Communications on Applied Electronics (CAE) – ISSN : 2394-4714
Foundation of Computer Science FCS, New York, USA
Volume 6 – No.1, October 2016 – www.caeaccess.org

SQL vs. NoSQL vs. NewSQL- A Comparative Study

Sneha Binani
Information Technology
VESIT
Mumbai, India

Ajinkya Gutti
Computer Engineering
VESIT
Mumbai, India

Shivam Upadhyay
Information Technology
VESIT
Mumbai, India

ABSTRACT
SQL Databases also known as RDBMS (Relational Database Management Systems) is the most common and traditional approach to database solutions. The data is stored in a structured way in form of tables or Relations. With advent of Big Data however, the structured approach falls short to serve the needs of Big Data systems which are primarily unstructured in nature. Increasing capacity of SQL although allows huge amount of data to be managed, it does not really count as a solution to Big Data needs, which expects fast response and quick scalability.

To solve this problem a new kind of Database system called NoSQL was introduced to provide the scalability and unstructured platform for Big Data applications. NoSQL stands for Not Only SQL. NoSQL databases consist of key-value pair, Documents, graph databases or wide – column stores which do not have a standard schema which it needs to follow. It is also horizontally Scalable as opposed to vertical scaling in RDBMS.

NoSQL provided great promises to be a perfect database system for Big Data applications; it however falls short because of some major drawbacks like NoSQL does not guarantee ACID properties (Atomicity, Consistency, Isolation and Durability) of SQL systems. It is also not compatible with earlier versions of database. This is where NewSQL comes into picture. NewSQL is a latest development in the world of database systems. NewSQL is a Relational Database with the scalability properties of NoSQL. This paper discusses each of these database systems and tries to find the ideal solution for Big Data requirements.

General Terms
Database, Software program, SQL, NoSQL, NewSQL, Big Data, ACID

Keywords
NewSQL, Big Data, Features of NewSQL, Difference between SQL, NoSQL and NewSQL, OLTP (On-Line Transaction Processing) and Big Data, ACID properties, BASE properties.

1. INTRODUCTION
Database is a collection of data which can be used alone or combined with some other data.[1] Database Management system is software that allows the computer to perform database function of storing, retrieving, adding, deleting and modifying data. [1] Database management system is a collection of interrelated data, set of programs to access the data and an environment that is both convenient and efficient to use.

2. WHERE ARE DATABASES USED?
They are used to support internal operations of organizations. They are used to take care of the transactions in banking, making reservations and displaying schedules in airlines, managing registrations and keeping record of the grades in universities and to name some more, they are used in Sales, Online retailers, Manufacturing, human resources etc.

3. SQL
3.1 Definition
SQL is a computer language used in databases for managing data in Relational Database Managing System (RDBMS). [2]SQL was originally developed in 1970 in the IBM laboratories. RDBMS uses relational model-which has relationship between tables using foreign keys, primary keys and indexes. Because of this fetching and storing of data becomes faster than the old Navigational model. SQL was originally developed by IBM. [2] SQL is a standardized language used for defining, querying and managing data. SQL can retrieve data from database and execute queries against it. SQL also works on many records in a database by inserting, updating and deleting records. It can create new databases, new tables in a database, stored procedure and views. [2] Along with this, SQL can set permissions on tables, procedures and views.

3.2 Architecture
There are two types of architecture: Physical and Logical. Physical architecture tells about how the data is actually stored in file system of an operating system. Page, extent, database files, transaction log files etc. are core components of physical architecture. Logical architecture tells about how the data is logically grouped and presented to the user. Tables, constraints, views, stored procedures, functions, triggers etc. are core components of logical architecture. [2] SQL language deals with the following issues: transaction control, integrity constraints, authorization, views and embedded SQL and dynamic SQL.

3.3 ACID
ACID stands for Atomicity, Consistency, Isolation and Durability. This property is mainly used in transaction. A transaction is a single logical operation on the data. ACID properties are important to ensure the integrity of the data. [4]Atomicity means that either all operations of the transaction are reflected in the database or none are reflected. Consistency ensures that the data is preserved in its consistent state even after the transaction. In Isolation, each transaction must be unaware of the other transactions which are executing concurrently. Durability means that even if there are system failures, the changes made to the database after a transaction will persist.

3.4 SQL in Big Data
Structured Query Language has dominated for several decades and is currently being invested in by big data companies and organization. [5]Data that is not interactive becomes useless and it is not beneficial to use them. Therefore, we use SQL which enables interaction with data and allows a broad question to be asked against a single

database design. [5]SQL allow users to apply their knowledge across systems and provides support for third-party add-ons and tools because it is standardized. [5]SQL solves problems ranging from fast write-oriented transactions to scan-intensive deep analytics. Since SQL is schema-oriented , the structure of the data should be known in advance which is difficult to obtain in big Data. Also, processing unpredictable and unstructured information is not possible for SQL. Hence, we have to switch to NoSQL so that these flaws can be fixed.

4. NoSQL

4.1 What is NoSQL

The rise of Big Data created a demand for horizontally scalable Data Management System. This led to development of different kinds of Database Management System which collectively come under NoSQL. NoSQL Databases are broadly divided into following types: Document, Graph, Native XML, Key-value, Native object, Table type, and Hybrid Databases. All RDBMS databases are based on the same model, whereas, each of the NoSQL database follows a different model. NoSQL moves away from the hefty standardized form of SQL database and enables simpler data storage solutions. Thus a NoSQL database is optimized for the specific application.

4.2 Data Models in NoSQL

The Architecture for each of the NoSQL data model varies. Common data models are key-oriented storage, graph model, or relational model.

4.2.1 Key based NoSQL Data Model

As the name suggests the data in key oriented NoSQL database is stored and accessed using keys [7]. Almost all queries are key lookup based i.e. to access data of a certain entity for ex. Student you have to access his data through his key in this case the Student ID.

The simplest form of key based model is key – value store. In this each key is mapped to a value containing any data. NoSQL has no information of the data; it just delivers the data based on the key. Some systems based on this model are Amazon DynamoDB, CouchDB, Membase.

Another type of key oriented model is key – Document model. The keys are mapped to documents that contain structured information. The documents are stored in JSON or JSON like format. Examples of systems in this category are MongoDB, CouchDB, RavenDB, FatDB [7].

4.2.2 Graph Based Data Model

Graph is one of the fundamental databases in Computer Science. Graph represents connected data or the relationship between the data. Graph system is layered and thus can be used to implement authorization and access control. Some famous graph database systems include Google knowledge graph, Facebook Open graph, HypergraphDB, Neo4J [7].

4.3 Properties of NoSQL Databases

4.3.1 No sharing Architecture

The NoSQL database is based on No Sharing Architecture in which neither memory nor the storage is shared [7]. This enables each node to operate independently. The scaling thus becomes very convenient as new nodes can be added to the system easily. Data is distributed across nodes in a non-overlapping way; this technique is known as sharding.

4.3.2 BASE Properties

BASE is to NoSQL what ACID is to SQL. BASE is the property on NoSQL databases that ensure its reliability in spite of loss of Consistency. BASE stands for Basically Available Soft state eventually consistent.

Basically Available- This states that the system guarantees availability of the data.

Soft state – The system state may change at any time, even when no input is given to the system.

Eventually Consistent- The system will eventually become consistent as its state can change when not receiving inputs. This means that sooner or later the data will be updated wherever necessary thus maintaining the consistency of the database

4.4 How NoSQL is better than SQL for Big Data applications

The data in Big Data applications varies widely. The data is collected from different sources like social media, mobile phones, etc. The data can be personal information of the user, location data, machine data, sensor generated data, etc. To handle such a data scalability and flexibility is of utmost importance.

Scaling in SQL systems means spending money on expensive hardware at a single node. This vertical scaling is not an effect and economical approach. NoSQL being horizontally scalable can be easily used to implement Big Data applications. Scalability in NoSQL is as easy as adding a server node into the system [6]. The load on the system is thus shared between the nodes.

Flexibility is inherent in NoSQL databases as it does not have to be restricted to a certain schema unlike Relational Databases.

4.5 Shortcomings of NoSQL

NoSQL is still in its infant stage. There is a long way to go for it to become richly functional and stable system.

Because of still being in the early stage there are very less advanced expertise in this field [6]. It does provide with BASE properties, however it is not as reliable as the ACID properties provided by SQL databases. ACID properties of transactions are vital in various cases such as banking firms.

ACID	BASE
ACID stands for **A**tomicity, **Consistency**, **I**solated and **D**urability.	**BASE** stand for **B**asically **A**vailable, **S**table state, **E**ventually consistent.
Focus is on Consistency and Availability	Focus is on Availability and Partition tolerance
Strong **Consistency**	Weak consistency
This is a pessimistic approach	This is an optimistic approach

Complex mechanisms	Simple and fast
Primarily used where data reliability and consistency is very important	Primarily used where data availability and speed is important

Figure 1: ACID vs. BASE

5. NewSQL

5.1 What is NewSQL

NewSQL is a technology that aims at making current relational SQL more scalable. It's an attempt to combine NoSQL and SQL. SQL provides ACID properties but isn't fast enough when it comes to concurrency. NoSQL aims at Brewers CAP theorem but doesn't necessarily provide ACID properties. NewSQL tries to provide relational DBMS that has same scalability as NoSQL for OLTP while still providing ACID properties [10].

5.2 Architecture of NewSQL

An ideal DBMS should scale elastically vertically as well as horizontally. It should allow new machines to be introduced easily in a system that's already up and running [8]. This wasn't possible to be performed efficiently with SQL. So NewSQL uses technology that's used in cloud computing and distributed applications. It implements distributed database technology. The databases are generally distributed. They follow the three tier architecture having three layers: an administrative tier, a transactional tier and a storage tier [8]. SQL provided vertical scaling but there was no provision for horizontal scaling. The NewSQL model provides horizontal scaling along with vertical scaling. The databases used are distributed while still providing ACID properties. They work efficiently in heavy load and are very robust. NewSQL also has a provision for distributed services which work with high efficiency.

5.3 Properties of NewSQL

NewSQL is a relational database. It supports ACID properties. Its schema is a combination of SQL and NoSQL. It provides horizontal scalability. It has cloud support and can also be used for OLTP. It supports SQL but query complexity is very high. NewSQL gives high performance by keeping all data in RAM. Scalability is provided by employing partitioning and replication in such a way queries generally do not have to communicate between multiple machines. They get the required information from a single host. This is why NewSQL is the best option for those who want to develop highly scalable and efficient OLTP systems.

Table 1: Comparison of SQL, NoSQL and NewSQL

Distinguishing Feature	OldSQL	NoSQL	NewSQL
Relational	Yes	No	Yes
ACID	Yes	No(Provides CAP)	Yes
SQL	Yes	No	Yes
OLTP	Not fully supported	Supported	Fully supported
Scaling	No	Yes	Yes
Query Complexity	Low	High	Very High
Distributed	No	Yes	Yes

5.4 NewSQL and Big Data

Big data is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications [11]. The amount of information is so huge it requires thousands of servers running in parallel to work with this data. It involves many challenges like capturing of data, storage of data, processing and analysis of data. NewSQL allows you to work with this Big Data more efficiently.

5.5 Application of NewSQL: Google Spanner

One application of NewSQL is Google Spanner. Spanner is a globally distributed database which is scalable. It is deployed at Google. It is based on NewSQL. It is a database that performs sharding of data that is horizontal partition of data and it is spread across many Paxos state machines [9]. Paxos state machines are used to solve consensus which is the process of agreeing upon one result when there are multiple participants. The datacenters are spread all over the world. The databases should be available globally. So for this purpose replication is used. The replication isn't completely random. It considers the geographic locality and what kind of data is required more frequently. Spanner works dynamically. It reshards and migrates data automatically for the purpose of load balancing [9]. For achieving low latency and high availability most applications would probably replicate data over three or five datacenters in one geographic region. Spanner is useful when applications want strong consistency and are distributed over a large area. Spanner performs versioning of the data and stores the time stamp which is the same as the commit time. Unrequired data can be deleted with proper policies for handling old data. Spanner is very useful for OLTP concerning Big Data. It uses SQL query language.

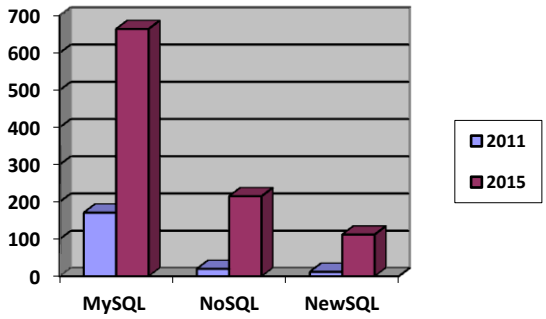

Figure 1 : [3] Increase in revenue from 2011 to 2015

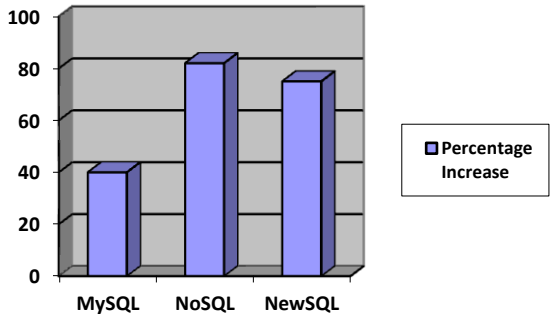

Figure 2 : [3] Percentage increase in revenue from 2011 to 2015

6. CONCLUSION

SQL provides vertical scaling with ACID properties while NoSQL is suitable for horizontal scaling providing BASE. However NoSQL does not provide ACID properties which are necessary for a reliable database. The need of modern enterprises where data is growing day by day and all they work with is Big Data especially even while working in OLTP system, NewSQL is the best choice. NewSQL is enhancement of SQL providing horizontal scaling while maintaining ACID properties. This not only allows working with Big Data by providing the ability to work concurrently, it also maintains ACID properties. NewSQL has found the sweet spot between consistency, scalability, speed and availability. While still being in its infant stage, NewSQL ticks all the right boxes to make it an ideal database for Big Data OLTP applications.

7. REFERENCES

[1] Imtiaz Rashid : “Term Paper on Database Management System”

[2] Don Chamberlin : “SQL”, IBM Almaden Research Center, San Jose, CA

[3] Matthew Aslett :"451 Research delivers market sizing estimates for NoSQL, NewSQL and MySQL ecosystem” May 22nd, 2012

[4] Shiwei Yu : "ACID Properties in Distributed Databases" , 20096

[5] Ryan Betts : “SQL Time-Tested and still flourishing” , VoltDB, Bedford.

[6] Jenny Richards, Advantages and Disadvantages of NoSQL databases – what you should know, Hadoop360, September 24, 2015, http://www.hadoop360.com/blog/advantages-and-disadvantages-of-nosql-databases-what-you-should-k

[7] Venkat Gudivada, Dhana Rao and Vijay Raghavan: "NoSQL Systems for Big Data Management”, June 2014, Conference: 2014 IEEE World Congress on Services, At Anchorage, Alaska.

[8] A B M Moniruzzaman: "NewSQL: Towards Next-Generation Scalable RDBMS for Online Transaction Processing (OLTP) for Big Data Management", Nov 2014.

[9] Hoff, Todd : "Google Spanner's Most Surprising Revelation: NoSQL is Out and NewSQL is In". Retrieved 2012-10-07.

[10] Stonebraker, Michael "NewSQL: An Alternative to NoSQL and Old SQL for New OLTP Apps". Communications of the ACM Blog. , 2012

[11] Vinay Jain "Rise of NewSQL". International Journal for Research in Emerging Science and Technology.