

# *Analysis of NoSQL database based on types, characteristics and performance: A review*

Pragya Kamal

(Assistant Professor)

Computer Engineering department  
College of Technology,  
GBPUA&T, Pantnagar  
pragya.kamal.cs08@gmail.com

Sukhwinder Singh

(Assistant Professor)

Computer Engineering department  
College of Technology,  
GBPUA&T, Pantnagar  
sukhwinder.cup@gmail.com

Deepak Kumar

(Assistant Professor)

Computer Engineering department  
College of Technology,  
GBPUA&T, Pantnagar  
deepakchauthary008@gmail.com

**Abstract:** *NoSQL is a non-relational database the board system which differs from relational database the board system from various perspectives. It don't use a forbidden structure for the capacity of data and SQL request for the recuperation of data as relational database the executives system does for the most part [1]. It don't seek after any development that is the reason it is furthermore called creation less or not fixed example database system. It can deal with semi-organized and unstructured data in all regards beneficially and relational databases have issue while taking care of it.*

**Keywords—***NoSQL, SQL, relational database, non-relational database, ACID properties, MongoDB*

## I. INTRODUCTION

NoSQL is a non-relational database management system which varies from relational database management system from numerous points of view. It don't utilise tabular structure for capacity of information and SQL inquiries for recovery of information as relational database management system does generally [1]. It don't pursue any construction that is the reason it is additionally called composition less or not fixed pattern database system. It can handle semi-structured and unstructured information in all respects productively, and relational databases have issue while handling it.[2]

NoSQL initially begun off an essential mix of two words 'No' and 'SQL' clarified as the ellipsis of not just SQL [3]. Henceforth, NoSQL is a nonexclusive term used to allude to any information store or procedure that does not pursue the customary model of relational database management system [4]. There are four fundamental sorts of NoSQL databases incorporates key-esteem store, report based store, and diagram based store [5]. The key-esteem store NoSQL database basically, utilises a hash table in which there exists an interesting key and incentive to particular information. The qualities are recognised and recovered through a key and put away qualities can be numbers, strings, JSON, XML, HTML,

pairs, pictures, recordings and a couple of others [4]. Report Store NoSQL Database, stores each record and information inside a solitary archive. A record store NoSQL database is utilised for putting away, recuperating, and handling semi-structured information [5]. In the column-oriented NoSQL database, information is put away in cells assembled in columns of information. Columns are consistently gathered into column families. A Graph-based NoSQL database that utilises connections and hubs to speak to and store information. NoSQL database system are rising be-side primary web and IT organisation, for example, Google, Amazon, Facebook, Alibaba, IBM; which organisation are managing gigantic measure of information with conventional Relational Database System couldn't handle. In this manner, the point of the investigation was to support clients, mainly to acquire an autonomous understanding of the qualities and shortcoming of NoSQL database approach and where we ready to im-demonstrate for overseeing gigantic volume of information.

## II. METHOD

### a. Article searching procedure

We utilised a systematic searching method to distinguish the majority of the accessible articles that talk about putting away and overseeing information for Big Data circumstance utilising NoSQL databases. In our systematic searching method, we sought two catchphrases from the Springer Link and IEEE Xplore advanced databases to evaluate the article. Right off the bat, we utilised the slogan "NoSQL" to discover diary articles distributed in the English language between years 2011 to 2017. We at that point utilised the watchword "Enormous Data" inside got set of indexed lists to limit the arrangement of broke down diary article additionally. Keeping up the Integrity of the Specifications

### b. Article inclusion and exclusion benchmark

For the last inclination of articles, we connected the NoSQL database framework for Big Data the executives. We utilised some benchmark to incorporate and avoid articles from the arrangement of articles that were chosen through the inquiry of IEEE Explore and Springer Link online databases. To incorporate and prohibit articles from the arrangement of articles found through our precise seeking system, we read the title, abstract, methodology and results of each article. We considered just those articles that were written in English and that utilised NoSQL Databases. The avoidance criteria were the accompanying: 1) Article that connected NoSQL database other than Big Data the board, 2) Article that oversaw Big Data utilising other than NoSQL database.

III. COMPARATIVE STUDY

References	Approach	Study
[5,9-15]	A general overview of NoSQL and Big Data	This study suggests the appropriate databases for explicit sort of use prerequisite and spotlight on the esteem chain of Big Data.
[16-20]	Performance comparison and evaluation of different NoSQL Databases.	Discussion on NoSQL best use cases and NoSQL Databases performance measurement.
[21-22]	Classifying NoSQL Databases according to the CAP theorem and Data Model.	This study describes the background basics characteristics data model of NoSQL.
[23]	DB-Engine Ranking	This research has concentrated on the storage aspect of cloud computing systems, in particular, NoSQL Databases
[24-25]	Transactional for MongoDB, Riak and NoSQL 's SQL condition based on Espresso Heuristic algorithm	To analyze the effects of the transaction on data consistency and efficiency and SQL query condition transformation for any NoSQL Databases

IV. CHARACTERISTICS OF NO-SQL DATABASE

The vast majority of the conventional database framework depend on exchanges. These transactional highlights are likewise well-known as ACID (Atomicity, Consistency, Isolation, Durability) [26]. Nonetheless, Big transactional process does not work legitimately with ACID framework [27]. Subsequently, ACID framework appeared to be an issue in various dispersed frameworks that are not entirely resolvable. In this manner, Eric Brewer [28] presented the CAP hypothesis (Figure 1) which is progressively productive in various dispersed frameworks. In any case, later the investigation [26] noticed that the CAP hypothesis is acknowledged just two properties among the three prerequisites (AP, CP, CA) for Big Data preparing at once. The more subtleties are the following:

- Available and Partition-Tolerant (AP): Achieve "eventual consistency" through emphasis and validation. Model: Voldemort, Couch DB, Cassandra and so on.
- Consistent and Partition-Tolerant (CP): CP framework experience difficulty with accessibility while keeping information reliable crosswise over apportioned hubs. Model: MongoDB, Redis, BigTable and so forth.
- Consistent and Available (CA): CA framework has issue with segments and ordinarily manage replication. Precedent: Vertica, MySQL and so on.

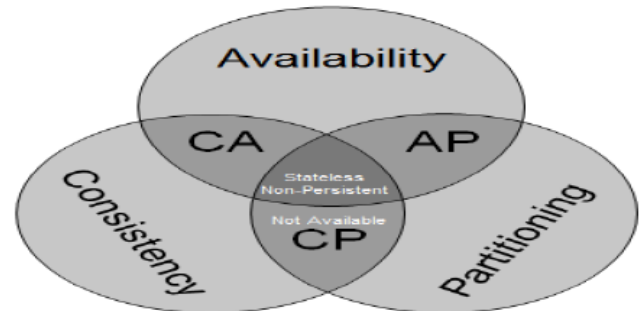


Fig. 1: Illustration of the CAP theorem

For example, when transactional and ACID issues are coming in NoSQL database, there is no other option without CAP theorem. Later, Gonza-lez-Aparicio et al. [25] developed a new transaction system using three components: i) Transmission Processing Engine (TPE), ii) Data Management Store (DMS), and iii) Times Stamp Manager (TSM). Also, they implemented their developed technique into NoSQL databases for MongoDB and Riak; where TPE allows join operation which is not previously supported in NoSQL, whereas DMS and TSM are applied in order to provide high scalability and concurrency of transactions. Other studies [24] described the Espresso heuristic algorithm for converting SQL syntax to a conditional expression of a specific NoSQL Database (MongoDB).

V. WHY NO-SQL?

On the off chance that customarily we have social databases to manage information, at that point why we need NoSQL databases. The answer to this inquiry is given in the following focuses:

1. **Horizontal Scalability or Scale-out:** Scaling on a level plane method adding new or more hubs to a framework. For example, adding a new PC to the appropriated framework.
2. **Workload Distribution:** NoSQL pursues disseminated engineering so outstanding burden is dispersed among different hubs. No inflexible pattern.
3. **Big data Applications:** Huge information is term utilised for large or complex datasets made up of an assortment of information structures including organised, semi-organised and unstructured information. RDBMS can't deal with it so NoSQL required. NoSQL is utilised as being developed of continuous applications.
4. **Performance:** NoSQL has better performance even at a high volume of data.
5. **Continuous Availability or Fault Tolerance:** There is no single point of failure in NoSQL as it follows distribution architecture.
6. **Higher Throughput:** Better throughput than Relational Database Management Systems.
7. **No Complexity:** There will be no complexity and cost of setting clusters.

## VI. TYPES OF NO-SQL

### Key-Value Databases:

- a. A key-value database is a combination of two main attributes: key and value that is key correspond to a value or group of benefits.  
For example:
- b. It is based on the hash table where the key is unique and points to value.
- c. It is a dictionary data structure used for insertion, deletion and search by unique key. E.g. hash table and binary tree.
- d. A key can point to a different number of attributes.
- e. This database support scalability, i.e. can handle a large amount of data.
- f. It is one of the simple databases among all of them.
- g. It supports unstructured data.
- h. It is useful for quickly finding data from the database.
- i. It uses less memory to store data.
- j. Better performance can be achieved by using these databases.
- k. E.g. of key-value databases :

Redis, Riak, Memcached, BerkeleyDB, Tokyo cabinet.

- l. If values are missing, then this will be Ok in case of key-value databases.

Unique key

No schema imposed

Fig 2: Demonstration of how data stored in Key-Value databases.

### 2. Document Databases :

- a. The document database is consist of two main attributes: key and document, i.e. key corresponds to one document.
- b. Key used is unique key, i.e. different key is used for the different document.
- c. In this database data is stored in the form of the document.
- d. This kind of databases support :

Structured data, Unstructured data(text) or semi-structured data (XML).

- e. The technique of hashing is used in document store databases.
- f. Data can be inserted, deleted and updated in the document using unique key pointing to document.
- g. E.g. of document database:

CouchDB, MongoDB

- h. The JSON format is used in the document store database with the dynamic schema. Dynamic schema means different documents can have the different number of fields[7].
- i. JSON stands for JavaScript Object Notation. JSON supports all basic data types: numbers, strings, Boolean, arrays.

### 3. Column Oriented Databases :

- a. Column-oriented databases are also known as Extensible Record Databases or Column family[6].
- b. These databases also contain rows and columns like relational databases. Column is a key-value pair in these databases.
- c. Each column is stored in a separate file, and for every row, column is at the same offset that is called auto indexing [6].

### 4. Graph Databases :

- a. Graph databases are based on the concept of graph theory.
- b. In these databases, data is represented in the form of graphs on behalf of interconnection between data.
- c. Graph database is a collection of nodes and edges where
  1. Nodes represent entities.
  2. Edges represent relationship among nodes.
- d. These databases are used for shortest path calculation.
- e. These databases are scalable in nature but complexity also get increased.
- f. E.g. of graph databases : Neo4J

## References

- [1] C. Strauch, "NoSQL Databases" Available: <http://www.christofstrauch.de/nosql dbs.pdf>.
- [2] Priyanka, AmitPal, "A Review of NoSQL Databases, Types and Comparison with Relational Database"
- [3] M. Stonebraker, SQL databases v. NoSQL databases, *Commun. ACM*, 2010, 53(4):10-11. <https://doi.org/10.1145/1721654.1721659>.
- [4] Tech Republic. 10 things you should know about NoSQL databases. <http://www.techrepublic.com/blog/10-things-you-should-know-about-nosql-databases/>. Accessed August 8, 2017.
- [5] J. Bhogal and I. Choksi, Handling Big Data Using NoSQL. in *Proceedings - IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, WAINA 2015*, 393-398: IEEE
- [6] Daniel Abadi, Peter Boncz, Samuel Madden "The Design and Implementation of Modern Column Oriented Database Systems" Available : <http://db.csail.mit.edu/pubs/abadi-column-stores.pdf>
- [7] Stefan Edlich, "List of NoSQL Databases" Available: <http://nosql-database.org>.
- [8] J. Bhogal and I. Choksi, Handling Big Data Using NoSQL. in *Proceedings - IEEE 29th International Conference on Advanced Information Networking and Applications Workshops, WAINA 2015*, 393-398: IEEE
- [9] R. Zafar, E. Yafi, M. F. Zuhairi, and H. Dao, Big Data: The NoSQL and RDBMS review, in *ICICTM 2016 - Proceedings of the first International Conference on Information and Communication Technology*, 2017; 120-126.
- [10] I. Chebbi, W. Boulila, and I. R. Farah, *Big Data: Concepts, Challenges and Applications*, Springer, Cham, 2015 638-647.
- [11] S. D. Kuznetsov and a. V. Poskonin, NoSQL data management systems. *Program. Comput. Softw.* 2014 40(6): 323-332. <https://doi.org/10.1134/S0361768814060152>.
- [12] V. Rajaraman, Big data analytics, *Resonance*. 2016 21(8):695-716.
- [13] M. Chen, S. Mao, and Y. Liu, Big data: A survey, in *Mobile Networks and Applications*, 2014; 19 (2):171-209. <https://doi.org/10.1007/s11036-013-0489-0>.
- [14] F. Gessert, W. Wingerath, S. Friedrich, and N. Ritter, NoSQL data-base systems: a survey and decision guidance, *Comput. Sci. - Res. Dev.* 2017; 32(3-4): 353-365.
- [15] L. Wu, L. Yuan, and J. You, Survey of Large-Scale Data Management Systems for Big Data Applications, *J. Comput. Sci. Technol.*, 2015; 30(1): 163-183. <https://doi.org/10.1007/s11390-015-1511-8>.
- [16] S. Swaminathan and R. Elmasri, Quantitative analysis of scalable nosql databases, 2016 IEEE International Congress on Big Data (BigData Congress), 2016: IEEE.
- [17] P. Srivastava, S. Goyal, and A. Kumar, Analysis of various NoSql database, 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), 2015: IEEE. <https://doi.org/10.1109/ICGCIoT.2015.7380523>.
- [18] E. Tang and Y. Fan, Performance Comparison between Five NoSQL Databases, in 2016 seventh International Conference on Cloud Computing and Big Data (CCBD), 2016; 105-109: IEEE.
- [19] D. Seybold, N. Wagner, B. Erb, and J. Domaschka, Is elasticity of scalable databases a Myth? , in 2016 IEEE International Conference on Big Data (Big Data), 2016 2827-2836: IEEE.
- [20] V. N. Gudivada, D. Rao, and V. V. Raghavan, NoSQL Systems for Big Data Management, 2014 IEEE World Congr. Serv., 2014; 190-197: IEEE. <https://doi.org/10.1109/SERVICES.2014.42>.
- [21] A. Mohan, M. Ebrahimi, S. Lu, and A. Kotov, A NoSQL Data Model for Scalable Big Data Workflow Execution, in 2016 IEEE International Congress on Big Data (BigData Congress), 2016;52-59:IEEE.
- [22] J. Han, E. Haihong, G. Le, and J. Du, Survey on NoSQL database, in *Proceedings - 2011 6th International Conference on Pervasive Computing and Applications, ICPCA 2011*, 2011; 363-366:IEEE.
- [23] K. Grolinger, W. a Higashino, A. Tiwari, and M. A. Capretz, Data management in cloud environments: NoSQL and NewSQL data stores, *J. Cloud Comput. Adv. Syst. Appl.*, 2013 2, 22. <https://doi.org/10.1186/2192-113X-2-22>.
- [24] Changqing Li and Jianhua Gu, A distinctive transformation approach of NoSQL's SQL conditions based on Espresso, in 2017 IEEE 2nd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), 2017; 61-69: IEEE.
- [25] M. T. Gonzalez-Aparicio, A. Ogunyadeka, M. Younas, J. Tuya, and R. Casado, Transaction processing in consistency-aware user's applications deployed on NoSQL databases, *Human-centric Comput. Inf. Sci.*, 2017; 7(1): seven.
- [26] j. Han, E. Haihong, G. Le, and J. Du, Survey on NoSQL database, in *Proceedings - 2011 6th International Conference on Pervasive Computing and Applications, ICPCA 2011*, 2011;363-366:IEEE.
- [27] A. B. M. Moniruzzaman and S. A. Hossain, NoSQL Database: New Era of Databases for Big data Analytics - Classification, Characteristics and Comparison, 2013; arXiv preprint arXiv: 1307.0191.
- [28] E.A. Brewer. Towards robust distributed systems. In *PODC*, 2000; seven.