Database for Unstructured,Semistructured data-NoSQL

Prof. Kirti
Govt College Safidon, Safidon(Jind)

Prof. Pardeep Maan
Govt College Safidon, Safidon(Jind)

Abstract - **Since the rise of the web, the volume of data stored about users, objects, products and events has exploded. Data is also accessed more frequently, and is processed more intensively for example, social networks create hundreds of millions of customized, real-time activity feeds for users based on their connections' activities. For this purpose we need a database that can handle all types of data (Structured, Semi Structured, Unstructured data) .The purpose of this paper is to explore NoSQL and its categories of Data Model to make use of it in storing and fetching data.**

Keywords – NoSQL; Scalability; BASE; ACID; Consistency.

I. INTRODUCTION

A form of database management system that is non relational. NoSQL database management systems are useful when working with a huge quantity of data when the data's nature does not require a relational model. The data can be structured, but NoSQL is used when what really matters is the ability to store and retrieve great quantities of data, not the relationships between the elements.
NoSQL databases are those databases that are open source, distributed in nature as well as it is having high performance in a linear way that is horizontally scalable. Non-relational database does not organize its data in related tables as was in the RDBMS . NoSQL databases are open source; therefore, everyone can look into its code freely, update it according to his needs and compile it. Distributed means data is spread to different machines and is managed by different machines so here it uses the concept of data replication so that can prevent data . NoSQL data stores vary widely in their offerings and have some distinct features on its own. Primary advantage of NoSQL Database is that, unlike relational databases they handle unstructured data such as documents, e-mail, multimedia and social media efficiently. NoSQL provides high performance with high availability, and offers rich query language and easy scalability.

Fig 1. Representation Of NoSQL

It states the querying to database without any interaction or interface of SQL language whwn we use NoSQL. NoSQL database technologies have excellent integrated caching capabilities, keeping frequently-used data in system memory as much as possible. This removes the need for a separate caching layer that must be maintained.

II. ACID TO BASE PROPERTY

Atomicity : All of the operations in the transaction will complete, or none will.
Consistency : Transactions never observe or result in inconsistent data.
Isolation : The transaction will behave as if it is the only operation being performed
Durability : Upon completion of the transaction, the operation will not be reversed.
Relational databases were introduced into the 1970s to allow applications to store data through a standard data modeling and query language (Structured Query Language, or SQL). At the time, storage was expensive and data schemas were fairly simple and straightforward. Since the rise of the web, the volume of data stored about users, objects, products and events has exploded. Data is also accessed more frequently, and is processed more intensively – for example, social networks create hundreds of millions of customized, real-time activity feeds for users based on their connections' activities. In order to deal with these requirements, these companies maintain clusters with thousands of commodity hardware machines. Due to their normalized data model and their full ACID[1][2] support, relational databases are not suitable in this domain, because joins and locks influence performance in distributed systems negatively.

ISSN: 2278 – 1323

The **BASE**[4] acronym is used to describe the properties of certain databases, usually NoSQL databases. It's often referred to as the opposite of ACID.

- Basically Available- The database is basically available such that if some part of the database becomes unavailable, other parts of the database continue to function as expected.
- Soft State-If the consistency is not ensured after each transaction, the system as a whole must recognize that the data is not always across instances. This means the system must deal with a Soft State.
- Eventually Consistent - a BASE system simply guarantees consistency after a reasonable time span. In other words, consistency is ensured, but not necessarily immediately.

ACID	BASE
Strong consistent	Weak consistent
Focus on "commit"	Availability first
Schema defined	Dynamic schema
Difficult evolution	Easier evolution
Used with relational database	Used with NoSQL

III. CATEGORIES OF NOSQL DATABASES

A. Key-Value Store

Key is a unique identifier to a particular data entry. Key should not be repeated if one used that it is not duplicate in nature. Value is a kind of data that is pointed by a key.
Key value databases seem to be as hash tables or look up tables. In this type of database, there is only one way to query that is with the help of key (unique) and all the keys may name in any data objects and are arranged in an alphabetical order.

Key–value stores allow the application to store its data in a schema-less way. Data is stored in key-value pairs .For higher availability of data stores data objects are replicated.

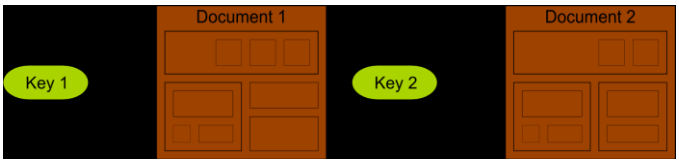
COLLEGE DATABASE

KEY	VALUE
1	ID:1 NAME:PARDEEP DESIGNTAION:PROFESSOR
2	ID:2 NAME:KIRTI DESIGNTAION:PROFESSOR
3	ID:3 NAME:AAINA DESIGNTAION:PROFESSOR
4	ID:4 NAME:RAHUL DESIGNTAION:PROFESSOR
5	ID:5 NAME:KAVITA DESIGNTAION:PROFESSOR

B. Document Store Databases

Data stored as whole documents. Every document[7] contains a special key "ID", which is also unique within a collection of documents and therefore identifies a document explicitly.
JSON(Java script object notation) & XML are popular formats. Documents[2] in the database are addressed using a unique *key* that represents that document.

ISSN: 2278 – 1323



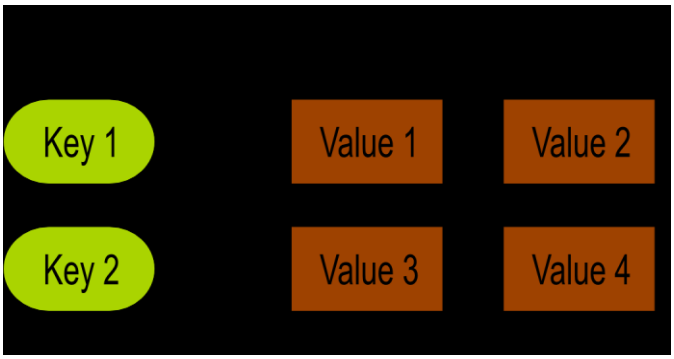
Storing new documents containing any kind of attributes can as easily be done as adding new attributes to existing documents at runtime. The most prominent document stores are CouchDB , MongoDB, RavenDB. CouchDB and RavenDB do in fact store their data in JSON. Maps well to an object oriented programming model.

{Name:value,mobile:"value",spouse:{name:value,age:value, hobby:value}}

SubDocument

C. *Columnar Databases*

Column Family[5] Stores are also known as column oriented stores, extensible record stores and wide columnar stores. In column stores, each key is associated with one or more attributes (columns). A Column store stores its data in such a manner that it can be aggregated rapidly with less I/O activity. It offers high scalability in data storage.



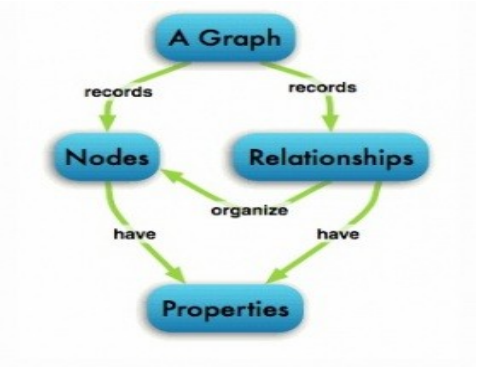
D. *Graph databases*

Graph databases[6] are based on the graph theory. In general, we see that graph usually consists of nodes, properties and edges. NoSQL Graph database consists of:

(1)Nodes represent entities

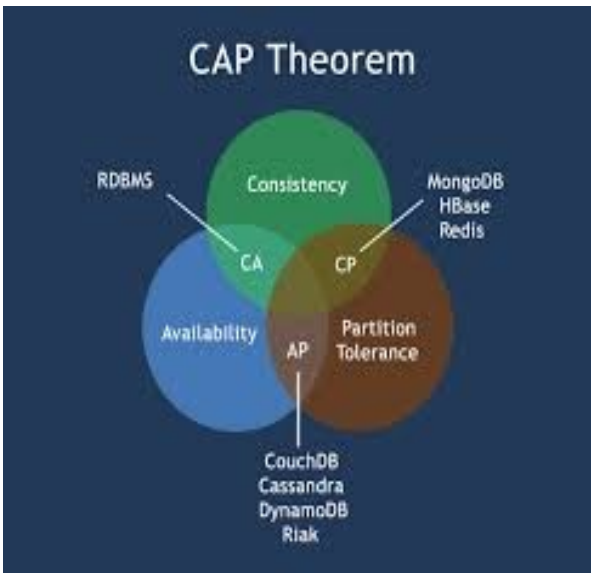
(2)Properties represent attributes

(3)Edges represent relationships



Graph databases are having high performance in context to their deep traversals. These are used for shortest path calculations. Example: FlockDB developed by twitter

IV. CAP THEOREM



ISSN: 2278 – 1323

- **Consistency** - All the servers in the system will have the same data so anyone using the system will get the latest data regardless of any updates happening.
- **Availability** - All of the servers will always return data they have (even if it's not the latest data or consistent across the system).
- **Partition Tolerance** - The system continues to operate as a whole even if individual servers fail or can't be reached.

V. NoSQL Properties

- All NoSQL doesnot make use of ACID properties
- NoSQL stores large volume of data.
- It is having more flexible structure.
- Dynamic Schemas
- It's high performance with high availability, and offers rich query language and easy scalability.
- Provides Auto sharding meaning that they natively and automatically spread data across an arbitrary number of servers.
- NoSQL databases also support automatic replication, meaning that you get high availability and disaster recovery without involving separate applications to manage these tasks.

VI. Conclusion

This paper focused on NoSQL Data models ,their uses and the theorem(CAP) ,Properties used by NoSQL(BASE). It also describes the requirement of NoSQL even if we have relational Database. Later, on the basis of the CAP theorem we described different types of NoSQL databases that are Key-Value databases, Document Store Databases, Columnar based databases and Graph databases with the help of an examples.

The reason why NoSQL has been so popular the last few years is mainly because, when a relational database grows out of one server, it is no longer that easy to use. In other words, they don't scale out very well in a distributed system. All of the big sites that we mentioned Google, Yahoo, Facebook and Amazon have lots of data and store the data in distributed systems for several reasons. It could be that the data doesn't fit on one server, or there are requirements for high availability.

References

[1] Manoj V "Comparative study of nosql document,column store databases and evaluation of cassandra", International Journal of Database Management Systems(IJDMS) Vol.6, No.4, August 2014.
[2] Ameya Nayak, Anil Poriya Dept. of Computer Engineering Thakur College of Engineering and Technology University of Mumbai " Type of NOSQL Databases and its Comparison with Relational Databases" International Journal of Applied Information Systems (IJAIS) – ISSN : 2249-0868 Foundation of Computer Science FCS, New York, USA Volume 5– No.4, March 2013
[3] Vatika Sharma, Meenu Dave "SQL and NoSQL Databases" International Journal of Advanced Research in Computer Science and Software Engineering Volume 2, Issue 8, August 2012.
[4] DAMA - Philadelphia / Delaware Valley, the "Role of Data Architecture in NOSQL", Wednesday January 11th, 2012
[5] F. Chang, *et al.,* "Bigtable: A Distributed Storage System for StructuredData," *ACMTransactions on Computer Systems,* vol. 26, pp. 1-26, 2008
[6] "GraphDB." Internet: http.z/www.sones.com
[7] 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
[8] http://www.slideshare.net/vanuganti/sql-nosql-bigdata-in-data-architecture
[9] http://www.slideshare.net/RTigger/sql-vs-no-sql
[10] http://www.nosqldatabases.com
[11] http://en.wikipedia.org/wiki/NoSQL
[12]http://www.oracle.com/technetwork/products/nosqldb/overview/index.html

Kirti received B.SC. in Computer Science in 2005 and M.Sc. in 2007 .After M.Sc. she has received MBA degree from MDU University in 2009. Presently she is professor in Govt. College , Safidon, Jind with two years experience as a lecturer in I.B. (P.G.) College, Panipat. She has qualified UGC-Net in 2012, UGC-JRF in 2014 ,HTET. Prof.Kirti has presented technical papers in ICET Conference and published around 5 research papers in International journals.

Pardeep Maan received B.SC. in Computer Science in 2006 and MCA from N.C. College of Engineering in 2009 . Presently working as Professor in Govt. College , Safidon, Jind. He has qualified UGC-Net in 2012, UGC-JRF in 2014, HTET. Prof.Maan has presented technical papers in ICET Conference, National Seminar Mobapp and published around 10 research papers in International journals.

ISSN: 2278 – 1323