A Review on NoSQL Databases

Miral Borad[1], Tejash Chauhan[2], Harsh Mehta[3]

[1]Computer department, Marwadi Education Foundation's Group of Institutions Rajkot, India

[2, 3]Professor, Information Technology department, Marwadi Education Foundation's Group of Institutions Rajkot, India

Abstract: ***As utilization of web is expanding step by step and consequently with this measure of information is likewise expanding. Relational database management system are confronting issues while taking care of huge measure of information because of low scalability, continuous information and unstructured information as information on web isn't appropriately organized, it can be semi-structure or unstructured. In this way, with a specific end goal to take care of the issues looked by Relational Database Management System, there is an up and coming classification of Database Management Systems that is "NoSQL". NoSQL databases normally known as "non-SQL" or "Not just SQL" databases. It isn't substitution of Relational Databases however it is another option to it. In this paper, we will examine about NoSQL: where they originated from, why NoSQL, nature of its sorts and correlation with social databases.***

Keywords: NoSQL, CAP theorem, key-value, document-oriented, column oriented, graph

1. Introduction

Relational database management system and Non-relational database management system or NoSQL differs from each other in many ways. It do not use traditional tabular form for storage of data and SQL queries for retrieval of data as relational database management system does traditionally [1]. It is called schema less database management system because it don't follow any schema. It can handle semi-structured and unstructured data very efficiently and relational databases have problem while handling it. The name "NoSQL" was first used by Carlo Strozzi in 1998 as name of file based database system that he was developing [2].

With the continuous development of the Internet and cloud computing, various types of applications have emerged, which made database technology more demands, mainly in the following aspects [3][4]:

High concurrent of reading and writing with low Latency:
Database were demand to meet the needs of high concurrent of reading and writing with low latency, at the same time, in order to greatly enhance customer satisfaction, database were demand to help applications reacting quickly enough [8].

Efficient big data storage and access requirements:
Extensive applications, for example, SNS and web search tools, require database to meet the productive information stockpiling and can react to the necessities of a great many activity.

High scalability and high availability:
With the expanding number of simultaneous requests and information, the database should have the capacity to help simple development and redesigns, and guarantee fast continuous administration.

Lower management and operational costs:
With the increment in information, database costs, including equipment costs, programming costs and working expenses, have expanded. In this manner, require bring down expenses to store enormous information. Although relational databases have involved a high position in the information storage, however when looking above necessities, it has some inalienable impediments.

Slow reading and writing:
A relational database itself has a certain logic complexity, with the data size increases, it is prone tobring about deadlocks and other concurrency issues, this has led to the rapid decline in the efficiency of reading and writing.

Limited capacity:
Existing relational database system can't carry huge information in web crawler, SNS or Big System.

Expansion difficult:
Multi-table correlation mechanism which exists inrelational database, turned into the main consideration of database scalability.

2. NoSQL Data Model

Example of traditional database model is mainly relational database, which clearly support ACID properties. In NoSQL database the current data model is following:

2.1 Key-value Stores

A data model based on keys-values which is easy to implement, but inefficient in updating and querying the part of a value [7]. Most popular key-value databases are Raik, Redis, Tokyo cabinet, Flare, etc.

2.2 Document-oriented databases

Semi-structured documents which are stored in JSON format. They support efficient querying and manage the nested values with associated keys [7]. Example of document-oriented database is MongoDB, CouchDB, OrientDB, Terrastore.

2.3 Column family databases

An efficient data model to store and process large amounts of distributed data over multiple machines [7]. Hbase, Cassandra, Google's BigTable are the example of this type of database.

2.4 Graph databases

Enable the scalability across multiple machines and allow data-model specific queries [7].Neo4J is the most popular database in graph database.

3. Nosql Database classification according to CAP theorem

As use of internet is increasing so load of traffic to server increased too for example Google, Facebook etc. Every day on Facebook, 2.4 billion content items are shared among friends [5]. On the off chance that information is expanding then we have to scale it. In the first place approach to scale is purchase a major server however there is issue with it: Cost and breaking point upto which it can be extend. Second way is when information turn out to be extensive at that point attempt to cluster it. Clustering implies when an arrangement of loosely or tightly associated nodes cooperate with the goal that they can be seen as single node. Difficulties with clustering is that while getting to information in the event that one cluster is refreshed and second one isn't then this is the issue of consistency. Therefore, the NoSQL databases follows the CAP theorem. CAP theorem was proposed by Eric Brewer in 2000 and CAP stands for [1]:

Consistency: It means if we try to access data or try to fetch data then same data should be displayed or we can say if one value written/updated to one node then another node should be updated automatically.

Availability: It means if there is any kind of failure in system then value can still be retrieved from system but value may not be consistent.

Partition Tolerance: It means if partitioning is done then there will be no impact on data that is data should be able to retrieve and write to multiple places. This feature is not available in relational databases [6].

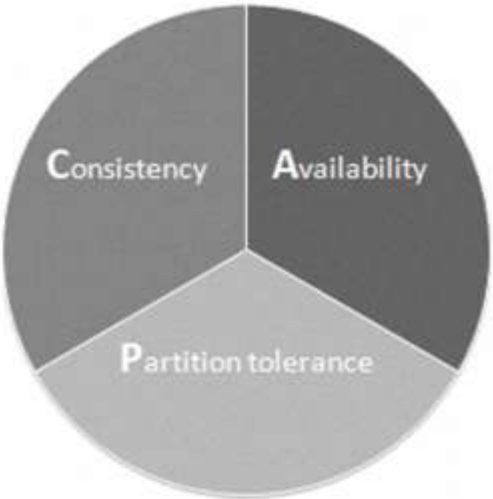

Figure 1: CAP Theorem

According to CAP theorem and different concerns of NoSQL database, a preliminary classification of NOSQL databases is as follows [9]:

- **Consistency and partition tolerance(CP):**

Such a database system stores data in the distributed nodes, but also ensure the consistency of these data, but support not good enough for the availability [8]. The main CP system example are MongoDB and Hbase

- **Availability and partition tolerance (AP):**

Such systems ensure availability and partition tolerance primarily by achieving consistency [7]. AP system example are CouchDB, DynamoDB

- **Consistency and availability(CA):**

Part of the database is not concerned about the partition tolerance, and mainly use of Replication approach to ensure data consistency and availability [7]. Systems concern the CA is relational database system.

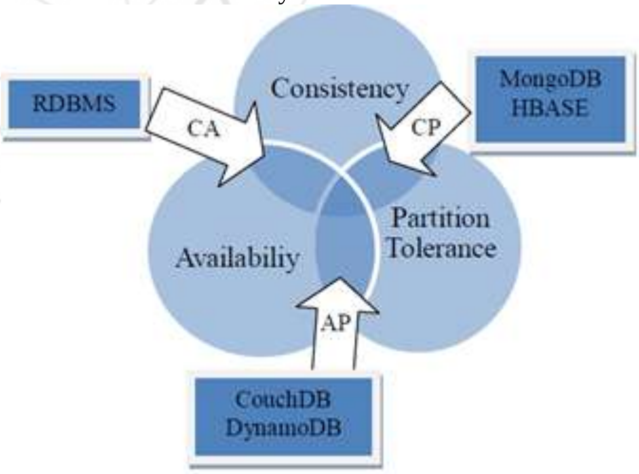

Figure 2: CAP theorem with example

4. Conclusion

In this paper, we have examined about NoSQL databases, why we require them, its sorts. Subsequent to checking on this we come to realize that NoSQL databases have many points of interest over relational databases that is the reason it is at present embraced by Facebook, Google, Amazon and so forth. Because of their substantial rate of reception they are ending up extremely mainstream among rest of databases. In any case, this is likewise genuine that picking right database additionally relies on the circumstance, we can't state what one is better. As per circumstance, we can likewise consolidate these databases in a single application for better execution.

References

[1] (2011), C. Strauch, "NoSQL Databases" Available: http://www.christofstrauch.de/nosqldbs.pdf,.
[2] History of NoSQL Available: http://www.w3resource.com/mongodb/nosql.php
[3] Kai Fan, "Suvey on Nosql", Programmer, 2010(6): pp.76-78
[4] Jing Han, Meina Song, and Junde Song. "A Novel Solution of Distributed Memory NoSQL Database for Cloud Computing", In ICIS 2011, 10th IEEE/ACIS International Conference on Computer and Information Science, 2011.
[5] (29 Sep 2013), Facebook Newsroom: A New data center for Iowa, Available: http://newsroom.fb.com/News/606/A-New-Data-Centre-for-Iowa
[6] Neal Leavitt " Will NoSQL Databases Live Up to Their Promise?" IEEE Computer Society 0018-9162/10 © 2010 IEEE.
[7] A. Bansel, H. González-Vélez and A. E. Chis, "Cloud-Based NoSQL Data Migration," 2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP), Heraklion, 2016, pp. 224-231.
[8] Jing Han, Haihong E, Guan Le and Jian Du, "Survey on NoSQL database," 2011 6th International Conference on Pervasive Computing and Applications, Port Elizabeth, 2011, pp. 363-366.
[9] Nathan Hurst, "Visual Guide to NoSQL Systems.",http://blog.nahurst.comlvisual-guide-to-NoSQL-systems/