Analysis of NoSQL Databases:Mongodb,HBase,Neo4J

[1]Suman Tiwari, [2] M.Akkalakshmi, [3]KrishnaKasyap Bhagavatula

[1]*Dept of CSE GITAM University, Hyderabad, India*,

[2]*Dept. of CSE GITAM University Hyderabad, India*,

[3]*Technical writer at Tutorialspoint, Hyderabad, India*

Abstract- *NoSQL gained popularity since April 2009 and from that time till date we have been using NoSQL databases for managingi huge data generated everyday. There are four different classes of NoSQL databases. These are Key value stores, Document databases, Column-oriented databases and Graph databases. Each type is precisely used for some specific type of dataset. The only thing common between all these datasets are, they are used to manage non relational datasets. One can store huge amount of data without any failure. This paper shortly briefs about the different categories of NoSQL databases. The main objective of this paper is to compare database from three different categories. Mongodb from Document databases, HBase from Column Oriented database and Neo 4J from Graph database are being compared in this paper.*

***Index Terms*—HBase,MongodbNeo4J,NoSQL**

I. INTRODUCTION

Since the 70's database means relational databases. They were known for their rich pool of features, query capabilities and transaction management. It was fit for all the possible tasks one can think of, to do with a database. But with all the positives thee is a major drawback in relational database, building distributed RDBMSs becomes very complicated. In simple words, it is not efficient and very difficult to make transactions and join operations in a distributed system.

To overcome this major flaw non relational databases or NoSQL databases were introduced. NoSQL is an abbreviation for Not Only SQL. This term was coined in 1998 as a database, but it came into limelight in April 2009 as seen n Google Trends. This non relational database works on the fact that one single technology is not always fit for everything. NoSQL databases are specifically designed to run on clusters that consists of commodity computers and thus have to be distributed and failure tolerant. It was mainly designed to meet the ever growing requirements of web services and generally they are schema fee and have their own query languages. The aim is to brief about the current state of NoSQL databases and their comparison with each other. One of the most important plus point of NoSQL databases are, they process data faster when compared to relational databases. The simple data models are the reason behind their fast processing.

NoSQL works on CAP theorm. It stands for Consistency, Availability and Partition Tolerance. These features are known are CAP theorem, according to this theorem all three features cannot be achieved fully at the same time. Only two out of the three different characteristics can be achieved fully at the same time.

NoSQL databases are ACID free. They are specifically designed for distributed database environment where data is distributed over different machines and every machine stores it's data and maintenance of consistency is required.

These non relational databases follow BASE properties.

II. LITERATURE SURVEY

I. NoSQL databases are broadly categorized into 4 types. The performance of all these four types are examined by storing and retrieving data. The paper used 10 databases for comparison through Yahoo! Cloud Servicing Benchmark. The paper used several operations that would help in understanding the skills and capabilities of non relational database for handling different requests and understanding performance of each database type along with their internal mechanisms.

II. The different features of NoSQL were described that differentiates it from SQL were mentioned. The paper further compares all the categories of NOSQL databases on the basis of different software attributes like availability, consistency, durability, maintainability, performance, reliability, robustness, scalability, stabilization time and recovery time.

III. The introduction to NoSQL along with its need, features and classification were mentioned in this paper. This paper mainly focuses on the working of NoSQL with big Data Analytics. The paper also covers the different advantages and disadvantages of different types of NoSQL databases. Alongwith several NoSQL database approaches for managing different applications processing huge volume of data for their project and applications.

IV. The paper describes NoSQL databases thoroughly along with brief introduction on classification, characteristics and evaluation of NoSQL databases in BigData Analytics. In addition to this the paper talks about the current state of NoSQL database. The paper further compares different NoSQL databases on the basis of design and features, integrity, indexing, distribution and system.

V. The paper introduces NoSQL and states the importance of NoSQL databases in today's world. The paper also distinguishes several NoSQL databases on the basis of Data Model, Query Model and Replication Model. It further concludes on selecting particular database for a particular scenario.

VI. This paper mentions the different research methodologies used in stating the literature review for NoSQL mechanisms. The various drawbacks of NoSQL and the solutions for each drawback were also stated. The paper presents all the results through ming mapping that gives a deeper understanding of the concepts.

VII. The paper introduces NoSQL and states how it is different from RDBMS by stating its features and classification. It further states the importance of NoSQL for growing data alongwith some drawbacks of NoSQL databases.

VIII. This paper points out the NoSQL movement. It gives a deep understanding the important features of NoSQL like sharding and replication, eventual consistency, CAP. The paper distinguishes various databases on the basis of sorting, range queries, aggregations, durability and CAP properties.

IX. This paper briefs about Relational databases and differentiates it from Non relational databases . The paper covers up the various features of RDBMS alongwith it's drawbacks. Followed by briefly introducing NoSQL, it's different features and classification.

III. NoSQL Databases

NoSQL is known for it's ability to process huge chunk of data and quickly distribute the data over computing clusters. It was precisely designed to create flexible schema for cloud and web systems that are updated continuously. NoSQL databases are broadly classified into four different categories.

A. Key Value store

The key-value databases execute a simple data model by pairing it with a unique key with a related value. The simplicity of this data model results in development of key-value databases. These databases are known for their high performance and high scalability for session management and caching in web applications. The executions in this database differ according to their working with RAM, solid-state drives.

Some examples of key-value database are Aerospike, Berkeley DB, MemchacheDB, Voldemort,Redis and Riak.

B. Document-Stores

The Document databases are also known as document stores. They store semi-structured data along with briefing of that data in document format. The document databases permits the developers to create and update programs without the presence of any reference master schema. The use of JavaScript and the JavaScript Object has further increased the use of document databases. Alongwith using several data interchange formats, XML and data formats. This type of database is widely used for content management and mobile application data handling.

Some examples of Document Store databases are Couchbase, CouchDB, Document DB,MarkLogic and MongoDB.

C. Column Oriented Database

The column oriented family are also known as wide-column stores. They arrange data tables as columns instead of arranging them as rows. The column oriented databases can be easily used for both relational and non relational databases. They are widely used for applications like recommendation engines, catalogs, fraud detection and several other types of data processing.
Some examples of Column Oriented databases are Google Big Table, Cassandra and HBase.

D.Graph Database
The Graph database arranges data in the form of nodes just like relational databse store records. The edges mark the relationship between nodes. Since the graph databases store the relationship between nodes, it also helps in richer presentations of data relationships. Just like relational models follow strict schemas, the graph databases have improvised over time and in terms of it's applications. These are highly recommended for systems that require map relationships like reservation systems or the customer relationship management.
Some examples of Graph databases are AllegroGraph,IBM Graph, Neo4J and Titan.

IV. NoSQL Databases used For the experiment

A. *MongoDB*
MongoDB is an open source document database written in C++. It is an object oriented, simple, dynamic and scalable database. Here the data objects are stored as different documents within a same collection, rather than storing the data into rows and columns like relational database. The objective behind MongoDB is to ensure the data store yields high performance, high availability and automatic scaling. It's an easy to install and execute database. It uses JSON and BSON documents for storing data. It runs on windows, Linux, Mac OS X and Solaris.

B. *HBase*
Apache HBase is known as one of the most popular NoSQL database constructed upon Hadoop and HDFS. It is widely known as Hadoop database. It is an open-source, versioned and distributed database. The HBase database is written the Java language and is constructed on concept's of Google's BigTable.

C. *Neo4J*
Neo4J belongs to graph database management system that was developed by Neo Technology Inc. It's considered as the most popular graph database .. It is executed in Java and is easily accessible from software written in Cypher Query Language. Basically Neo4J is a native graph database used to save and process graphs from bottom to top. It uses typical graph storage management technique to save and handle interconnected data.

V. Set UP
For the experiment we have used the latest version of MongoDB, HBase and Neo4J in Windows 8.1 Pro.

i. Go to the official page of Mongodb to download it's latest version.
ii. Install Mongodb in the system.
iii. Open two command prompts.
iv. Type mongod in one and mongo in the other command prompts to make the connection.
v. Now you can type your queries in Mongo.
vi. Then go to the official page of Neo4J to download it's latest version.
vii. Install Neo4J in the system.
viii. Double click on the Neo4J icon displayed in your desktop.
ix. Click on the start button and connect to the local host.
x. Give your user name and password
xi. Your Neo4J is ready to use. Write your queries.
xii. To close the connection click on stop.
xiii. To install HBase first install java.
xiv. We have installed HBase in standalone mode.
xv. Go to the official page of Apache HBase to download the latest version of HBase. You can also refer to the site http://www.interior-dsgn.com/apache/hbase/stable/ and with the use command "wget" then extract it with the help of tar "zxvf" command.
xvi. After doing some configuration settings you can start the HBase shell.

After. successfully installing MongoDB, HBase and Neo4J in our systems we try some sample queries to check creating their working.
For our experiment we have used different datasets and certain queries to retrieve the desired results. The outputs are different from the one in relational database. The objective was to create the database and then compare the output on different criterias.

VI. Experiment
For comparing the three databases we have used different scenarios describing the different results obtained in each case scenario.

A. *Scenario 1*

In the first case we create a dataset in all the three databases and observe the output.

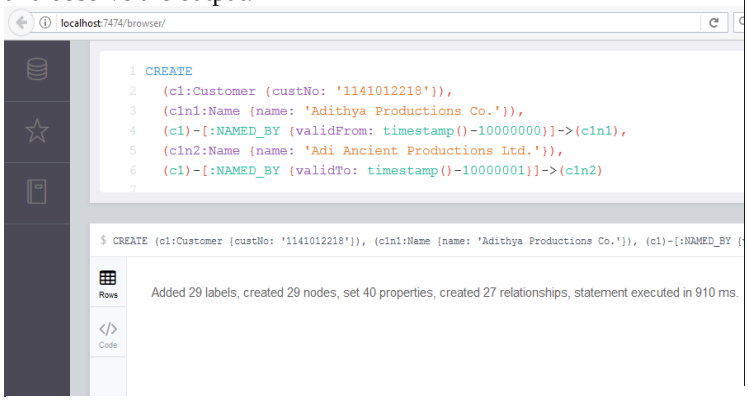


Fig.1 Creating dataset with Neo4J

In the Fig.1 we have created a dataset with customers and employees . We take a small dataset from a local grocery store. The relational model will have a customer node,Orgunit node, Employee node, Name and Address node. In order to enhance the features of Neo4J we will show all the relations between the nodes.

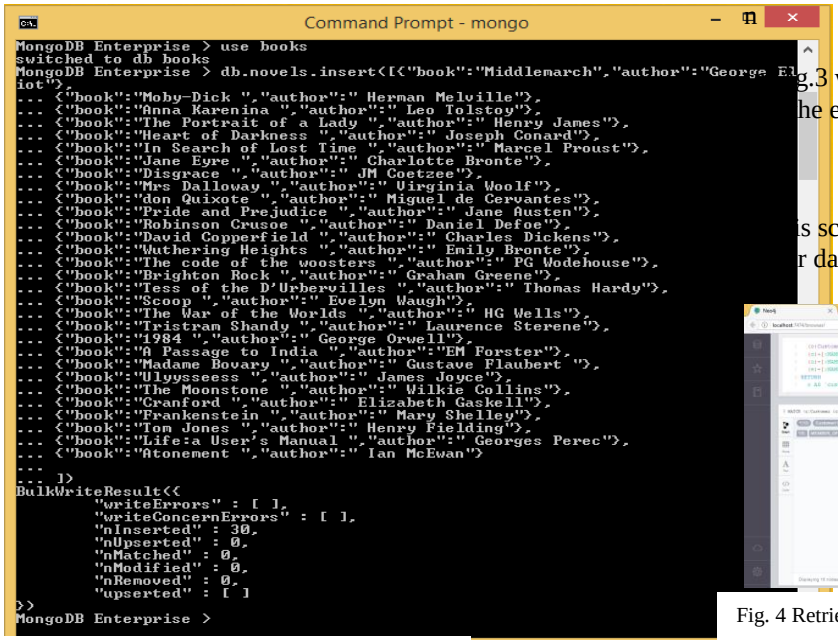


Fig.2 Creating dataset with MongoDB

In the Fig.2 we have created a dataset regarding 50 must ready novels with their respective authors.

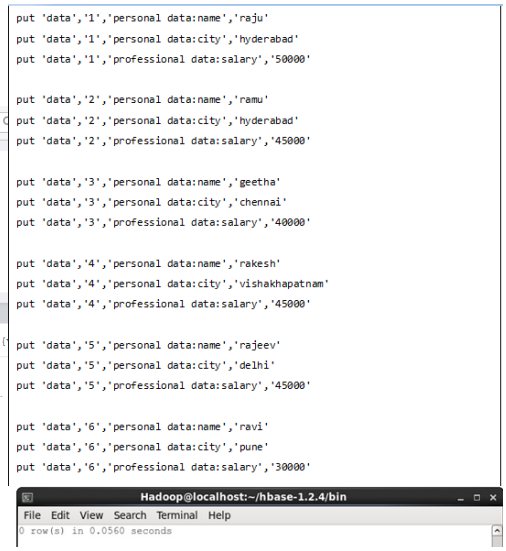
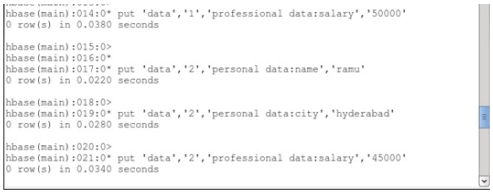


Fig.3 Creating dataset with HBase



g.3 we have created a dataset with employee details
he employee name, city and salary.

B. *Scenario 2*

is scenario we try to retrieve all the data we have stored
r databases.

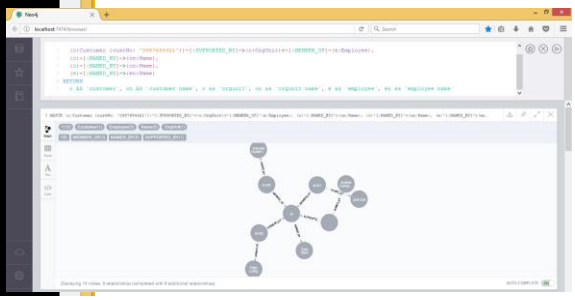


Fig. 4 Retrieving all the customer and employee details in Neo4J

In Fig.4 we have retrieved all the data created in the dataset by using MATCH and RETURN. We can clearly see the relationships between the nodes.

Fig. 5 Retrieving all the book details in MongoDB

In Fig.5 we see all the book records along with unique object id that is automatically generated for each record. We retrieved all the records using db.collectionname.find() command, we have used pretty() to make the presentation look better.

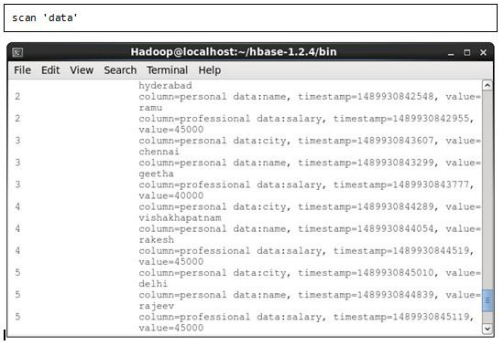
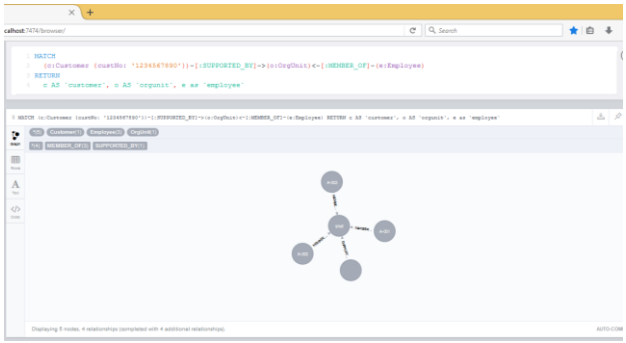


Fig. 6 Retrieving all the employee details in HBase

In Fig.6 we see all the employee details we have created. To retrieve all the details we have use the Scan command. All the data records have their unique time stamps that differentiates them from the other one.

C. *Scenario 3*

In this scenario we try to search for one particular record from the dataset.



In Fig.7 we have received all the details of a particular customer with a particular customer id.

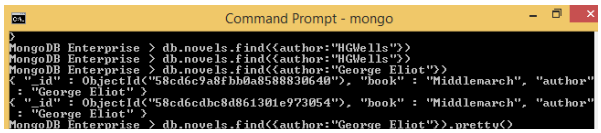


Fig.7 Search for a particular customer details in neo4J

Fig. 8 Search for a particular book in MongoDB

InFig.8 we have retrieved the details of one particular book by giving the Author name.

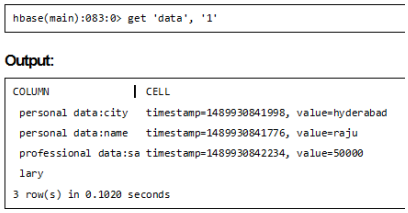


Fig. 9 Search for details of particular employee in HBase

The Fig.9 illustrates the details of a particular employee through the "get" command.

VII. CONCLUSION

We have performed some basic operations in all the three databases and observed the outputs in all the scenarios. From our observations we have concluded when to use and when not to use any of these databases.

MongoDB is widely recommended while working with E-commerce product catalog, blogs, content management. Projects related to Real -time analytics, configuration management, mobile and social networking use MongoDB. When handling data with changing requirements or data with loosely coupled objectives MongoDB is the best database to use. However MongoDB is not recommended for highly transactional systems or for tightly coupled systems.

HBase is widely recommended for projects demanding real time and random read/write access for huge chunks of Big data. Another important point here is the data should be stored in terms of peta bytes that could be easily processed in a distributed environment. However Hbase is not recommended while using transactional applications, large volume map-reduce applications and relational analytics. It's very important to have good hardware support for using Hbase efficiently.

Neo4J is recommended for projects like Fraud detection, real time recommendations, social networks, data center management and master data management. It's very popular and the best database for reservation system and customer relationship data.Neo4J are not recommended for projects with large datasets. They donot have a declarative language which makes them unable to optimize queries in a proper way.

REFERENCES

[1] Veronika Abramova, Jorge Bernardino and Pedro Furtado "Experimental Evaluation of NoSQL Databases", International Journal of Database Management Systems(IJDMS) Vol.6,No.3,June 2014.

[2] Joao Ricardo Lourenco, Bruno Cabral, Paulo Carreiro, Marco Vieira and Jorge Bernardino, "Choosing the right NoSQL database for the job:a quality attribute evaluation",Lourenco et al, Journal of Big Data (2015)2:18.

[3] Raju Sharma,Yatendra Kashyap,"A study of NoSQL databases and working overviews", International Journal of Recent Trends in Engineering and research Volume 02, Issue 02, February 2016.

[4] ABM Moniruzzaman and Syed Akhter Hossain, "NoSQL database: New era of databases for big data analytics-classification, characteristics and comparison", International Journal of Database theory and application Volume 6, No. 4, 2013.

[5] Clarence J.M.Tauro,Baswanth Rao Patil and K.R.Prasanth,"A comparative Analaysis of different noSQL databases on /data model, query model and replication model", Elsevier publications 2013.

[6] Jaroslaw Kurpanik, Malgorzata Pankowska, "No SQL problem literature review", Studia Ekonomiczne.Zeszyty naukowe Uniwersytetu Ekonomicznego w Katowicach ISSN 2083-8611 Nr 234.2015.

[7] Michael Madison, Mark Barnhill, Cassir napier, Joy Godin, "NoSQL Database Technologies", Journal of International Technology and Information Management volume 24, Number 1 2015.

[8] kai Orend, Florian Matthes, Thomas Biichner,"Analaysis and classification of NoSQL databases and evaluation of their ability to replace an object relational persistence layer, Master's Thesis software Engineering for business information system 2010.

[9] Neal Leavitt,"Will NoSQL database live up to their promise", IEEE computer society 0018-9162/10 2010.

[10] case study", Association of computing machinery 2015.

[11] https://www.tutorialspoint.com/hbase/

[12] https://www.tutorialspoint.com/mongodb/

[13] https://www.tutorialspoint.com/neo4J/

[14] https://dzone.com/articles/why-mongodb

[15] http://blog.cloudera.com/blog/2011/04/hbase-dos-and-donts/

[16] http://searchdatamanagement.techtarget.com/definition/NoSQL-Not-Only-SQL