**NoSQL Database Technologies**

**1.0 Introduction**

NoSQL databases have become a popular alternative to traditional relational databases due to their ability to handle large volumes of unstructured data and their ability to scale horizontally. It does not use the traditional tabular form for the storage of data and SQL queries for retrieval of data as a relational database management system does. It is called a schema-less database management system because it doesn’t follow any schema. It can handle semi-structured and unstructured data very efficiently and relational databases have problems while handling it. The name “NoSQL” was first used by Carlo Strozzi in 1998 as the name of a file-based database system that he was developing (Haseeb & Pattun, 2017). There is a common misconception that NoSQL seeks to eliminate the use of relational databases. NoSQL is best considered with the acronym "NOSQL" - Not Only SQL - which represents the approach of combining non-relational databases and relational databases. It balances performance, scalability, and schema flexibility with data integrity, consistency, and integrity to meet the demands of performance, scalability, and schema flexibility (Purdue, 2014; Penchikala, 2013).

**1.1 CLASSIFICATION OF NOSQL DATABASES**

Research of NoSQL database technologies reveals many varieties of NoSQL databases. These varieties of NoSQL technologies include document-oriented, key-value, column-family, and graph databases each tailored to serve specific functions (Madison, et al., 2015) .

Document-oriented databases, such as MongoDB and Couchbase, store data in XML (eXtensible Markup Language), JSON (JavaScript Object Notation) and BSON (Binary JSON) format and offer flexibility and scalability (Moniruzzaman & Hossain, 2013).

Key-value databases, such as Redis and Riak, are highly scalable and efficient for handling large volumes of data. Key-value stores pair alphanumeric keys with associated values in standalone hash tables. These are useful for high-speed, scalable value retrievals (Stonebraker, 2012).

Column-family databases, such as Apache Cassandra and HBase, are designed for distributed environments and provide high availability and fault tolerance. These systems provide time-stamping functions that make them especially useful for managing versioned data and have applications for predictive analytics (Moniruzzaman & Hossain, 2013).

Finally, Graph databases, such as Neo4j and OrientDB, are optimized for storing and querying graph data. Graph databases are best suited for analyzing relationships between data rather than the data itself (Moniruzzaman & Hossain, 2013).

Each type of NoSQL database has its strengths and weaknesses. For example, document-oriented databases are excellent for storing and retrieving large volumes of complex data structures. Key-value databases are ideal for storing large amounts of data that must be accessed quickly, while column-family databases are perfect for use cases that require high availability and fault tolerance. Graph databases are suitable for use cases that require complex graph traversal and querying.

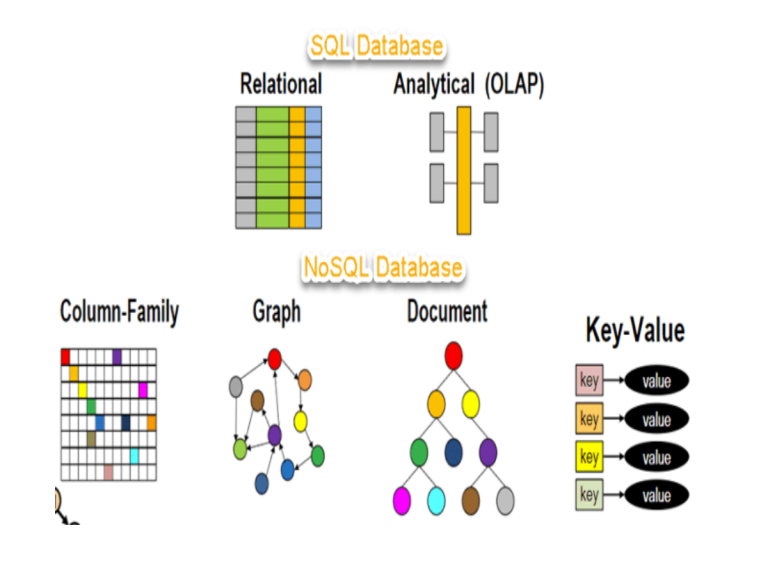


Fig. 1 SQL and NoSQL Database(guru99.com)

**1.2 Application cases**

NoSQL databases have been successfully utilized in several problem areas, including big data, real-time analytics, Fraud detection, social networking, and e-commerce. For example, Netflix uses Apache Cassandra to manage its massive content catalog, which includes over 100 million members across the globe. eBay uses MongoDB to handle its real-time e-commerce transactions, while Twitter uses Apache HBase to store its tweet data.

Also, NoSQL databases are used in content management. Forbes, which lives on viewership and ad revenue, exemplifies the use of NoSQL technology for content management. Forbes built a custom content management system based on MongoD giving them greater agility—including the ability to incorporate contributor content and analyze social sharing within clickstream data—at a lower cost. The same data store also feeds their mobile site, which now gets 50% of their total traffic.

In the area of real-time big data processing, Payments leader PayPal is a prime example of a leading digital enterprise that processes big data on the fly and leverages it in multiple ways. PayPal captures vast quantities of raw clickstream data, more than 20 TB per day, in multiple formats by processing it through Hadoop and Apache HBase NoSQL databases and storing it all in the cloud for worldwide access by business analysts and data scientists. Fraud detection, data mining, customer segmentation and delivering personalized ads to customers are just some of the differentiating capabilities that PayPal has built on NoSQL.

**1.3 Strengths and Weaknesses**

According to Okman et al. (2011), NoSQL databases are highly scalable, and reliable, have a simple data model, extremely bare query language, no mechanism for handling consistency and integrity amongst data, and almost no support for security at the database level. One of the most important advantages of NoSQL databases is that they can handle unstructured data. Unstructured data can be in the form of Word documents, emails, audio, video, or even social network data. Also, NoSQL databases tend to scale very well on commodity hardware. For companies with large amounts of data, NoSQL databases may even provide better performance (Leavitt, 2010). Unlike relational databases, NoSQL databases typically don't adhere to ACID (atomicity, consistency, isolation, durability) constraints which are present in the relational database. An example of NoSQL database performance is Facebook implementation (Cassandra), which can handle over 100 million users continuously (Okman, et al., 2011).

NoSQL databases are not without their flaws and limitations. There are several issues with these databases, such as the lack of encryption for data files, weak authentication between the client and the servers (and server members), and vulnerability to SQL injection or DOS attacks (Okman, et al., 2011). Additionally, Cassandra uses Apache's Thrift framework for client communication. An unpleasant side effect is that all passwords are sent in plain text. This allows an attacker to reveal passwords in transit using a packet sniffer. Implementations like MongoDB only support authentication when running in standalone or replica set mode. MongoDB's sharded mode does not currently support authentication (Nance, et al., 2013).

**2.0 Strategies and challenges associated with NoSQL.**

NoSQL (Not Only SQL) is a type of database that provides a non-relational, distributed, and horizontally scalable storage solution. They are ideal for use cases such as big data, real-timeanalytics, and web-scale applications that handle large volumes of unstructured or semi-structured data. NoSQL databases, however, come with several challenges, which organizations must address by adopting appropriate strategies.

**NoSQL strategies include:**

* **Choose the appropriate NoSQL database:** There are several types of NoSQL databases, including document-based, key-value, column-family, and graph databases. Each type has its strengths and weaknesses, and the choice of the right database will depend on the specific use case.
* **scalability:** NoSQL databases are made to scale horizontally, which entails expanding the cluster with additional nodes to boost the database's capacity. To ensure that the database can manage the additional workload, proper preparation is necessary**.**
* **Ensure data consistency:** Data consistency must be ensured since NoSQL databases frequently forgo data consistency in favour of scalability and performance. Nonetheless, it is crucial to keep the data correct and consistent, particularly in applications that need real-time data.

**Challenges/Issues with NoSQL include:**

* NoSQL databases do not have a standardized query language or schema, in contrast to relational databases. This makes the integration of NoSQL databases with other applications difficult.
* Low transaction support: As most NoSQL databases do not handle transactions, maintaining data consistency in intricate applications can be challenging**.**
* Replication and sharding are used by NoSQL databases to ensure data persistence. Unfortunately, this may make it difficult to guarantee data availability and consistency**.**
* Expertise: The design, deployment, and management of NoSQL databases call for specialized knowledge and abilities. To deal with NoSQL databases, organizations must spend money on training and employing seasoned employees**.**

**3.0 Problem areas where NoSQL has been successfully utilized.**

NoSQL systems are typically adopted by businesses when traditional SQL systems fall short of requirements for performance, flexibility or scalability (Stonebraker, 2010, Moniruzzaman & Hossein, 2013). NoSQL databases are mostly used as big data storage because they ensure high agility, relatively small delays during data reads, and high efficiency of data recording. These advantages have been noticed by huge IT firms, i.e `Facebook, Amazon, Twitter, and Google and they strongly support the NoSQL database technology development. Beyond that, NoSQL databases are applied in biological and medical research (Shukla, et al., 2015). Some other problem areas where NoSQL has been successfully utilized according to Raj and Deka, (2018) are;

* **Detecting Fraud:**

Globally billions are lost by companies due to financial fraud, the complexity and scale in which these fraudsters operate are evolving every day. NoSQL provides an efficient and versatile solution for this problem. These interactions can help find patterns in fraud transactions, leading to identifying fraud rings and red flags on malicious transactions. Conventional relational databases find it hard to keep up with the speed and increasing scale of data production. NoSQL databases can be elastically scaled and provide low latency.

* **E-commerce:**

E-commerce platforms deal with a variety of products and their details; the type of products and product information are changing. A relational model cannot easily manage product information. In contrast, NoSQL databases can manage information like product, order, value, wish list, product viewed and much more data easily, and can update the database scheme as more data is being collected. Relational databases are run on a single server, which on demand can be scaled vertically making it expensive on the hardware. NoSQL databases can run on relatively cheaper servers and can be scaled horizontally, meaning cheaper servers can be added for scaling. This can be very useful in the case of an e-commerce site as the amount of seasonal traffic is very high.

* **The Internet of Things. (IoT)**:

Everything from mobiles, smart appliances, devices, and sensors connected to the internet generates huge amounts of data in different formats. There is a need to logically connect these devices and process this vast amount of data. The volume, velocity, and variety of these semi-structured data generated from these devices are increasing and RDBS is struggling to keep up. NoSQL provides a unique solution for connecting and handling large data. NoSQL provides flexible data for IoT, we can easily scale these systems and still get better performance.

* **Social Network:**

Social network sites like LinkedIn, Facebook, Twitter, and YouTube were the first to be challenged by the big data problem. There was a lot of dynamic and diverse data coming in from distributed locations, growing at an unimaginable scale. NoSQL was able to address these challenges of Volume, Velocity and Variety for the social network sites. NoSQL databases were able to scale up to handle massive transactions and almost provided real-time performance. The implementation of NoSQL graph databases can help in recommending content or connections increasing user engagement on the site. These recommendations are used for targeted marketing for effective monetization from adverts. Social network sites use a combination of relational and NoSQL databases, as both have their own advantages and disadvantages.

* **Real-Time Recommendation:**

Recommendation engines play a vital role in user website engagement. For example, travel companies use real-time recommendations based on what you’ve clicked on or Amazon’s most brought together to increase the interaction between the website and customer. The RDMS is a rigid schema, in the real-world situation new data sources may need to be accommodated, the NoSQL database can evolve without much change to the data model accommodating new data types and sources. A real-time recommendation engine should be able to adapt to a new interest by the user, which is difficult to accommodate in conventional RDMS.

.

# References

Haseeb, . A. & Pattun, G., 2017. A review on NoSQL: Applications and challenges. *International Journal of Advanced Research in Computer Science,* Volume 8(1).

Leavitt, N., 2010. Will NoSQL databases live up to their promise?. Volume 43(2), pp. pp.12-14.

Madison, M., Barnhill, M., Napier, C. & Godin, J., 2015. NoSQL database technologies. *Journal of International Technology and Information Management,* Volume 24(1), p. p.1..

Moniruzzaman, A. & Hossain, S. A., 2013. Nosql database: New era of databases for big data analytics-classification, characteristics and comparison. *International Journal of Database Theory and Application ,* Volume Vol. 6, No. 4.

Nance, C., Losser, T., Iype, R. & Harmon, G., 2013. *Nosql vs rdbms-why there is room for both.* s.l., s.n.

Okman, L. et al., 2011. *Security issues in nosql databases. 10th International Conference on Trust, Security and Privacy in Computing and Communications (pp. 541-547).* s.l., s.n.

Raj, P. & Deka, G. C., 2018. *A deep dive into NoSQL databases: the use cases and applications.* s.l.:Academic Press.

Shukla, R., Pandey, P. & Kumar, V., 2015. Big data frameworks: at a glance. *Int. J. Innovations Adv. Comput. Sci. IJACS,* Volume 4(1), pp. pp.169-175.

Stonebraker, M., 2012. New opportunities for new sql. Communications of the ACM. Volume 55(11), pp. pp.10-11.

Strauch, . C., Sites, U. & Kriha, W., 2011. *NoSQL databases.* 20(24), p.79 ed. s.l.:Stuttgart Media University.

Zaki, A. K., 2014. NoSQL databases: new millennium database for big data, big users, cloud computing and its security challenges. *International Journal of Research in Engineering and Technology (IJRET),* Volume 3(15), pp. pp.403-409.

https://www.mongodb.com/scale/types-of-nosql-databases.

Gupta, A., Tyagi, S., Panwar, N., Sachdeva, S. and Saxena, U., 2017, October. NoSQL databases: Critical analysis and comparison. In *2017 International conference on computing and communication technologies for the smart nation (IC3TSN)* (pp. 293-299). IEEE.

Perdue, T. (2014, October 14). An overview of NoSQL. Retrieved from newtech.about.com:

http://newtech.about.com/od/databasemanagement/a/Nosql.htm

Penchikala, S. (2013, July 23). NoSQL database adoption trends. Retrieved from Infoq.com:

http://www.infoq.com/research/nosql-databases

Sicari, S., Rizzardi, A. and Coen-Porisini, A., 2022. Security&privacy issues and challenges in NoSQL databases. *Computer Networks*, p.108828.