International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 9 • Number 45 • 2016

NoSQL Databases and its Use Cases: Comprehensive Review

Geetika Sharma[a] and Richa Sharma[b]

[a]*Student, Department of CSE, Lovely Professional University, Phagwara, Punjab, India.*

[b]*Assistant Professor, Department of CSE, Lovely Professional University, Phagwara, Punjab, India.*

E-mail: richa20489@gmail.com

Abstract : Data is the most important asset for any company. So a system is required to store and process data efficiently. Database management system has more recently emerged as a fairly standard part of any company back office. But with the development of internet, there arises a need of database which can handle big data, can provide high performance and high availability. Because of the limitations of traditional databases, companies started looking for other alternatives and then NoSQL database comes into existence. This paper firstly describes NoSQL databases and then a comparison is made between traditional SQL and NoSQL database. Finally, different variants of NoSQL database are described along with their respective use cases and strengths which will help companies to select an appropriate database.

Keywords: *NoSQL, big data, database, key-value, column-oriented, document-oriented.*

1. INTRODUCTION

In the past decade or so, relational databases have been the choice of many developers and system administrators. It was the year 1970 when relational database popularly known as SQL Database was firstly proposed. It changed the whole way of storing data and become so popular that from large enterprise to the smaller ones, all shifted to relational database. Relational databases have dominated the industry for a long time providing mechanism to store data, transactions, standard interfaces and mechanism to integrate application data.

But after ruling for last 40 years, there comes a change. With the continuous increase in real-time web applications and development of internet, traditional database failed to meet the needs of this continuous increase of development. This need of handling big data, providing high performance and high availability forced developers to look for an alternative. This was the time when "NoSQL" came to an existence[1,2].

NoSQL is a new way of thinking about database. In the early twenty-first century, NoSQL databases gained popularity when leading companies like Google, Facebook, Amazon, and LinkedIn pioneered this technology. Today, many large enterprises are adopting NoSQL databases, a choice that is driven by the development of Internet. Big data, Cloud Computing, Internet of things are the drivers of NoSQL's rise. NoSQL databases provide certain functionality that traditional databases lack in[3].

Some inherent limitations of relational databases are:

1. Limited Capacity
2. Unable to handle unstructured data
3. Less Flexible
4. Scalability issues

RDBMS does fail in certain aspects but it doesnot mean that they are dead. NoSQL is not a replacement for RDBMS but it compliments in such a way that they can co-exist. However, now RDBMS is no longer an automatic choice.

Characteristics of NoSQL Databases[4]:

1. Schema-less data model
2. Flexible to handle big data
3. Works well with distributed data stores
4. Scales horizontally
5. Faster , more efficient and flexible
6. Doesnot support ACID properties

2. COMPARISON OF SQL AND NOSQL DATABASE

Table 1
Comparison of SQL and NoSQL Databases (5, 6, 9-11, 14)

Features	*SQL*	*NoSQL*
Data Model	Data is stored in tables with rows and columns.	Data is stored as collection like document, key value, graph, etc.
Schema	It has fixed schema *i.e.* pre-defined schema.	It supports dynamic schema.
Type of data stored	Structured	Structured, Semi structured and Unstructured.
Scalability	SQL databases scale vertically.	Horizontal scaling is done in case of NoSQL databases.
Transaction reliability	Guarantee high transactional reliability.	It does not guarantee transactional reliability.
Data warehouse	Serves data warehouse applications.	These databases do not serve data warehouse applications.
To the cloud	Relational storage in the cloud often comes as service that is replicated, highly available and distributed for greater fault tolerance through horizontal sharding technique.	NoSQL databases hosted as a cloud service benefits from inherent auto-sharding, flexible elasticity, integrated caching and tremendous computing power to capture, store and analyze data.

Features	*SQL*	*NoSQL*
Development model	Some of them are open source like Postgres, MySQL and some are closed source like example Oracle DB.	These databases are mostly open source.
Consistency	It can be configured for strong consistency.	Depends on products. Some databases provide strong consistency like MongoDB and some offers eventual like Cassandra.
Features	SQL	NoSQL
Data manipulation	These databases make use of language where insert, select, update like statements are used for manipulation of data.	Here data is manipulated using object oriented APIs.
Types	It has only single type which is SQL databases with minor variations.	It has various types including key value, document, column based, graph databases.
Property supported	Based upon ACID property.	It is based on BASE and CAP.
Recovery	It guarantees crash recovery via recovery manager which is responsible for ensuring transaction atomicity and durability by the use of log files and ARIES algorithm.	It depends on replication as backup to recover from the crash.
Complexity	It is less complex.	It is more complex.
Data capacity	Has limited capacity.	Has more data capacity and can store big data.

3. ADOPTION OF NOSQL DATABASES

Rank			DBMS	Database Model
Feb 2016	Jan 2016	Feb 2015		
1.	1.	1.	Oracle	Relational DBMS
2.	2.	2.	MySQL	Relational DBMS
3.	3.	3.	Microsoft SQL Server	Relational DBMS
4.	4.	4.	MongoDB	Document store
5.	5.	5.	PostgreSQL	Relational DBMS
6.	6.	6.	DB2	Relational DBMS
7.	7.	7.	Microsoft Access	Relational DBMS
8.	8.	8.	Cassandra	Wide column store
9.	9.	9.	SQLite	Relational DBMS
10.	10.	10.	Redis	Key-value store

Figure 1: Ranking of different DBMS(17)

NoSQL databases got popularity in early 21st century with the increase in demands which traditional database does not fulfill. NoSQL is generally referred as "Not only SQL". Increase in the use of NoSQL database does not mean that relational databases are no more. They do exit and will be integral part of any company. Today, most of the Web 2.0 companies make use of both relational as well as nosql databases as both of them have their own specific advantages and disadvantages. So the facilities that relational database does not provide can be compensated with the use of nosql database and vice versa. In this way both types of databases go side by side. Despite of gaining huge popularity in recent years, NoSQL database still lags behind relational database. Most of the large companies have deployed them into their business but small ones still relies on relational databases only. As of February 2016, the ranking of top ten most popular database management systems is shown in Figure 1.

This survey clearly depicts that Relational DBMS still dominates.MongoDB, one of the variant of NoSQL databases, leads among all of them with significant adoption among the Fortune 500 and Global 500. Their lies many metrics which can be used to analyze the leadership of a database. Among those metrics, few are- google searches and job posting which illustrate MongoDB's leadership.

4. CLASSIFICATION OF NOSQL DATABASES

NoSQL databases are classified into to the following categories:

1. **Document :** MongoDB, CouchDB, MarkLogic, OrientDB, Qizx
2. **Column :** Cassandra, HBase, Vertica
3. **Key-value:** Dynamo, FoundationDB, Redis, Riak, Memcached
4. **Graph :** Neo4J, InfiniteGraph, Allegro

4.1. Document-oriented Databases

7Document-oriented databases manage and store data in the form of documents. Documents encode data in certain standard formats or encodings. Encodings include XML, JSON (JavaScript Object Notation), YAML and BSON (Binary JSON). Every document in a database has a unique key. As of February 2016 ranking, MongoDB is the most popular document-oriented database.

MongoDB: MongoDB was developed in year 2007 by the company named MongoDB Inc. The company released its open source development model in 2009. MongoDB is for CIOs, Architects, DBAs and developers. It is adopted by number of companies, websites and services including Metlife, eBay, Otto, Bosch, Expedia, Forbes, ADP and many more[18,19].

One should try out MongoDB if the required technical demands match the following:

1. High Performance both reads as well as writes.
2. Need flexible schema and rich querying.
3. Need for replication across multiple data centers.
4. High Availability.
5. Use it if your data set is going to be big.
6. Real time analysis in the database.
7. Geospatial querying.
8. Need to promote agile coding methodologies.
9. Need strong data consistency.
10. Need advanced security.

These where the technical demands that MongoDB supports but apart from technical various business demands which are supported by it are:

1. Management tooling and services.
2. Ease of hiring.
3. Commercial License
4. Ease of developer adoption.
5. Global support.
6. IT ecosystem integration.
7. Company stability

4.2. Column-oriented database

It is a database management system in which data is stored as columns instead of rows. This way of storing data is very helpful in many of the applications like data analysis, CRM, etc. The column-oriented databases are preferred over row-oriented when the aggregate values are to be computed and further processing is to be done on those values. In this case column-oriented is faster and efficient than row-oriented. It also works efficiently when the same value has to be applied to all the rows in a single go. Cassandra is the most popular column-oriented database as per the February 2016 rankings.

Cassandra: It was first released in July 2008 and was developed at Facebook. Many companies and businesses have adopted Cassandra and it benefited their business in many ways. Different companies and organizations those adopted Cassandra include Accenture, Aeris, EBuddy, Globant, WSO2, EBay, The New York Times, GoDaddy, Healthx, Nasa and many more. Cassandra should be preferred if any organization's demand matches the following[20,21]:

1. Cassandra is best suited for time series data.
2. It is fault tolerant. It replicates data to multiple nodes. So if any node fails, load is transferred to other node.
3. In Cassandra, writes operations are very fast but read might not be that fast.
4. It supports CQL which is SQL like language but many operations like joins, aggregate operations, etc are not supported by CQL.
5. It should not be preferred for those applications where column values are to be updated frequently.
6. It is supported for the applications that can not afford to lose data.
7. In Cassandra, there is no network bottleneck.

4.3. Key-value databases

This type of database maintains hash table which contains unique key which points to a particular data. It has higher performance as compared to traditional databases and provides high availability. But this database also suffers from some of the drawbacks. It does not provide atomicity and consistency, etc. Its complexity increases as the size of data increases because it has to maintain key for every value. Redis is the most popular key-value database according to February 2016 ranking.

Redis: It was initially released in April 2009. Redis[22] means Remote Dictionary Server. Different companies using redis are Twitter, Snapchat, Digg, Flickr, StackOverflow, GitHub, Weibo, Pinterest and many more. It is best suited for an organization whose requirements match the following:

1. Redis can be used for those applications which require very high read and write speed. This better application performance is achieved because of the unique architecture of redis which stores whole data in memory.
2. It can also be used for handling complex problems. These problems can be solved using optimized data structures with greater simplicity.
3. It should not be preferred for the applications whose dataset is very large.
4. Redis easily scales a database.
5. It can be used for use cases like Analytics, Distributed Locks, Caching, Geo searches, Cookie storage, and Content distribution.

4.4. Graph databases

Graph databases are designed to store inter-connected data. The acceptance of these databases increased because of the several factors. Early adopters of graph databases are facebook, twitter and google. Use cases of graph databases are Content management and access control, Insurance risk analysis,Identity and access management, Network and IT operations, Bioinformatics, Network asset management, Geo routing (public transport), web browsing, real-time recommendations and fraud detection. The most popular graph database is Neo4j.

Neo4j: Neo4j[23]is the most popular graph database. Its first version was released in February, 2010. Neo4j is a graph database in which data is stored as nodes and relationships between those nodes. The nodes and relationships both can contain properties and values. Reasons for choosing Neo4j:

1. Neo4j provides high read and write performance and still provides data integrity.
2. This graph database is easy to learn and use.
3. It is easier and will take less time to load large and complex data into Neo4j.

5. CONCLUSION

Based on the description mentioned above, it is important for any company to properly analyze and decide which database will be more beneficial and will suit them the most. For deciding this, many different properties should be considered like performance, data model, capacity, scalability, ACID support, data persistence, availability and business support. Selection of an appropriate database according to the requirement is most important for any organization.In this paper, firstly an overview of NoSQL database, its characteristics and limitations of traditional database is described. Then comparison between SQL and NoSQL database is analyzed. After that brief summary of NoSQL database adoption is mentioned. Finally, categories of NoSQL database, their current mainstream databases and their strengths and various use cases are analyze.

REFRENCES

[1] He, C. Survey on NoSQL Database Technology. 2015;*2*[2]: 50–54.

[2] Gajendran, S.A Survey on NoSQL Databases.*Masters.Donntu.Edu.Ua*.1998.

[3] Indrawan-Santiago, M. Database research: Are we at a crossroad? Reflection on NoSQL.*Proceedings of the 2012 15th International Conference on Network-Based Information Systems, NBIS 2012*, 45–51.

[4] Moniruzzaman, A. B. M., &Hossain, S. A.Nosql database: New era of databases for big data analytics-classification, characteristics and comparison. *Nosql Database: New Era of Databases for Big Data Analytics-Classification, Characteristics and Comparison*. 2013; *6*[4]: 1–14.

[5] Nayak, A., Poriya, A., &Poojary, D. Type of NOSQL Databases and its Comparison with Relational Databases. *International Journal of Applied Information Systems*.2013; *5*[4]: 16–19.

[6] YishanLi;SathiamoorthyManoharan. A performance comparison of SQL and NoSQL\ndatabases. *978-1-4799-1501-9/13-2013 Ieee*: 15–19.

[7] Bajpayee, R. Big Data : A Brief investigation on NoSQL Databases.2015;*4*[1].

[8] Leavitt, N. Will NoSQL Databases Live Up to Their Promise? *Computer*.2010;*43*[2]: 12–14.

[9] Mohamed, M. a., Altrafi, O. G., & Ismail, M. O. Relational vs. NoSQL databases: A survey. *International Journal of Computer and Information Technology (IJCIT)*.2014;*03*(03): 598–601.

[10] Li, Y., &Manoharan, S. A performance comparison of SQL and NoSQL databases.*IEEE Pacific RIM Conference on Communications, Computers, and Signal Processing – Proceedings*.2013: 15–19.

[11] Mohamed, M. a., Altrafi, O. G., & Ismail, M. O. Relational vs. NoSQL databases: A survey. *International Journal of Computer and Information Technology (IJCIT)*.2014;*03*(03): 598–601.

[12] Oracle. Oracle: NoSQL Database: an Oracle White Paper. *Oracle White Paper*.2011 September;12.

[13] Brewer, E. CAP twelve years later: How the "rules" have changed. *Computer*.2012;*45*[2]: 23–29.

[14] Medero, H., & Mitchell, H. Comparison of NoSQL and SQL Databases in the Cloud.2014: 1–8.

[15] A, V. A., A, J. B., & B, P. F. Which NoSQL Database? A Performance Overview.*Open Journal of Databases (OJDB)*.2014;*1*[2]: 17–24.

[16] .Bhogal, J., &Choksi, I. Handling Big Data Using NoSQL. *Proceedings - IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, WAINA. 2015*: 393–398.

[17] Ranking of Databases [Internet]. Available From: http://db-engines.com/en/ranking

[18] MongoDB [Internet]. Available From: https://www.mongodb.org/

[19] Gyorodi, C., Gyorodi, R.,Pecherle, G., &Olah, A. A comparative study: MongoDB vs. MySQL. *2015 13th International Conference on Engineering of Modern Electric Systems, EMES 2015*: 0–5.

[20] Cassandra [Internet]. Available From: http://cassandra.apache.org/

[21] Wang, G., & Tang, J. The NoSQL principles and basic application of cassandra model. *Proceedings - 2012 International Conference on Computer Science and Service System, CSSS. 2012*: 1332–1335.

[22] Redis [Internet]. Available From: http://redis.io/

[23] Neo4j [Internet]. Available From: http://neo4j.com/