International Journal of Engineering & Technology, 7 (2.6) (2018) 83-87

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Review Paper

Comparative study of NoSQL databases for big data storage

Gourav Bathla[1]*, Rinkle Rani[2], Himanshu Aggarwal[1]

[1]Punjabi University Patiala, India
[2]Thapar University Patiala, India
**Corresponding author E-mail: gouravbathla@gmail.com*

Abstract

Big data is a collection of large scale of structured, semi-structured and unstructured data. It is generated due to Social networks, Business organizations, interaction and views of social connected users. It is used for important decision making in business and research organizations. Storage which is efficient to process this large scale of data to extract important information in less response time is the need of current competitive time. Relational databases which have ruled the storage technology for such a long time seems not suitable for mixed types of data. Data can not be represented just in the form of rows and columns in tables. NoSQL (Not only SQL) is complementary to SQL technology which can provide various formats for storage that can be easily compatible with high velocity, large volume and different variety of data. NoSQL databases are categorized in four techniques- Column oriented, Key Value based, Graph based and Document oriented databases. There are approximately 120 real solutions existing for these categories; most commonly used solutions are elaborated in Introduction section. Several research works have been carried out to analyze these NoSQL technology solutions. These studies have not mentioned the situations in which a particular data storage technique is to be chosen. In this study and analysis, we have tried our best to provide answer on technology selection based on specific requirement to the reader. In previous research, comparisons among NoSQL data storage techniques have been described by using real examples like MongoDB, Neo4J etc. Our observation is that if users have adequate knowledge of NoSQL categories and their comparison, then it is easy for them to choose best suitable category and then real solutions can be selected from this category.

Keywords: *NoSQL Database, Column oriented, Graph based, Document based, Key Value*

1. Introduction

Structured data is handled by traditional relational databases over the years. In traditional relational database, data is stored in rows and columns format. Big data is combination of structured, semi-structured and unstructured data. A lot of information is generated due to interactions on social networking sites and mobile applications. Semi-structured and unstructured data can not be stored in the form of tables as in relational databases [1]. These forms of data can be stored and processed by Big Data technologies only. Moreover sql data storage is horizontally scalabale i.e. if large scale of data is to be stored and processed, then storage capacity is to be increased only inside single server. There is limit on server capacity enhancement. Distributed data storage, cloud storage, NoSQL and NewSQL are latest techniques to deal with large scale of mixed data. NoSQL (Not Only SQL) term was introduced by C. Strozzi in 1998 in which SQL interface was not used [2]. In 2009, NoSQL was re-introduced by Johan Oskarsson in a conference on "open-source, distributed, non-relational databases". NoSQL database will not replace relational database, rather these databases compliments each other [3]. Atomicity, Consistency, Isolation and Durability are the properties which are provided by relational databases. In the era of Big Data, when query response time matters a lot, then it is necessary to distribute the large scale of data. NoSQL database supports BASE (Basically Available, Soft state, Eventual Consistency) properties [3]. BASE prioritizes availability of data than consistency. It also allows approximate answers provided with fast response time. Performance requirements are not only required these days, rather many quality attributes like availability, consistency, durability, maintainability, reliability and scalability are need of current business and research organizations. It is easily distributed and scalable to handle large scale of data [4]. The main characteristics of NoSQL data storages are high availability and strong consistency [4]. There are 120 solutions available for NoSQL databases at present. Several research works have compared most popular solutions but in this paper, we have compared NoSQL databases architecture. This comparison seems more effective as it completely demonstrates real differences based on schema, query languages, consistency, availability and response time. In this paper, important differences in NoSQL databases are identified which can provide guidance to researchers and practitioners to select the most appropriate solution.

Standard query language is not defined for NoSQL data stores. This is due to the fact that various NoSQL solutions use different structure for storage and query. Researchers are able to solve it by articulating standard query language for one category based solutions. It is in scope of further research to articulate and develop standard query language for various categories.

The rest of the paper is organized as follows: In Section 2, Big data storage difficulties and opportunities are elaborated. Categories of NoSQL databases with real examples are explained in Section 3. Comparison between NoSQL databases are highlighted in Section 4. NoSQL query languages are described in Section 5. Section 6 concludes the paper and provides the further research perspectives to researchers.

2. Big Data Storage

Big data is large scale of data which is generated due to Social networks, Business organizations, interaction and views of social connected users. It is used for important decision making in business organizations. It is represented by 3Vs (Velocity, Variety and Volume). Velocity is the rate at which data comes from small scale to become large scale. Variety is different types of data-structured, semi-structured and unstructured. Volume is the amount of data which is very large in scale. Data acquisition, data analysis, data curation, data storage and data usage are important phases of Big data mining [6]. There are several characteristics of big data storage- Cloud storage, query interfaces, NoSQL, NewSQL, Scalability, Consistency, Security and Performance etc. Traditional storage can be efficiently implemented by relational databases like SQL and processed on Weka, Java etc. Scalability is not well managed by relational databases [3].These can be efficiently deployed for structured data. Several techniques have been articulated and implemented by researchers to deploy large scale of data. These novel strategies to store large scale of data provide scalability with less complexity. This is verified with the popularity of storage like Cloudera, MapR and NoSQL solutions [6].

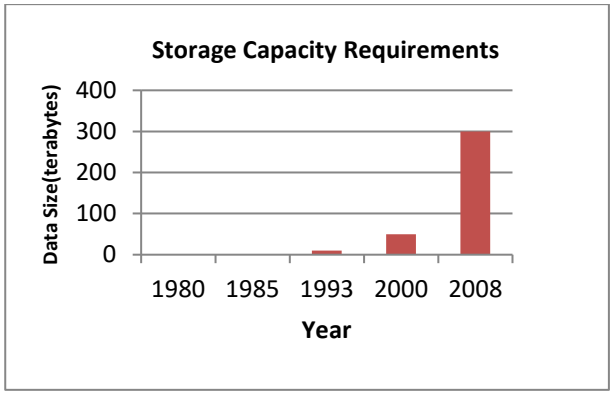

Fig. 1: Requirement of storage capacity

It is clear from Figure 1 that there is need of storage technique which can handle large scale of unstructured data, which is not possible for relational database.

Big data storage is provided by following three techniques [Strohbach]:

1. Distributed File Systems
2. NoSQL
3. NewSQL

Distributed File System: HDFS (Hadoop Distributed File System) is used for distributing large scale of data on different clusters. These clusters work in parallel on chunks of data and after processing merge to form final results. Hadoop MapReduce is deployed for mapping and reducing unstructured type of data.

NoSQL : This storage technique is used for data where tables in rows and columns can not be applied. There are many NoSQL solutions available which can remove the drawbacks of relational databases, which is explained in next section. NoSQL databases provide scalability but with the increase in scale of data, scalability limits are reduced slightly**.**

NewSQL : Relational databases with novel techniques to process large scale of data comes under this category. This is area where further research is required. The advantage of using this technique is relational databases benefits for Big data are provided. NewSQL is used for multi-object transactions like in finance services, where multiple objects can use concurrent transactions. NoSQL databases can not be deployed for this scenario. It is expected that NewSQL is 50 times faster than simple SQL. VoltDB,Clusterix etc. are examples of this storage technique. Query in NewSQL is in the form of relational tables but internally it can store the record in other format also.

3. NoSQL databases categories

NoSQL databases are used for many different applications. Different categories of NoSQL databases are defined based on these domain specific applications which are as follows:

- Column oriented
- Graph based
- Key value
- Document oriented

3.1. Column Oriented

In this storage structure, values are not stored in rows. It is stored based on the values of columns. In traditional relational databases, values are stored in rows, so values are stored as null for columns where values are not known. This drawback is removed by column oriented storage structures. Column data is distributed on different clusters; hence large scale of data can be easily handled. Scalability is improved by using this data storage technique. Many column oriented databases can be easily deployed on Mapreduce[7], hence easily deployed for big data. It is most suitable for data mining applications. Column oriented databases provide better indexing and query structure than key value databases [8]. Google BigTable[9], HBase[10], SciDB, Amazon SimpleDB and Cassandra are examples of column oriented database.

ID	Name	Address	Age
1	Abc	Abc1	25
2	Xyz	Xyz1	28
3	Klm	Klm1	30
4	Wer	Wer1	27

Fig. 2(a): Row oriented data storage

ID
1
2
3
4

Name
Abc
Xyz
Klm
Wer

Address
Abc1
Xyz1
Klm1
Wer1

Age
25
28
30
27

Fig. 2(b): Column oriented data storage

Figure 2(a) demonstrates rows based storage and Figure 2(b) demonstrates storage based on columns**.**

3.1.1. BigTable

This data storage is developed by Google and uses GFS (Google File Systems) [3]. In transaction, data is written until memtable reaches threshold value. Multiple set of data is read at once. BigTable does not support SQL like structure. It is used in Google

app engine. It manages many clusters by using CMS (Cluster Management Systems). Data is stored in SSTable format which is a persistent and ordered map.

3.1.2. Cassandra

This data storage is developed by Apache Software Foundation. This storage technique is implemented in Java. In this data storage structure, data is distributed on multiple nodes. Relational database format is not followed in this storage technique, rather dynamic schema and content is used. Fault tolerance with no single point of failure and high throughput are important features of this storage technique. Social networking sites, banking and finance are the application areas of this storage technique. It supports SQL like query language CQL. It is same as SQL but to implement scalability some features of SQL are not present like Joins, aggregate functions. Values are stored in the form if triple (row, column, timestamp). Its throughput is consistently better than many other databases.

3.1.3. HBase

It is open source implemented in Java and developed by Apache Software. It uses HTTP/REST protocol. Storage is provided by Hadoop Distributed File System (HDFS) and Hadoop MapReduce framework. Search engines and log data analysis are the application area of HBase.

3.2. Graph based

Social networking sites use connection amongst users to provide them information, latest views or recommendations from connected users. This information can not be stored by relational database. Moreover, relational database works only for predefined schema, dynamic schema can be used by graph based databases. Graph based data can be stored using nodes as users and edges as connection amongst users. Semi-structured and unstructured data is well handled by this storage technique [3]. Graph datbases are not suitable for horizontal scaling i.e. when connected nodes are distributed on clusters, it is very difficult to traverse and manipulate graph. Neo4J, OrientDB and InfoGrid are examples of graph based databases.

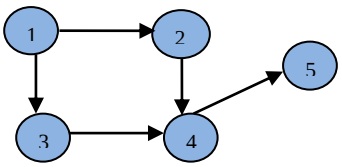

Fig. 3: Graph based data storage

It is clear from Figure 3 that users are represented as nodes and relationships are represented by directed edges. These nodes are connected through direct edges and there is no need to store the index. For example, social connected users on social network can be better represented by this storage technique.

3.2.1. Neo4J

It is the most popular graph based data storage. Entities are represented by nodes and relationships amongst entities are represented by edges. Traditional relational dataset like schema is not present in this data storage. Indexing of nodes is provided by this storage technique. It is compatible with Java, Python and Ruby. Cypher query language is used for finding nodes and edges representations. Its application is in various fields like storing record in healthcare [11] .

3.2.2. OrientDB

Graph based and document based storage are combined in this NoSQL database. It uses the scheme-less as well as scheme-mixed format. Sql queries can also be implemented in this data storage technique.Social networking and recommendations are some of key application area of OrientDB.

3.3. Key Value

This data storage technique is very simple but effective. Data is stored in the form of key which have unique value like Map or dictionary. Key-value pair structure is fast in index i.e. value can be retrieved faster as compared to traditional relational database. Key can not be duplicate and has unique value. Response time for query using this storage technique is very less. Data is stored with schema less format [3].This storage technique is very efficient for distributed storage. In scenario where relations or structures are required, key-value storage is not suitable [8]. Key value store is used in Web sessions or any user specific information sites. The reason is that user data (i.e. value) should not be accessed directly, it must be accessed by the use of key. Dynamo [12] from Amazon, Azure Table and Redis are examples of key value database.

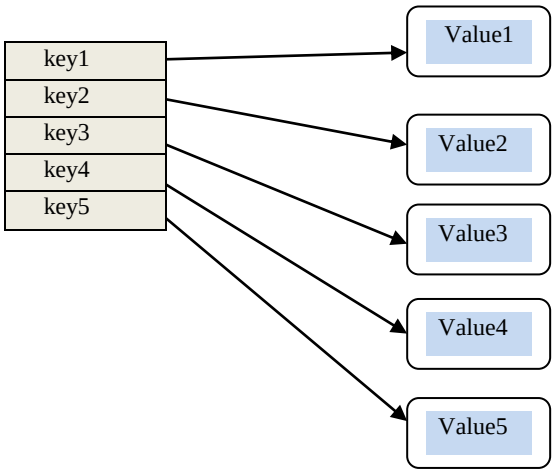

Fig. 4: Key value data storage

In Figure 4, it is clearly explained that for a unique key,there is corresponding unique value.

3.3.1. DynamoDB

It is NoSQL database that provides reliable and cost effective storage. When many replics are not available, even then reliability is provided by this data storage. It uses solid state drives (ssd) instead of hard drives. It is implemented by using Amazon's Dynamo model [3]. Data is stored on multiple data centres (at least three) to provide high reliability. It provides replication with MVCC (multi version concurrency control). Synchronized clusters are also the main characteristics of this storage technique for replicas. These nodes are having equal privileges and roles to provide fault tolerance. The disadvantage is that range queries are not possible.

3.3.2. Oracle NoSQL

Big data storage can be well handled by Oracle NoSQL. It combines oracle and hadoop to store and process unstructured data. JSON format is implemented in this database. It is very simple key-value data storage.

3.4. Document based

Data is stored in the form of documents rather than simple row-column values. XML [eXtensible Markup Language] or JSON [Javascript Object Notation] format is used which stores relevant information in the form of documents. The advantage of using JSON format is that different programming object structure can be easily mapped in this format [1]. Document oriented data storage technique should not be used when there are a lot of relations amongst different tables and normalization is to be incorporated. This storage technique uses dynamic schema, the advantage is new attribute can be easily added for some documents. This is different from relational database fixed schema structure where new attribute is to be added for all records, if values are not known; many null values are to be added. Document based databases provides indexing based on primary key [8]. Blogs and content management systems are easily stored in document oriented databases. MongoDB, CouchDB, OrientDB and MarkLogic are examples of document oriented databases.

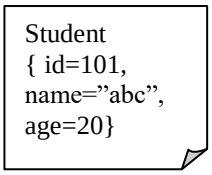

Fig. 5: Document oriented data storage

3.4.1. MongoDB

It is open source document oriented data storage. It works on Master-Slave storage architecture. Master can read and write in the form of documents while slave can only read. Format for storage is BSON (more compact format i.e. Binary JSON) which uses dynamic schema. Fault tolerance is the main feature of MongoDB. MongoDB allows data to be organized in the form of Collections and not on tables. Querying specified record from this collection is used by dot (.) notations. Many programming languages are supported by this storage system. It is widely used for storing records in the form of documents in Healthcare [13]. It can provide high throughput than many other databases.

3.4.2. CouchDB

It is used for implementing web interface using JSON format. It is written in Erlang and not based on schema as in relational database. It is also defined as combination of unreliable clusters. Http and Rest protocol is used in this data storage [4]. Map and Reduce is used for deploying Big data in CouchDB. Javascript query language is used to fetch unstructured data. High scalability and high availability are important features of this storage structure.

4. Comparison of NoSQL databases categories

Several research works have compared NoSQL varieties based on real examples. In this paper, we have compared NoSQL varieties based on storage architecture. It is very easy for reader to differentiate different varieties and select storage techniques best suitable for specific application. Comparison of NoSQL database categories is given in Table 1.

Various applications of NoSQL data storage techniques are elaborated in Table 1. If any application requires social connections analysis, then graph based NoSQL should be used. Programming languages objects can be stored using JSON in Document based storage. If query response time should be reduced then key-value storage should be used. In applications where column values are not known, column oriented storage is best suitable. Main differences in the architecture of NoSQL databases are explained and also most popular solutions are explained in previous sections. It is clear from Table 1 that comparison using category of NoSQL is far much better than comparison using real examples.

Following properties need to be considered for selecting NoSQL solutions based on specific requirements:

1. Data model
2. Schema
3. Distributed storage
4. Query response time
5. Query API

Table 1: Comparison of NoSQL database categories

Parameter	Column Oriented	Graph Based	Key Value	Document Oriented
Storage	Columns	Nodes and edges	Unique Key with value	XML, JSON, BSON
Applications	Sparse data	Social connections	Indexing	Programming objects storage
Examples	Big Table, Cassandra	Neo4J, InfoGrid	Dynamo, Oracle NoSQL	MongoDB, CouchDB
Format	Flexible schema	Dynamic schema	Schema-less	Schema-less
Advantages	Scalability	High fault tolerance	Less query response time	Generalized storage for objects
Flexibility	Average	High	High	High
Performance	Good	Varying	Good	Good
Scalability	High	Varying	High	Varying
Flexibility	Average	Good	Good	Good
Complexity	Low	High	Low	Low

Scalability is important parameter to analyze the processing for large scale of data [14][15]. It is explained in [16] that Big Data frameworks can retain scalability.Scalability in this comparison is not only for read but also for write, as many research works have concluded that write requires more replication storage [17]. Readers can also decide by the use of Table 1 that which schema and format requirement is most suitable for application. Advantages of categories are also described. There are 120 solutions for NoSql databases [18]. It is very difficult to compare with real solutions.

5. NoSQL Query Language

The analysis of query model is very important for any data storage [19]. SQL (Structured Query Language) is the standard used by relational database vendors but, there is no standard query language articulated for NoSQL databases. This can be assumed as disadvantage of this storage technique. UnQL query language is developed to work same as SQL interface, but it is not popularly used till now. Many NoSQL database use its own query language that is not applicable for other types of database. CQL (Cassandra Query Language) for Cassandra, Cypher for Neo4J, MongoDB query language, Javascript for CouchDB are some of examples. It is active research area to define and develop NoSQL databases standard query language. The difficulty in developing standard query language for NoSQL is that every solution is designed for specific purpose and use different formats and schema..Category

based standard language is designed by researchers as same category use same format and schema generally. SPARQL is standard query designed for many graph databases. Command line interface is used in NoSQL and NewSQL data stores [8].

For example, MongoDB query structure is same as Sql[20]. MongoDB query language has find() method which works same as select in SQL. db.collection.find() is used to extract some value from database. In this method, argument is used which works similar as where in SQL.

db.<collection>.find({title= "document_name"})

Documents are inserted in the collection by the use of following query:

db.<collection>.insert(<document>) or
db.<collection>.save(<document>)

Updation in database use following query:

db.<collection>.update(<criteria>)

Following query is used for deletion of document:

db.<collection>.remove(<criteria>)

It does not provide foreign key, rather DBRef is used to refer documents in collection.

Features which are supported by MongoDB query language are as follows:

1. Comparators (<,>,<=,>=)
2. Logical operators
3. Group By
4. Conditional Operators
5. Queries over documents and subdocuments

Cypher is used to query important information from graph database. It is declarative language and can extract information from database without altering graph structure. It is used to extract the relationships amongst nodes and not on actual structure of nodes. In this language, aggregate functions like max, min, sum and count are same as SQL.

6. Conclusion

In this paper, techniques for Big data storage are highlighted. Several techniques and solutions are available for efficient storage of structured, semi-structured and unstructured data. Distributed data storage, NewSQL and NoSQL are most commonly used technique identified in this paper. Applications of these techniques are elaborated with advantages and disadvantages. NoSQL data storage technique is the most popular amongst these solutions. Categories of NoSQL data storage- Column oriented, Graph based, Key value based and document based are explained in detail with real examples. In this paper, these categories are compared based on several parameters. Requirement specific solutions are provided for guidance of reader to select most suitable technique. It is explained that query language is not standardized for NoSQL databases due to different protocols and schema used by four categories. Query languages are briefed for some solutions. In future, these standard query languages can be further researched and developed by readers. Moreover, extra parameters can be added to compare the NoSQL databases based on architecture.

References

[1] Kaur K and Rani R (2013), Modeling and Querying Data in NoSQL Database, IEEE International Conference on Big Data, pp: 1 – 7, DOI: http://dx.doi.org/10.1109/BigData.2013.6691765.

[2] Strozzi C (1998), Nosql – a relational database management system, Lainattu 5, 2014.

[3] Nayak A ,Poriya A and Poojary D (2013), Type of NOSQL Databases and its Comparison with Relational Databases, IJAIS, vol. 5, no. 4,pp.16-19.

[4] Moniruzzaman A B M and Hossain SA (2013), NoSQL Database : New Era of Databases for Big Data Analytics-Classification, Characteristics and Comparison, International Journal of Database Theory and Application ,vol. 6, no.4.

[5] Chen CLP and Zhang CY(2014), Data-intensive applications, challenges, techniques and technologies: A survey on Big Data, Information Sciences, vol. 275, pp. 314-347.

[6] Strohbach M, Daubert J, Ravkin H and Lischka M (2016) , Big Data Storage, In New-Horizons for a Data-Driven Economy , pp.119-141.

[7] Dean J and Ghemawat S (2008), Mapreduce: simplified data processing on large clusters, Communications of the ACM, vol. 51, no. 1, pp. 107–113, http://doi.acm.org/10.1145/1327452.1327492.

[8] Grolinger K, Higashino WA, Tiwari A and Capretz MAM (2013) , Data management in cloud environments : NOSQL and NEWSQL data stores, Journal of Cloud Computing : Advances, Systems and Applications, vol. 2 no.1.

[9] Chang F, Dean J, Ghemawat S, Hsieh WC, Wallach DA, Burrows M, Chandra T, Fikes A, and Gruber RE (2008), Bigtable: a distributed storage system for structured data, ACM transaction on Computer Systems, vol.26 , no.2.

[10] HBase, http://hbase.apache.org

[11] Kaur K and Rani R (2015), Managing Data in Healthcare Information Systems: Many Models, One Solution", Computer, IEEE Computer Society, vol.48, no.3, pp.52-59.

[12] DeCandia G, Hastorun D, Jampani M, Kakulapati G, Lakshman A, Pilchin A, Sivasubramanian S, Vosshall P, and Vogels W (2007), Dynamo: amazon's highly available key-value store, In ACM SIGOPS operating systems review, vol.41 no.6, pp. 205–220.

[13] Kaur K and Rani R (2015), Smart Polyglot Solution for Healthcare Big Data, IT Professional, IEEE Computer Society, vol.17 no.6, pp. 48-55.

[14] Khan N, Yaqoob I, Hashem IAT, ,Inayat Z,Ali W, Kamaledin M,Alam M, Shiraz M and Gani A (2014), Big Data : Survey, Technologies, Opportunities and Challenges, Scientific World Journal , vol. 2014, Article id 712826.

[15] Lourenco JR, Cabral B, Carreiro P, Vieira M and Bernardino J (2015), Choosing the right NOSQL database for the job : a quality attribute evaluation , Journal of Big Data vol.2 , no.1.

[16] Bello-Orgaz G, Jung JJ and Camacho D(2016), Social Big Data: Recent achievements and new challenges, Information Fusion, Science Direct, vol. 28, pp. 45-59.

[17] Strauch C, Sites ULS and Kriha W(2011), NoSQL databases, Lecture Notes,Stuttgart Media University .

[18] Tudorica BG and Bucur C (2011), A comparison between several NoSQL databases with comments and notes, Roedunet International Conference, pp.1-5.

[19] Hecht R and Jablonski S (2011), NoSQL evaluation: A use case oriented survey, In Cloud and Service Computing (CSC) International Conference, IEEE, pp. 336-341.

[20] Tauro CJ, Patil, BR and Prashanth KR (2013). A comparative analysis of different nosql databases on data model, query model and replication model. In Proceedings of the International Conference on ERCICA.