ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: https://doi.org/10.29121/granthaalayah.v7.i7.2019.777

INTERNATIONAL JOURNAL OF RESEARCH – GRANTHAALAYAH
A knowledge Repository

Science

# CASE STUDY OF TRADITIONAL RDBMS AND NOSQL DATABASE SYSTEM

Yashraj Sharma [1], Yashasvi Sharma [1]
[1] Department of Information Technology, Acropolis Institute of Technology and Research, India

## Abstract

On the basis of reliability, rational models are useful but not in terms of systems which involve huge amount of data; in such cases, non-relational models are much more useful. To store large chunks of data, NoSQL databases are used. NoSQL databases are scalable and wide ranged because they are non-relationally distributed.

In relational databases, it was not possible to manage data which involved very large number of Big Data applications hence the concept of NoSQL database was introduced.

There are a lot of advantages of NoSQL which not only involve its own features but also some features of relational database management system. The severe benefit of NoSQL database is that it is an open source system which helps to adapt many numbers of features for newly generated applications.

This paper is focused on understanding the concepts of non-relational database system architecture with relational database system architecture and figure out the advantages and disadvantages of both simultaneously

***Keywords:*** Database; NoSQL; RDBMS; ACID; BASE; CAP.

***Cite This Article:*** Yashraj Sharma, and Yashasvi Sharma. (2019). “CASE STUDY OF TRADITIONAL RDBMS AND NOSQL DATABASE SYSTEM.” *International Journal of Research - Granthaalayah*, 7(7), 351-359. 10.29121/granthaalayah.v7.i7.2019.777.

## 1. Introduction

### 1.1. RDBMS

Relational database management system helps a user to execute queries of a traditional database system such as insert, update, delete and modify. It involves a user with structuring and interact with the database. RDBMS is the basic unit behind all the modern database software such as ORACLE, My-SQL, MS SQL Server. It arranges information into allied rows and columns. In a model like RDBMS, gathering of data is much slower when compared to other database models. In RDBMS, data is not stored in a single table instead; it is stored in the form of rows and columns for better spreading of datasheets. Redundancy of data can be obtained in this model.

## 1.2. No SQL Database

NoSQL is a type of non-relational database which does not require any fixed operations unlike relational databases. It helps in better mounting of data. Platforms such as Big Data and website applications use NoSQL for storing their data. The naming of this database might be a little confusing but NoSQL stands for “Not Only SQL” which means that this database is not only replacing RDBMS but also harmonizing some key features to it. NoSQL databases are used in real-time web applications and big data and their use are increasing over time. NoSQL systems are also sometimes called Not only SQL to emphasize the fact that they may support SQL-like query languages.

For problems such as tempo, pliability and scalability, developers usually focus on implementing simple and easy solutions. For better understanding of key solutions, NoSQL is diversified into 4 types stated as: Column, Key values, Documents and Graphs. On the basis of type and requirement, selection of the following can be done accordingly. There is a term named transaction which is considered to be a very tiny unit and its purpose is to handle many minute-level tasks. In order to gain accuracy there are some properties which are to be maintained by transaction. These properties are called ACID properties. ACID property helps to handle integrity of data and its full time completion. Whereas, NoSQL depends on a much lighter model known as BASE model. BASE model helps to provide flexibility to the NoSQL data so that the data might be structured properly.

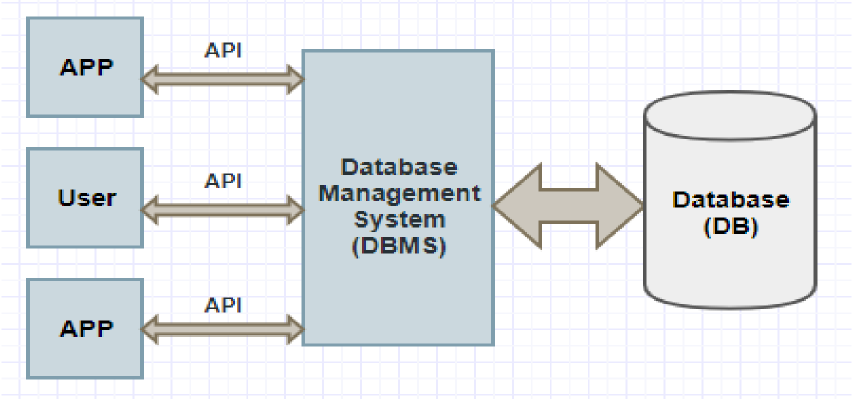
## 2. Literature Review

Innocent Mapanga, Prudence Kadebu(2013) studied Database Management Systems: Analysis of NoSQL system was done. RDBMS included many problems of security, network, mounting limitations, consistency management, and performance. These problems were getting very difficult to be solved. These challenges were highly abrupt when it came to the storage of databases. Though it could be seen that DBMS still had factors to obtain desirable integrity and security. Not only RDBMS but NoSQL also faces huge challenges in adopting a database. The problems basically dealt with serving the data which was to be transited. For problems like this, it is harder to achieve reliability for transited data thus making NoSQL not at all ideal. In such cases, a revisit to previous technique i.e. RDBMS could be done. By looking on to these disadvantages, it is clear that both RDBMS and NoSQL are made for different uses and hence they can’t be considered as opponents.

Salah A. Jowan1, Ramadan Faraj Swese, Antisar Yousf Aldabrzi, Mahmood Saad Shertil (2016) studied TRADITIONAL RDBMS TO NOSQL DATABASE: NEW ERA OF DATABASES FOR BIG DATA. It is stated that the new generation of more complex and unique databases require severe attention to themselves as the data these days is much larger and it is generated by either user or machine. There can also be seen that the cloud storage for multiple databases is also in high demand which increases the capacity of cloud itself. Hence, a new variety of database is introduced named NoSQL, this database promises to overcome all the disadvantages of RDBMS and obtain all the modern techniques for storage of data such as flexibility and scaling. People call NoSQL databases as cloud, non-relational, distributed or Big Data databases. For web and mobile applications, NoSQL is considered to be the alternate source of relational database.

Mohamed A. Mohamed, Obay G. Altrafi, Mohammed O. Ismail (2014) studied Relational vs. NoSQL Databases: A Survey. NoSQL databases and why many of big companies using them. In traditional database management systems, there were many issues regarding the structure of schema, methods of transactions, complex databases, crashing of a database and dealing with huge chunks of data stored within a file. Thus, it was important to obtain NoSQL as a feature which resolved all the disadvantages stated above. But NoSQL lags behind in terms of security and security is the most undertaken factor today. So, it can be considered that NoSQl hasn't been considered as a primary method because it is an open source database and is not as secured as RDBMS.

H.R. Vyawahare, Dr. P. P. Karde, Dr. V. M. Thakare (2017) studied Brief Review on SQL and NoSQL. It is stated in the paper that the data stored in a database is inversely proportional to the performance of that particular database hence, it can be understood that storage has a huge impact on the performance occurring in the database. While keeping this theorem in mind, it can be possible that the time taken to execute a query might get slow. This can be improved by nourishing it with the concept of Horizontal Scaling provided to the SQL by the model; NoSQL.



## 3. Techniques Used

### 3.1. ACID Properties

The modification and accessing of a unit amount of data logically is done by transaction. In transaction, the data is accessed by reading and writing operations.

ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: 10.5281/zenodo.3364448

There are some properties of transactions which should be followed in order to obtain and maintain the consistency of the database. In a database, the system has to consider all the important points for maintaining its priorities say; Atomicity, Consistency, Isolation and Durability. These four important priorities are said to be the ACID properties of transaction.

### 3.2. Atomicity

The concept of Atomicity could be in such a way that everything in a database must be considered as a whole i.e. one atomic unit which means that either one operation should be executed at a time or none. While executing a database, there should not be any case which should be partially left in the transaction. If the transaction is getting executed, aborted or failed, the states should be defined at prior notice.

### 3.3. Consistency

The basic principle of consistency is that no transaction should take place when there is another transaction which is running already hence, the execution should be isolated. For example, if a shopkeeper wants to return $600 to two of his customer. So it can be observed that there are two transactions here. If these transactions run at the same time (concurrently) then both the transactions will read $600 and in that case, the first customer may be $400 instead of $300. This should not happen. If the transactions were isolated, then both the transactions might have read the correct balance since first would've obtained $300 and then the remaining $300 would be given to the second customer.

### 3.4. Isolation

When there is a scenario where there are two or more transactions running at the same time, there should be a property of isolation which states that all of these transactions remain isolated from each other. The responsibility of isolation is to ensure that when two or more transactions run subsequently, they can occur at a same time without harming each other by causing inconsistency between them. Transactions should occur without any interference i.e. independently. The procedure of isolation is in such a way that if there is a change in one transaction, it will not be visible any other transactions until and unless it is written to the memory or else it has been committed.

### 3.5. Durabilty

In the case of failure or restart of a system it should be ensured that the systems hold all of its information that was stored before. This is the factor which helps to ensure the durability of a system. If there is a case that while a data is updated, the system crashes then that particular data will be updated when the system starts again.

### 3.6. BASE

The properties required to access NoSQL databases is called BASE. Basically it acts as ACID for non-transitional databases but it is usually referred as the opposite of ACID.

Whenever there is a possibility where ACID model is not able to handle any uncertainty of the database, NoSQL is used. So, for assurance of operations, NoSQL uses a model known as BASE model, which has three basic principles:

- Basic Availability: Even if there are multiple failures in the database, NoSQL aims to avail as much data as possible. The availability of data is conquered by applying a high distributed approach to the management system. NoSQL does not focuses on the storage of a single data and handling its fault tolerance instead, it spreads the data to other storage systems resulting in high replication degree. When a failure contradicts a portion of data, it is unlikely to happen that the complete database is destroyed.
- Software State: BASE databases abandon the consistency requirements of the ACID model pretty much completely. The requirements of ACID model are abandoned completely by BASE. BASE's main principle is that it does not let the database handle consistency problem; it lets the developer to handle consistency.
- Eventual Consistency. Consistency in a NoSQL database is only required when in some point in the future, data will try to get into a consistence state. While the occurrence of any event, no guarantee of consistency is made. This completely changes the concept of consistency which was required in ACID model as it stops the transaction from executing unless the previous transaction is completed.

### 3.7. CAP

CAP theorem says that in the theoretical field of computer science, it is not possible for a database which is distributed, to give more than two out of the given three properties:

- Consistency —It is a type of guarantee which says that every change in database will always result in successful write. When every client has a same view for data, it is referred as consistency. There are two very strong types of consistencies used to prove CAP theorem named linearizability and sequential consistency.
- Availability — If there is a non failed node, it will always respond to all the read and write requests in a certain amount of time. It is important for every node to be able to respond in a certain time to be available.
- Partition Tolerant —.Instead of network partitions, the systems sticks on to hold its consistency. Network partitions can be considered as a fact of life. Once the partition is healed, distributed systems can guarantee that partition tolerance can be recovered from them.

The meaning of C and A in ACID is quite different than the meaning of C and A in CAP theorem.

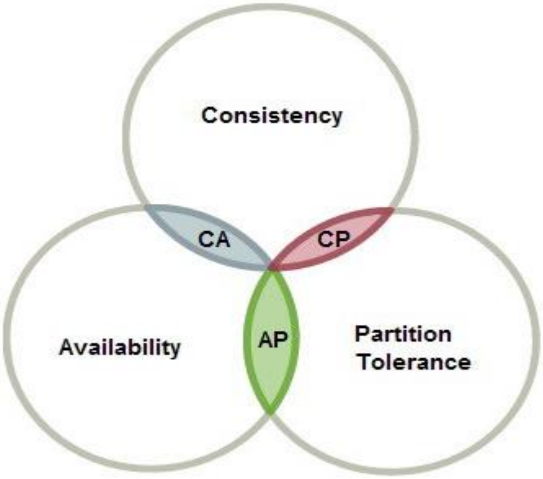
The CAP theorem is divided into in to three system categories:

- CP (Consistent and Partition Tolerant) —CP category could be a little confusing to understand since there are very less chances for a system to be consistent and partition tolerant. To make it simpler to understand, it could be put in a way that the availability of a system is sacrificed only when there is network partition.
- CA (Consistent and Available) —CA systems are referred to the systems which are consistent and available when there is no availability of network partition. A single node DB server can also be put as a CA system. Since, single node database servers never deal with partition tolerance therefore, they are taken in to the CA systems. The only problem

ISSN- 2350-0530(O), ISSN- 2394-3629(P)
DOI: 10.5281/zenodo.3364448

with this theory is that the single node database systems do not lie into the view of CAP because they are not a network of shared data systems.

- AP (Available and Partition Tolerant) — The systems which are available and partition tolerant but can never ensure consistency are called AP (Available and Partition Tolerant).



## 4. Advantages and Disadvantages

Advantages of NoSQL Database

- NoSQL databases can  be scaled easily.
- The administrators for database are not required.
- The data which is either structured, semi structured or not at all structured can also be stored.
- There are some databases which are unique enough to undertake hardware failures.
- The basic principles such as speed, efficiency and flexibility are obtained.
- These are the best solution to the problem of cloud databases because they can handle large amount of data.
- The database for clouds can be handled best with NoSQL databases.
- In terms of performance and retrieving of huge chunks of Big Data, NoSQL plays a significant role by providing methods to handle such sort of data.
- When the data of a database is been structured into a table, there may be a possibility of high complexity faced and this could possibly slow down the system. In case of NoSQL databases, data can be stored in structured, semi-structured and unstructured format.

Disadvantages of NoSQL Database

- No standard query language is provided.
- Maintenance of NoSQL databases is very difficult.
- There may be a possibility in NoSQL that some databases do not cover ACID properties.
- Security is the biggest issue of NoSQl databases. There are much of varied databases due to which there is no single solution provided for different purposes.
- There are many modeling techniques in NoSQL which might be very confusing such as key value, graph and document data model.

- NoSQL databases came up with many solutions concerning the storage of big data with the help of maintaining security with them but there has not been enough innovation in increasing these storage techniques.

Advantages of RDBMS

- As compared to NoSQL, RDBMS is quite easy to use.
- It is much more secured than non relational database systems.
- The manipulation of data can be done quite easily.
- The repetition, redundancy and replication of data is reduced in high amount.
- Helps to achieve data integrity.
- The data independence can be better in RDBMS.
- One data can be views by multiple users at multiple times, which shows that Logical data independence is also achieved in RDBMS.
- Recovery and backup is much better.
- Multiple interfaces are given in RDBMS.
- In RDBMS, multiple data can be accessed by multiple users which is not possible in DBMS.

Disadvantages of RDBMS

- The software for a RDBMS is very expensive.
- When the software gets complex then the hardware requirements also increase, resulting in increase of overall cost.
- There is a lot of skilled human resources required to construct a RDBMS.
- There might be some applications which are very slow to process.
- When the data gets lost, it is very hard to recover it.

## 5. Result

In relational databases, it is very huge to achieve the property of scalability and elasticity. It was also observed that if there was a small change in one table then the overall system will also be changed. It was taken as an observation that the response time of a SQL query may change in many factors one of them being scalability. The factors like these are also interlinked with the dependency of data volume in performance. There is a severe problem of fixed set of columns in SQL which is not at all suitable for Big Data queries and thus, there is a high demand of non-relational databases these days.

## 6. Conclusion

The various problems faced in realtional databases are overcome by NoSQL databases. The techniques which NoSQL provides are related to high scalability, availability and fault tolerance. For each requirement, different data stores different techniques in order to achieve their aim. The severe benefit of NoSQL database is that it is an open source system which helps to adapt many numbers of features for newly generated applications. RDBMS is the basic unit behind all the modern database software such as ORACLE, My-SQL, MS SQL Server. NoSQL databases are used in real-time web applications and big data and their use are increasing over time. NoSQL is diversified into 4 types stated as: Column, Key values, Documents and Graphs. Each database has its own advantages.

NoSQL depends on a much lighter model known as BASE model. BASE model helps to provide flexibility to the NoSQL data so that the data might be structured properly. The CAP Theorem states that, you can only have two out of three guarantees amongst the following order of distributed system of data across a write/read pair. The order is; Consistency, Availability and partition tolerance.

## 7. Future Scope

Various data models can be selected by the user wherever needed. It can be observed later that there is some major difference between speed and reliability. NoSQL could be much more flexible as compared to relational databases. The amount of data present in a SQL document is inversely proportional to its performance which reduces the speed of execution of a query, this can be improved by denoting a proper horizontal scaling to the SQL.The only solution which would be easier in the future for relational and non relational databases is to provide an extension on RDBMS which will allow to deliver for non relational data. In future, we will be able to accommodate the features of RDBMS and NoSQL into one single database model which will be much easier for any database to be executed. The main advantage of NoSQL database is that there are many varieties of databases which are available to the developers to install a database for their system and implement it outside the legacy of it. These days people do not depend on relational model when they have an alternative approach towards the non relational one. RDBMS will never vanish away as there are still some of its features which are quite useful and may not be accomplished by NoSQl. The requirements of storage for the upcoming generation are different from the legacy applications. Variation in the key value of database will continue to evolve as the performance and scalability increase.

## References

[1] Behjat U Nisa, A comparision bw relational database and no sql database, International Journal of Trend in Scientific Research and Development (IJTSRD) ISSN: 2456-6470, March – April 2018.

[2] H. R. Vyawahare, Dr. P. P. Karde, Dr. VM Thakare, Brief Review on SQL and NoSQL, International Journal of Trend in Scientific Research and Development (IJTSRD) ISSN: 2456-6470, Nov-Dec 2017.

[3] Innocent Mapanga, Prudence Kadebu, Database Management Systems: A NOSQL Analysis, International Journal of Modern Communication Technologies & Research (IJMCTR) ISSN: 2321-0850, September 2013.

[4] Rabi Prasad Padhy, Manas Ranjan Patra, Suresh Chandra Satapathy, RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database's, INTERNATIONAL JOURNAL OF ADVANCED ENGINEERING SCIENCES AND TECHNOLOGIES Vol No. 11, Issue No. 1, 015 – 030 ISSN: 2230-7818, 2011.

[5] Mohamed A. Mohamed, Obay G. Altrafi, Mohammed O. Ismail, Relational Vs. NoSQL databases: A survey, International Journal of Computer and Information Technology (ISSN: 2279 – 0764), MAY 2014.

[6] Salah A. Jowan, Ramadan Faraj Swese , Antisar Yousf Aldabrzi1, Mahmood Saad Sherti, TRADITIONAL RDBMS TO NOSQL DATABASE: NEW ERA OF DATABASES FOR BIG DATA, Journal of Humanities and Applied Science (JHAS), December 2016.

[7] Pavel Seda, Jiri Hosek, Pavel Masek, Jiri Pokorny, Performance testing of NoSQL and RDBMS for storing big data in e-applications, ResearchGate, April 2018.

[8] Jardel Ribeiro, Jonas Henrique, Rodrigo Ribeiro and Rosalvo Neto, NoSQL vs Relational Database: A Comparative Study About the Generation of Most Frequent N-Grams, ResearchGate, November 2017.

[9] Dr. Alshafie Gafaar Mhmoud Mohmmed, Prof. Saife Eldin Fatoh Osman, SQL vs NoSQL, Journal of Multidisciplinary Engineering Science Studies (JMESS) ISSN: 2458-925X Vol. 3 Issue 5, May – 2017.

[10] Sourav Mukherjee, The battle between NoSQL Databases and RDBMS, ResearchGate May-2019.

*Corresponding author.
*E-mail address:* yashrajsharma141@ gmail.com/sharmayashasvi.124@ gmail.com