International Journal of Trend in Scientific Research and Development (IJTSRD)
International Open Access Journal
ISSN No: 2456 - 6470 | www.ijtsrd.com | Volume - 2 | Issue – 3

A Comparison between Relational Databases and NoSQL Databases

Behjat U Nisa
M.Tech Scholar, Department of CSE, Swami Devi Dyal Group of Professional Institutes, Haryana, India

ABSTRACT

Databases are used for storing and managing large amounts of data. Relational model is useful when it comes to reliability but when it comes to the modern applications dealing with large amounts of data and the data is unstructured; non-relational models are usable. NoSQL databases are used to store large amounts of data. NoSQL databases are non-relational, distributed, open source and are horizontally scalable. This paper provides the comparison of the relational model with NoSQL

Keywords: *Relational Model, NoSQL, ACID, BASE, CAP*

I. Importance

Database is a collection of logically related data. Database Management System (DBMS) is a software that is used to store and manage data present in a database. It uses data models to show the various ways in which the data can be managed. Several data models are available like the hierarchical model, the network model, relational model, non-relational model etc. Relational model has dominated the industry for over four decades now. In this model data is organized into tables i.e. data is stored in rows and columns format. This data needs to be reliable. It thus follows the ACID properties (Atomicity, Consistency, Isolation and Durability) to ensure the reliability of data.

- Atomicity: means that either all occurs or nothing occurs. It prevents partial updates to a database. Thus, if a transaction is left incomplete then the whole transaction fails. This ensures that the data is consistent.
- Consistency: ensures that the various operations in a transaction are performed correctly, accurately. It also ensures that no constrains on the database are violated. It ensures that the effects of the transactions done in the past should be seen in future transactions.
- Isolation: guarantees that various transactions that occur simultaneously should not affect the execution of one another. It ensures that the data available in the database after each of these transaction is valid and reliable.
- Durability: makes sure that the transactions which have been committed they will be stored permanently.

In the non-relational model data is not organized in tables. These databases store data in various formats like XML format or Key Value Pairs. These databases are called the NoSQL databases. NoSQL databases are more easily scalable than the relational databases and hence can handle large amounts of data. The NoSQL data model does not ensure the ACID properties but in place of that, it guarantees BASE properties (Basically Available, Soft state, Eventual consistency).It is in accordance with the CAP (Consistency, Availability, Partition tolerance) theorem.

BASE is the reverse of ACID

- Basically Available: System ensures availability of data i.e. data is made available even if multiple failures are present. This is achieved by using a highly distributed approach.
- Soft State: the state of the system may change with time even without input, the idea of consistency is abandoned completely.

- Eventual Consistency: indicates that the system will be consistent over time (permanent availability).

CAP Theorem: CAP stands for Consistency, Availability and Partition tolerance. This theorem states that a system can have only two out of three properties i.e. consistency, availability, and partition-tolerance.

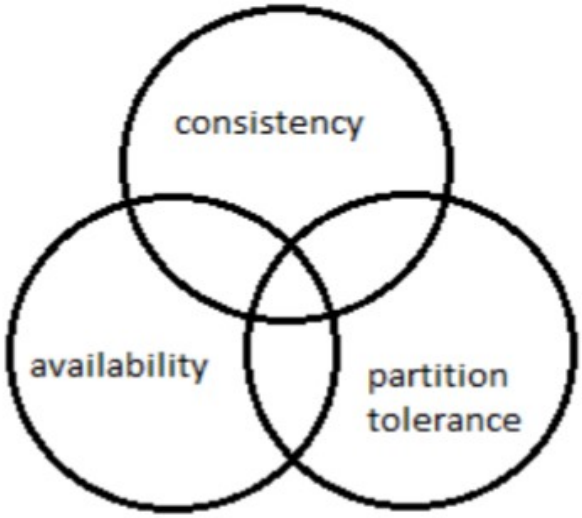

Fig 1. CAP Theorem.

The relational database, NoSQL and the conclusion are now elaborated in the sections that follow.

II. Relational Database.

RDBMS is an advanced version of DBMS. It is based on the Relational model introduced by E.F.Codd. This model is an attempt the database structure. Relational database is table based i.e. it consists of data arranged in the form of rows and columns. Each row contains a value of data for the corresponding column. Constraints are also applied to this data. The tables are related to one another and because of this these tables are also called relations. Many popular databases currently in use are based on the relational model. Some of the most popular RDBMS are Microsoft SQL Server, DB2, Oracle, MYSQL etc. The relational model was well-suited to client-server programming. Relational databases use SQL (Structured Query Language) to access the data and make changes to it. The following table shows the details of a student. This example shows how data in a relational database is stored.

ID	Name	Age	Semester
1001	Jane	20	First
1002	Thomas	25	Third
1003	Catherine	22	First
1004	Joe	23	Third

The table format is simple and easy to understand and use. The data can be easily inserted, updated and deleted. Data can be accessed simultaneously by multiple users and still data consistency is maintained.

A. Problems with Relational Databases

1. Relational databases do not support high scalability. In other words we can say that these databases are not capable of handling exponential growth of data.
2. Setting up and maintaining relational databases is expensive. Also some features of relational databases are not used and hence result in higher cost as well as complexity.
3. SQL can be highly complex when it comes to working with unstructured data.
4. Since data is stored in the form of tables, data cannot be easily encapsulated.
5. In case of complex relational database systems it becomes difficult to share information from one database to another.

Thus, in order to solve the problems encountered non-relational databases are used.

III. NoSQL

NoSQL is a non-relational database. It means 'Not Only SQL'. This model gained popularity after the introduction of database as a service. NoSQL databases are horizontally scalable databases. These databases differ from relational databases in a number of ways. No SQL databases can handle unstructured data such as documents, e-mail, multimedia and social media unlike the relational databases. These databases do not use tables as storage structures. They do not ensure the ACID properties instead they guarantee the BASE properties. SQL which is used as a query language in relational databases is not used in NoSQL. This database can retrieve unstructured, semi-structured as well as structured data. They provide high availability at the cost of consistency.

Fig 2. NoSQL Symbolic Representation.

Some of the features of NoSQL are as follows:

1. Ability to scale horizontally.
2. Replicate and distribute data over different servers.
3. Weaker concurrency model than ACID
4. Flexible schema
5. Efficient use of distributed indexes and RAM
6. Adding new attributes to data records

Types of NOSQL Databases:

NoSQL databases can be classified into five types:

1. Key-value store databases
2. Column oriented databases
3. Document store databases
4. Graph databases
5. Object oriented databases

Each database is individually good at dealing with size and complexities.

A. Key – Value Store Databases

Key-value Store databases: The key-value data stores are simple but efficient and powerful. They allow data to be stored in a schema-less manner. The data as the name suggests consists of two parts a key which is a unique identifier to a particular data entry and a value that corresponds to the key. The data can be any primitive of a programming language or it can be an object. It has a simple API. It favors high scalability over consistency. These data stores are like hash tables where keys are used as indexes thus the query speed is higher than a relational database. The stores can hold structured as well as unstructured data. These data stores can be used to save a user's session or user's shopping cart. Some of key value stores available in market are Berkeley DB, Tokyo Tyrant, and Amazon Dynamo.

B. Column Oriented Databases

Column oriented databases as the name suggests store the data in the form of columns. In column stores, each key is associated with one or more attributes. Column data can be written efficiently and replace old data without altering any other columns for the rows. It offers high scalability in data storage. Column oriented databases are suitable for data mining and analytic applications. Some of the column store databases used are BigTable, Cassandra and HBase.

C. Document Store Databases

Document store databases are also known as 'Document Oriented Databases'. The data is stored and organized in the form of documents. These documents are in some ways similar to the records in relational databases but more flexible and schema-less. The documents are in standard formats like XML, PDF or JSON. The documents are represented by a unique key which is a simple string or a URI or path. They offer great performance. Each document may store similar as well as dissimilar data. These data stores are slightly more complex than the key value data stores. They can be used for content management system and blog software. Examples of document databases include MongoDB, Apache CouchDB.

D. Graph Databases

Graph databases store data in the form of graphs in a schema-less manner. The graphs consist of nodes, edges and properties. Nodes represent entities like people and objects, edges represent the relationships between these nodes and properties contain any information pertinent to the nodes. It provides an efficient way to store semi structured data. These databases are faster than the relational databases as queries are expressed as traversals. Graph databases support ACID properties. They can be used in social networking applications, network and cloud management, bioinformatics etc. An example of graph database is the Neo4j.

E. Object Oriented Databases

Object oriented database also called OODBMS is a database in which data is stored in the form of objects. It is considered to be a combination of database principles and object oriented programming. All the features of object oriented programming i.e. encapsulation, abstraction, polymorphism and

inheritance are provided by the object oriented data store. Objects have an identifier which uniquely represent it. Object oriented databases provide faster access to data as objects are accessed using pointers. These databases are used in scientific research, computer aided design etc. Example of object oriented database is Db4o.

IV. Relational Database and NoSQL

A. Advantages of NOSQL over Relational Databases

1. Easily scalable.
2. Database administrators are not required
3. Structured , semi structured and unstructured data can be stored
4. Some databases are programmed to handle hardware failures
5. Faster, more efficient and flexible.

B. Disadvantages of NOSQL over Relational Databases

1. There is no standard query language
2. Maintenance is difficult
3. Some databases do not follow the ACID properties

V. Conclusion

A general comparison has been made between relational databases and NoSQL databases. Each NoSQL data store addresses the various problems in relational databases. High availability, high scalability and fault tolerance are provided by them. The different data stores use different techniques to achieve this goal and seem to suit well for their requirements. The limitations of relational databases and the advantages of NoSQL database have been discussed. The users can select various data models as per their needs. There is some trade-off between speed and reliability. NoSQL is flexible compared to relational databases.

References:

1) Codd, Edgar F.: A Relational Model of Data for Large Shared Data Banks. In: Communications of the ACM 13 (1970), June, No. 6, p. 377–387

2) Uma Bhat, Shraddha Jadhav, "Moving Towards Non-Relational Databases "©2010 International Journal of Computer Applications (0975 – 8887) Volume 1– No. 13

3) Leavitt, N.,"Will NoSQL Databases Live Up to Their Promise?" Computer, vol.43, no.2, pp.12-14, Feb. 2010doi: 10.1109/MC.2010.58

4) MongoDB, http://www.mongodb.org/about/introduction/

5) Clarence J M Tauro, Aravindh S, Shreeharsha A. B, "Comparative Study of the New Generation, Agile, Scalable, High Performance NOSQL Databases", International Journal of Computer Applications (0975 – 888) Volume 48– No.20, June 2012 doi:10.5120/7461- 0336

6) http://en.wikipedia.org/wiki/NoSQL

7) Vatika Sharma, Menu Dave, "SQL and NoSQL Databases", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 8, August 2012

8) Dr.K.Chitra, B.JeevaRani, "Study on Basically Available, Scalable and Eventually Consistent NOSQL Databases", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 7, July 2013

9) Pramod J. Sadalage and Martin Fowler, "NoSQL Distilled"

10) Riak, http://basho.com/technology/why-use-riak/

11) Jing Han; Haihong, E.; Guan Le; Jian Du; , "Survey on NoSQL database," Pervasive Computing and Applications (ICPCA), 2011 6th International Conference on , vol., no., pp.363-366, 26-28 Oct. 2011 doi: 10.1109/ ICPCA.2011.6106531

12) Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C. Hsieh, Deborah A. Wallach Mike Burrows, Tushar Chandra, Andrew Fikes, Robert E. Gruber," Bigtable: A Distributed Storage System for Structured Data", Google Inc

13) Apache CouchDB, http://wiki.apache.org/couchdb/

14) Amazon DynamoDB, http://aws.amazon.com/dynamodb/

15) Neo4j, http://www.neo4j.org/learn

16) Apache Cassandra, http://wiki.apache.org/cassandra/

17) Rabi Prasad Padhy et al., International Journal of Advanced Engineering Sciences and Technologies, Vol No. 11, Issue No. 1, 015 - 030