model allows for views in the graph data warehouse. Also, Lawrence [2], worked on SQL and NoSQL integration using MongoDB and MySQL. The architecture of the system is based on the construction of a JDBC driver which accepts SQL queries through the use of SQL parser which produces a parse and relational operator tree. This process is made possible through a virtualization Execution Engine which serves as the middle-layer accepting and translating information across the two management systems. As stated by Kaur and Rani [1] graph databases represent data in their natural format using graphical forms which shows better representation rather than tabular forms.

This research seeks to serve as furtherance of the work carried out by Lawrence (2014) at the University of British Columbia, Canada. Lawrence (2014), stated that "Future work involves benchmarking the performance of other supported NoSQL systems such as Cassandra. We are also working on parallelizing the virtualization engine for cluster environment."

4. RESEARCH APPROACH AND METHODOLOGY

SQL systems run on relational algebra. This puts both referential and integrity constraints on data. This is based primarily on set theorem which identifies a relation between different variables and links them up. The join operations are made possible through this method. Statistical tools and algorithms employed for data analysis allows for 'polyglot persistence' (the use of various technologies in data management) thereby meeting different storage needs.

5. PRELIMINARY RESULTS

Some applications (visual Studio 2013, Mongo DBMS, software Ideas Modeler, EC2 have been downloaded to assist in the development of this cross-platform system. Graphical interface using C# programming has been created and this can run on the different NoSQL systems. MongoDB which is a document-store NoSQL management system is being worked on using

RoboMongo as the GUI/client to connect to the mongo database. The column-family NoSQL system will be next as Cassandra

6. EVALUATION PLAN

Comparison of different drivers on which the different NOSQL systems and SQL run on will be examined to find the peculiarities which help ascertain uniformity of syntaxes which grants an integrated system. The systems will be evaluated individually to find algorithms will are similar, then collectively for interoperability.

7. EXPECTED CONTRIBUTION

An adaptive and deep learning cross-platform database system for efficient/effective data management.

8. REFLECTIONS

Data management is an evolving topic and as such needs dynamism in its approach. Cloud computing and BigData are most recent trends with need for both theoretical and hands-on approach. Other data management needs will definitely arise and future work will be in terms of a unified query language for both SQL and NoSQL database systems.

REFERENCES

[1] Kaur, K. and Rani, R (2013). Modeling and querying Data in NoSQL Databases. IEEE International Conference on Big Data. Pp.1-7.

[2] Lawrence, R. (2014). Integration and Virtualization of Relational SQL and NoSQL Systems Including MySQL and MongoDB. International Conference on Computational Science and Computational Intelligence (CSCI). Pp. 285-290.

[3] Liu, Y. and Vitolo, T.M. (2013). Graph Data Warehouse: Steps to Integrating Graph Databases Into the Traditional Conceptual Structure of a Data Warehouse. IEEE International Conference on Big Data (BigData Congress). Pp.433-434.