www.ijecs.in
International Journal Of Engineering And Computer Science ISSN: 2319-7242
Volume 5 Issue 1 January 2016, Page No. 15598-15604

Emerging Technologies For Big Data Processing: NOSQL And NEWSQL Data Stores

Deepika Aggarwal[1], Roopam[2], Sonika[3]

Department of Computer Science and IT, DAV College Chandigarh,
Panjab University, India
deepika.agg90@gmail.com, sroopam22@gmail.com, sonika.chd4@gmail.com

Abstract: ***In this incessant science and technological era, where advances in web technology and the production of mobile devices and sensors connected to the Internet are resulting to voluminous amount of structured, semi-structured and unstructured data, called Big Data, the demand for technologies with extensive processing and storage requirements is rising to persuasively process such data i.e. Big Data. Traditional relational databases are facing challenges in meeting the performance and scale requirements of Big data. To meet these requirements enterprises are adopting diversified technologies like NoSQL and NewSQL which are emerging as alternative to relational database technologies for the various interrelated megatrends like Big Data and Cloud Computing. This paper discusses the prominent features of NoSQL and NewSQL data stores in the context of cloud computing.***

Keywords: Big Data, Cloud Computing, NoSQL, NewSQL, ACID, BASE, CAP

1. Introduction

In recent years with the expansion of cloud computing, problems of data-intensive services that use internet and require big data have come to forefront. Big data is a term for any collection of datasets so large and complex that it becomes difficult to work with the relational database management systems. Even though RDBMSs have provided database users with the best mix of simplicity, robustness, flexibility, performance, scalability and compatibility but these solutions have been encountering many challenges in meeting the performance and scaling requirements of the "Big Data" reality. To face these challenges, a number of specialised solutions have emerged in the last few years in an attempt to address the mentioned concerns. The "NoSQL" and "NewSql" data stores present themselves as data processing alternatives to RDBMSs that can handle the huge volume of data and provide the required scalability. This paper is organized as follows: "Big Data" section describes what the big data is and the requirements to process such data, "Cloud Computing" section describes the role of cloud computing as a computational paradigm in the processing of Big Data, "RDBMS" section describes the features of relational database management systems and the problems faced by relational databases to process and store Big data, "NoSQL and New Sql" section describes features of these emerging technologies; how these technologies like NoSQL and NewSQL are emerging as solutions to the problems faced by relational database management systems. The "Classification of NoSQL Databases" section presents the various types of NoSQL databases and table of comparison between NoSQL databases based on various parameters and the "Conclusion and Future Work" section concludes the paper by describing the future work on discussed big data processing technologies.

2. Big Data

Big data is a largest buzz phrases in domain of IT, new technologies of personal communication driving the big data new trend and internet population grew day by day but it never reaches by 100%. The need of big data generated from the large companies like facebook, yahoo, Google, YouTube etc. for the purpose of analysis of enormous amount of data which is in unstructured form or even in structured form. Google contains the large amount of information. So there is a need of processing the complex and massive datasets. Fig.1 shows how these datasets are different from structured data (which is stored in relational database systems) in terms of five parameters –variety, volume, value, veracity and velocity (5V's) [1].

Figure 1: Parameters of Big Data

The features of Big Data are:

- It is huge in size.
- The data keep on changing from time to time.

- Its data sources are from different phases.
- It is free from the influence, guidance, or control of anyone.
- It is too much complex in nature, thus hard to handle [2].

To provide strong storage, computation and distributed capability in support of Big Data processing, Cloud computing has emerged as a computational paradigm.

3. Cloud Computing

Cloud Computing is a model for enabling convenient, universal and on-demand network access to shared pool of configurable computing resources (network, server, storage, application and services) that can be rapidly provisioned and released with minimum management effort or service provider interaction. The idea of cloud computing is that every type of computation can be delivered to public via internet. Any stuff can be shared across any device by users via cloud computing without any problem. Network bandwidth, software, processing power and storage are represented as the computing resources to users as the publicly accessible utility services [3]. Fig. 2 shows the different service models of the cloud computing: Software as a service (Saas), Platform as a service (Paas) and Infrastucture as a service (Iaas), which are used to deliver different types of services to the Cloud service customer (CSC) as shown in Fig. 3.

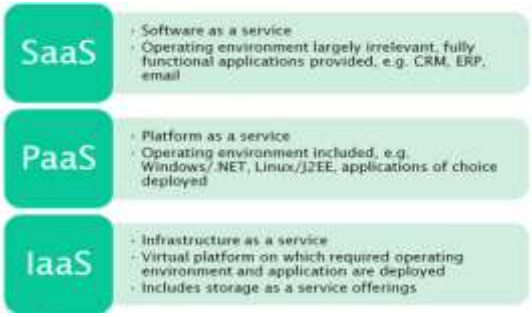

Figure 2: Cloud Computing Models

Figure 3: Cloud Computing Services delivered to CSC

Overall, a cloud computing model aims to provide benefits in terms of lesser up-front investment, lower operating costs, higher scalability and elasticity, easy access through the Web, and reduced business risks and maintenance expenses [4]. Even though Cloud has various advantages but it has certain limitations too: It can be extremely time-consuming to transfer a large amount of data into or out of a cloud environment. Cloud improves the flexibility but as it is fairly new, cloud solutions are not as flexible as they will be someday. Therefore, to overcome the limitations and to meet the challenges faced by cloud, a number of specialized solutions: NoSQL and NewSQL data stores have emerged in the last few years in an attempt to address the mentioned concerns.

4. RDBMS (Relational Database Management System)

RDBMS uses structured query language (or SQL) to define, query, and update the database and is convenient with structured data. However, using the language with other types of information is difficult because it's designed to work with structured, relationally organized databases with fixed table information, therefore users must convert all data into tables. When the data doesn't fit easily into a table, the database's structure can be complex, difficult, and slow to work with.

4.1 Limitations of RDBMS to support "big data"

First, the data size has increased tremendously to the range of petabytes- one petabyte = 1,024 terabytes. RDBMS finds it challenging to handle such huge data volumes. To address this, RDBMS added more central processing units (or CPUs) or more memory to the database management system to scale up vertically. Second, the majority of the data comes in a semi-structured or unstructured format from social media, audio, video, texts, and emails. However, the second problem related to unstructured data is outside the purview of RDBMS because relational databases just can't categorize unstructured data. They're designed and structured to accommodate structured data such as weblog sensor and financial data. Also, "big data" is generated at a very high velocity. RDBMS lacks in high velocity because it's designed for steady data retention rather than rapid growth. Even if RDBMS is used to handle and store "big data", it will turn out to be very expensive. As a result, the inability of relational databases to handle "big data" has led to the emergence of new technologies [5]: NoSQL and NewSQL data stores. Fig. 4 shows the decline in Dominance of SQL (Structured Query Language) used by relational databases.

Figure 4: Decline in Dominance of SQL

5. NoSQL Database

One of the key advances in resolving the problem of big data has been the emergence of NoSQL as an alternative database

technology that is non-relational, usually avoid join operations, distributed (means data is spread to different machines and is managed by different machines), open source (everyone can look into its code freely, update it according to his needs and compile it), typically scalable horizontally (horizontal-scaling is often based on partitioning of the data i.e. each node contains only part of the data), schema-free (not require fixed table schemas) [6]. As opposed to transactions in RDBMS, confirming to ACID. NoSQL DBMS follows the CAP theorem and thus its transactions conform to the BASE principle. Table 1 shows the comparison between RDBMS and NoSQL Databases.

Tab le 1: Comparison between RDBMS and NoSQL

RDBMS	NoSQL
Structured and organized data	Semi-structured and unorganized data
Structured query language (SQL)	No Declarative Query language
Tight Consistency	Eventual Consistency
ACID Transactions	BASE Transactions
Pre-defined Schema	No pre defined schema
Data and its relationships are stored in separate tables	Key-Value pair storage, Column Store, Document Store, Graph Databases

5.1 Axiomatic of NoSQL

5.1.1 ACID Free

ACID stands for Atomicity, Consistency, Isolation and Durability. ACID concept basically comes from the SQL environment. But in NoSQL concept of ACID is not used because of Consistency feature of SQL. In this paper we will see how ACID concept creates problems to NoSQL. As in the distributed environment, data is spread to different machines, each machine stores its data and therefore, maintenance of consistency is needed. For example, if there is change in one tupple of the table then changes are needed in each and every machine on which that particular data resides. If information regarding an updation spreads immediately, then consistency is given; if not, then inconsistency is carried out.

5.1.2 BASE

BASE - **B**asically **A**vailable replication and sharding techniques which are used in NoSQL databases to reduce the data unavailability, even if subsets of the data become unavailable for short periods of time. BA**S**E - **S**oft State ACID systems assume that data consistency is a hard requirement. Nosql systems allow data to be inconsistent and provide options for setting tunable consistency levels. BAS**E**-- **E**ventually Consistency when nodes are added to the cluster while scaling up, need for synchronization arises**.** If absolute consistency is required, nodes need to communicate when read/write operations are performed on a node Consistency over availability.

5.1.3 CAP

CAP stands for **C**onsistency, **A**vailability and **P**artition tolerance. CAP is basically a theorem (Brewer's theorem) that follows three principles (Fig. 6).

- The data available on all machines should be same in all respects and updations to be made on all machines frequently i.e. consistent data.
- Data must be available permanently and should be accessible each and every time i.e. availability.
- During machine failure or any faults in the machines database going to work fine without stopping their work i.e. partition tolerance [7].

IMPORTANT: For all of the database storage, they could take only two of those characteristics. Existing RDBMS takes Consistency and Availability but it can't be applied to Partition Tolerance. So the NoSQL takes Partition Tolerance while giving up either Consistency or Availability (Fig. 6).

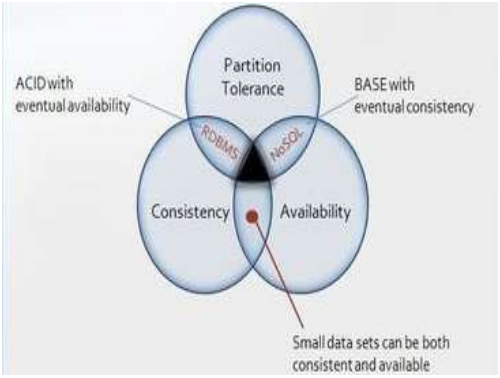

Figure 5: CAP Theorem with ACID and BASE

Figure 6: CAP Theorem visualized Databases

6. Classification of NoSQL Databases

On a basic level, the core categories of NoSQL databases are:

6.1 Key-value Stores

Data is stored as key-value pairs such that values are indexed for retrieval by keys. These systems can hold structured and unstructured data. An example is Amazon's SimpleDB.

6.2 Column-oriented Databases

These types of databases contain one extendable column of closely related data rather than sets of information in a strictly structured table of columns and rows as is found in relational databases. The Column Family databases stem from Google's internally-used BigTable, Cassandra and HBase.

6.3 Document-based Stores

Data is stored and organized as a collection of documents. Users are allowed to add any number of fields of any length to a document. They tend to store JSON-based documents in their databases. Examples of document databases include MongoDB, Apache CouchDB.

6.4 Graph Databases

Graph Databases allow for queries on the graph structure, and implementations of Graph Databases can support such queries efficiently by using well- studied graph algorithms. Graph Database examples: Neo4j, InfoGrid, AllegroGraph, InfiniteGraph [8].

Comparison between various non-relational databases is given in Table II.

6.5 Characteristics of NoSQL

- NoSQL does not use the relational data model thus does not use SQL language.
- NoSQL stores large volume of data.
- In distributed environment (spread data to different machines), we use NoSQL without any inconsistency.
- If any faults or failures exist in any machine, then in this there will be no discontinuation of any work.
- NoSQL is open source database, i.e. its source code is available to everyone and is free to use it without any overheads.
- NoSQL allows data to store in any record that is it is not having any fixed schema.
- NoSQL does not use concept of ACID properties.
- NoSQL is horizontally scalable leading to high performance in a linear way.
- It is having more flexible structure [7].

Table 2: Comparison BetweenNoSQL Databases [8]

	Google BigTable	Amazon SimpleDB	Apache CouchDB	MangoDB	Cassandra	Hbase
Data Model	Column database	Document Oriented	Document Oriented (JSON)	Document Oriented (BSON)	Column database	Column database
Interface	TCP/IP	TCP/IP	HTTP/REST	TCP/IP	TCP/IP	HTTP/REST
Storage Type	Columns	Document	Document	Document	Columns	Columns
Data Storage	GFS (Google File System)	S3 (Simple Storage Solution)	Disk	Disk	Disk	Hadoop
Query Method	Map/Reduce	string-based query language	Map/Reduce	Map/Reduce	Map/Reduce	Map/Reduce
Replication	Asynchronous / Synchronous	Asynchronous	Asynchronous	Asynchronous	Asynchronous	Asynchronous
Concurrency Control	Locks	None	MVCC (Multi Version concurrency Control)	Locks	MVCC (Multi Version concurrency Control)	Locks
Transactions	Local	No	No	No	Local	Local
Written In	C, C++	Erlang	Erlang	C++	Java	Java
Operating System	Linux Mac OS X Windows	Linux Mac OS X Windows	Linux Mac OS X Windows	Linux Mac OS X Windows	Linux Mac OS X Windows	Linux Mac OS X Windows
Characteristics	Consistency High Availability Partition Tolerance Persistence	Highly Available And Scalable	High Availability Partition Tolerance Persistence	Consistency Partition Tolerance Persistence	High Availability Partition Tolerance Persistence	Consistency Partition Tolerance Persistence

6.6 Benefits of NoSQL

- Elastic scaling: organisations are able to scale out as well as take benefits of new nodes accordingto their data storage needs.
- No need for data to fit a schema: Both types of data (structured and unstructured) can be stored as there is no fixed data model, so organisations access to much larger quantities of dataAbility to cope with hardware failure: NoSQL database was designed with redundancy in mind.
- Quick and easy development: it is very easy to change that how data is stored using refactoring orbatch processing and more.

There are many varieties of NoSQL offerings, but they are typically characterised by lack of SQL support, and non-adherence to ACID (Atomicity, Consistency, Isolation and Durability) properties. NoSQL could help enterprises manage large distributed data, but enterprises cannot afford to lose the

ACID properties. NoSQL solutions do not provide SQL support, which most current enterprise applications require, this pushes enterprises away from NoSQL. So, new data-management solutions are emerging to address large data OLTP concerns, without sacrificing ACID as well as SQL interfaces [9].

7. NewSQL Database

NewSQL is a class of modern relational database management systems that seek to provide the same scalable performance of NoSQL systems for online transaction processing (OLTP) read-write workloads while still maintaining the ACID guarantees of a traditional database system [10]. The architectures of cloud computing using horizontal scaling, preserving ACID and fault-tolerant database will obviously require other research.

Few of the NewSQL databases are as follows:

- Google Spanner solutionis considered to be one of the most prominent representatives of this category, as is also VoltDB, which is based on the H-Store research project.
- Clustrix andNuoDB are two commercial projects that are also classified as NewSQL [4].

7.1 Characteristics of NewSQL

- For any application interaction, SQL as the primary mechanism
- For transactions, ACID properties support
- Support a non-locking concurrency control mechanism
- An architecture providing higher per-node performance than traditional RDBMS solutions
- Support a scale-out, shared-nothing architecture
- NewSQL systems are approx. 40-50 times faster than traditional OLTP RDBMS [9].

8. Conclusion and future work

In recent years, cloud computing has emerged as a computational paradigm that can be used to meet the continuously growing storage and processing requirements of today's applications. This paper focuses on the storage aspect of cloud computing systems, in particular, NoSQL and NewSQL data stores. These solutions have presented themselves as alternatives to traditional relational databases, capable of handling huge volumes of data by exploiting the cloud environment. Specifically, this paper reviews NoSQL and NewSQL data stores with the objectives of providing a perspective on the field, providing guidance to practitioners and researchers to choose appropriate storage solutions, and identifying challenges and opportunities in the field. A comparison among the most prominent databases is performed on a number of dimensions including Data Model, Interface, Storage type, Data storage, Query method, Replication, Concurrency control etc. to assist practitioners in choosing the best storage solution for their needs. In future, we expect many NewSQL engines employing a variety of architectures.

References

[1] Sabia and Love Arora, "Technologies to Handle Big Data: A Survey", available at: www.sbsstc.ac.in/icccs2014/Papers/Paper2.pdf

[2] Rohit Pitre, Vijay Kolekar, "A Survey Paper on Data Mining With Big Data", International Journal of Innovative Research in Advanced Engineering (IJIRAE) Volume 1 Issue 1 (April 2014)

[3] HirdeshShivhare, Nishchol Mishra, JitendraAgarwal and SanjeevSharma, "Cloud Computing and Big Data", International Conference on Cloud, Big Data and Trust 2013, Nov 13-15, RGPV

[4] Katarina Grolinger, Wilson A Higashino, Abhinav Tiwari and Miriam AM Capretz, "Data management in cloud environments: NoSQLand NewSQL data stores", Journal of Cloud Computing: Advances, Systems and Applications 2013, 2:22, doi: 10.1186/2192-113X-2-22 available at: http://www.journalofcloudcomputing.com/content/2/1/22

[5] Market Realist, "RDBMS for data storage", available at: http://marketrealist.com/2014/07/traditional-database-systems-fail-support-big-data/

[6] OpenSource, "NewSQL — The New Way to Handle Big Data", available at: http://opensourceforu.efytimes.com/2012/01/newsql-handle-big-data/

[7] Vatika Sharma andMeenu Dave, "SQL and NoSQL Databases", Volume 2, Issue 8, August 2012International Journal of Advanced Research in Computer Science and Software Engineering Research Paper Available online at: www.ijarcsse.com

[8] Rabi Prasad Padhy, Manas Ranjan Patra and Suresh Chandra Satapathy, "RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database's", (IJAEST) International journal of advanced engineering sciences and technologies Vol No. 11, Issue No. 1, 015 – 030

[9] Rakesh Kumar, Bhanu Bhushan Parashar, Sakshi Gupta**,** Yougeshwary Sharma, Neha Gupta, "Apache Hadoop, NoSQL and NewSQL Solutions of Big Data", International Journal of Advance Foundation and Research in Science & Engineering (IJAFRSE) Volume 1, Issue 6, October 2014. Impact Factor: 1.036, Science Central Value: 10.33

[10] NewSQL, available at: https://en.wikipedia.org/wiki/NewSQL

DOI: 10.18535/Ijecs/v5i1.24

Authors Profile

Deepika Aggarwal received BCA and MCA degrees from Panjab University and Punjab Technical University in 2011 and 2014 respectively. She is Assistant Professor in Department of Computer Science and IT, DAV College, Sector-10, Chandigarh, Panjab University, India. Her research areas include Network Security and Database Management System.

Sonika received BCA and MCA degrees from Panjab University in 2009 and 2012 respectively. She is Assistant Professor in Department of Computer Science and IT, DAV College, Sector-10, Chandigarh, Panjab University, India. Her research areas include Cloud Computing, Big Data and Database Management System.

Roopam received BCA and MCA degrees from Panjab University in 2010 and 2013 respectively. She is Assistant Professor in Department of Computer Science and IT, DAV College, Sector-10, Chandigarh, Panjab University, India. Her research areas include Computer Networks and Database Management System