INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

A NOVEL TECHNIQUE IN NoSQL DATA EXTRACTION

Renu Chaudhary *[1], Gagangeet Singh[2]
[*1] Computer Science, Chandigarh Engineering College, Landran, Punjab, INDIA
[2] Computer Science, Chandigarh Engineering College, Landran, Punjab, INDIA
* Correspondence Author: renufazilka1@gmail.com

Abstract:
NoSQL databases (commonly interpreted by developers as „not only SQL databases" and not „no SQL") is an emerging alternative to the most widely used relational databases. As the name suggests, it does not completely replace SQL but compliments it in such a way that they can co-exist. In this paper we will be discussing the NoSQL data model, types of NoSQL data stores, characteristics and features of each data store, query languages used in NoSQL, advantages and disadvantages of NoSQL over RDBMS and the future prospects of NoSQL.
Motivation/Background: ***NoSQL systems exhibit the ability to store and index arbitrarily big data sets while enabling a large amount of concurrent user requests.***
Method: ***Many people think NoSQL is a derogatory term created to poke at SQL. In reality, the term means Not Only SQL. The idea is that both technologies can coexist and each has its place.***
Results: ***Large-scale data processing (parallel processing over distributed systems); Embedded IR (basic machine-to-machine information look-up & retrieval); Exploratory analytics on semi-structured data (expert level); Large volume data storage (unstructured, semi-structured, small-packet structured).***
Conclusions: ***This study report motivation to provide an independent understanding of the strengths and weaknesses of various NoSQL database approaches to supporting applications that process huge volumes of data; as well as to provide a global overview of this non-relational NoSQL databases.***

Keywords:
NoSQL, Big Data, SQL, Database, Scalability, MySQL

1. INTRODUCTION

The problem with relational model is that it has some scalability issues, that is, performance degrades rapidly as data volumes increases. This led to the development of a new data model i.e. NoSQL. Though the concept of NoSQL was developed a long time ago, it was after the introduction of database as a service (DBaaS) that it gained a prominent recognition. Because of the high scalability provided by NoSQL, it was seen as a major competitor to the relational database model. Unlike RDBMS, NoSQL databases are designed to easily scale out as and when they grow. Most NoSQL systems have removed the multi-platform support and some extra unnecessary features of RDBMS, making them much more lightweight and efficient than their RDMS counterparts. The NoSQL data model does not guarantee ACID properties (Atomicity, Consistency, Isolation and Durability) but instead it guarantees BASE properties (Basically

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

Available, Soft state, Eventual consistency).It is in compliance with the CAP (Consistency, Availability, Partition tolerance) theorem.

Of the many different data-models, the relational model has been dominating since the 80s, with implementations like Oracle databases [36], MySQL [35] and Microsoft SQL Servers [34] - also known as Relational Database Management System (RDBMS). Lately, however, in an increasing number of cases the use of relational databases leads to problems both because of deficits and problems in the modeling of data and constraints of horizontal scalability over several servers and big amounts of data. There are two trends that bringing these problems to the attention of the international software community: 1. The exponential growth of the volume of data generated by users, systems and sensors, further accelerated by the concentration of large part of this volume on big distributed systems like Amazon, Google and other cloud services. 2. The increasing interdependency and complexity of data accelerated by the Internet, Web2.0, social networks and open and standardized access to data sources from a large number of different systems.

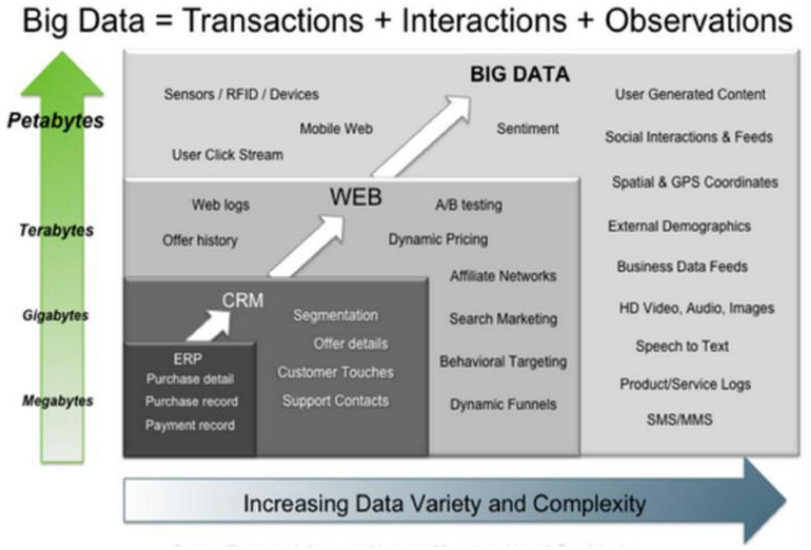

Figure 1: Big Data Transactions with Interactions and Observations. (Source: http://hortonworks.com/blog/7-key-drivers-for-the-big-data-market/)

Organizations that collect large amounts of unstructured data are increasingly turning to non-relational databases, now frequently called NoSQL databases [4]. NoSQL databases focus on analytical processing of large scale datasets, offering increased scalability over commodity hardware [8]. Computational and storage requirements of applications such as for Big Data Analytics [9], Business Intelligence [10] and social networking over peta-byte datasets have pushed SQL-like centralized databases to their limits [5]. This led to the development of horizontally scalable, distributed non-relational data stores, called No-SQL databases, such as Google's Bigtable [6] and its open-source implementation HBase [33] and Facebook's Cassandra[7]. The emergence of distributed key-value stores, such as Cassandra and Voldemort [44], proves the efficiency and cost effectiveness of their approaches [3]. The main limitations with RDBMS are it is hard to scale with Data warehousing, Grid, Web 2.0. and Cloud

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

applications [16]. Pokorny, J. (2011), focuses on NoSQL databases in context of cloud computing, particularly their horizontal scalability and concurrency model [17]. NoSQL databases are differing from Relational Database Management Systems (RDBMS) But NoSQL databases did not guarantee ACID properties [19]. The non-relational databases raised in recent years Motivated by requirements of Web 2.0 applications [2]. The strict relational schema can be a burden for web applications like blogs, which consist of many different kinds of attributes. Text, comments, pictures, videos, source code and other information have to be stored within multiple tables. Since such web applications are very agile, underlying databases have to be flexible as well in order to support easy schema evaluation [2]. Adding or removing a feature to a blog is not possible without system unavailability if a relational database is being used. NoSQL systems exhibit the ability to store and index arbitrarily big data sets while enabling a large amount of concurrent user requests [8].

In order to guarantee the integrity of data, most of the classical database systems are based on transactions. This ensures consistency of data in all situations of data management. These transactional characteristics are also known as ACID (Atomicity, Consistency, Isolation, and Durability) [32]. However, scaling out of ACID-compliant systems has shown to be a problem. Conflicts are arising between the different aspects of high availability in distributed systems that are not fully solvable - known as the CAP- theorem [38]: **Strong Consistency**: all clients see the same version of the data, even on updates to the dataset - e. g. by means of the two-phase commit protocol (XA transactions), and ACID, **High Availability**: all clients can always find at least one copy of the requested data, even if some of the machines in a cluster is down, **Partition-tolerance**: the total system keeps its characteristic even when being deployed on different servers, transparent to the client. The CAP-Theorem postulates that only two of the three different aspects of scaling out are can be achieved fully at the same time.

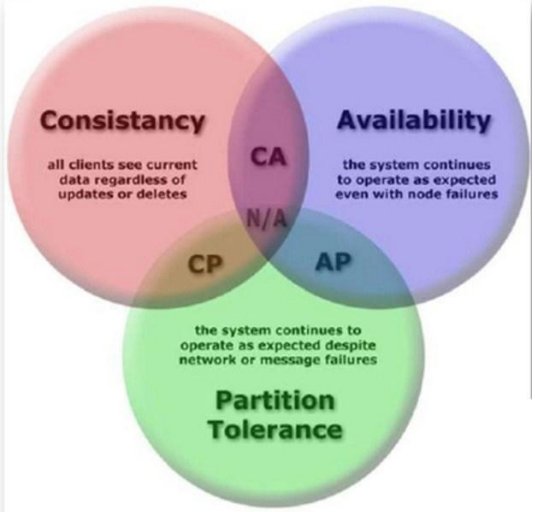

Figure 2: Characteristics of NoSQL Database (Source: NoSQLtips.blogspot.com)

Many of the NoSQL databases above all have loosened up the requirements on Consistency in order to achieve better Availability and Partitioning. This resulted in systems know as BASE

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

(Basically Available, Soft-state, Eventually consistent) [39]. These have no transactions in the classical sense and introduce constraints on the data model to enable better partition schemes. Han, J., Haihong, E., Le, G., & Du, J. (2011) classifies NoSQL databases according to the CAP theorem [14]. Tudorica, B. G., & Bucur, C. (2011), compares using multiple criteria between several NoSQL databases [15]. Primary Uses of NoSQL Database (1) Large-scale data processing (parallel processing over distributed systems); (2) Embedded IR (basic machine-to-machine information look-up & retrieval); (3) Exploratory analytics on semi-structured data (expert level); (4) Large volume data storage (unstructured, semi-structured, small-packet structured). Accordingly, they provide relatively inexpensive, highly scalable storage for high-volume, small-packet historical data like logs, call-data records, meter readings, and ticker snapshots (i.e., ―big bit bucket‖ storage), and for unwieldy semi-structured or unstructured data (email archives, xml files, documents, etc.). Their distributed framework also makes them ideal for massive batch data processing (aggregating, filtering, sorting, algorithmic crunching (statistical or programmatic), etc.). They are good as well for machine-to-machine data retrieval and exchange, and for processing high-volume transactions, as long as ACID constraints can be relaxed, or at least enforced at the application level rather than within the DMS. Finally, these systems are very good exploratory analytics against semi-structured or hybrid data, though to tease out intelligence, the researcher usually must be a skilled statistician working in tandem with a skilled programmer.

TYPES OF NoSQL

NoSQL can be categorized into 5 types:
Key-Value Store Databases: The key-value data stores are pretty simplistic, but are quiet efficient and powerful model. It has a simple application programming interface (API). A key value data store allows the user to store data in a schema less manner. The data is usually some kind of data type of a programming language or an object. The data consists of two parts, a string which represents the key and the actual data which is to be referred as value thus creating a „key-value‟ pair. These stores are similar to hash tables where the keys are used as indexes, thus making it faster than RDBMS Thus the data model is simple: a map or a dictionary that allows the user to request the values according to the key specified. The modern key value data stores prefer high scalability over consistency. Hence ad-hoc querying and analytics features like joins and aggregate operations have been omitted. High concurrency, fast lookups and options for mass storage are provided by key-value stores. One of the weaknesses of key value data sore is the lack of schema which makes it much more difficult to create custom views of the data.
Key value data stores can be used in situations where you want to store a user‟s session or a user‟s shopping cart or to get details like favourite products. Key value data stores can be used in forums, websites for online shopping etc. Although key-value data stores existed for long time ago, the development of large number of recent key value data store was influenced by the introduction of Amazon‟s Dynamo. Some notable DBaaS providers using key-value data stores are mentioned below.

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

Amazon DynamoDB: Amazon DynamoDB is a newly released fully managed NoSQL database service offered by Amazon that provides a fast, highly reliable and cost-effective NoSQL database service designed for internet scale applications. It is implemented using Amazon‟s Dynamo model. It offers low, predictable latencies at any scale. It stores data on solid state drives (SSD) instead of traditional hard drives thus providing faster access to the data. The data is replicated synchronously across multiple AWS Availability Zones in an AWS Region to provide built-in high availability and data durability. It replicates data across at least three data centers, thus providing high availability and durability even under complex failure scenarios.
RIAK : Riak is a distributed, fault tolerant, open source database developed by Basho technologies using C, Erlang and JavaScript. It implements principles from Amazon‟s Dynamo paper. It has a flexible data schema. It offers high availability, partition tolerance and persistence. Components of Riak are Riak Clients, Webmachine, Protocol Buffers, Riak Replication, Riak SNMP/JMX, Riak KV, Riak Search, Riak Pipe and Riak Core.It can be used for following purposes:
-Managing personal information of the user for social networking websites or MMORPGs(Massively Multiplayer Online Role Playing Games)
-To collect checkout or POS(Point of sales) data
-Managing Factory control and Information systems
-Building Mobile Applications on cloud etc
-Riak should be avoided for highly centralized data storage projects with fixed, unchanging data structures. Riak is used by Mozilla, AOL and Comcast.

2. ADOPTION OF NoSQL DATABASE

The acronym NoSQL was coined in 1998. Many people think NoSQL is a derogatory term created to poke at SQL. In reality, the term means Not Only SQL. The idea is that both technologies can coexist and each has its place. The NoSQL movement has been in the news in the past few years as many of the Web 2.0 leaders have adopted a NoSQL technology. Companies like Facebook, Twitter, Digg, Amazon, LinkedIn and Google all use NoSQL in one way or another. Couchbase Survey [37] was conducted in the year 2012. Key data points from the Couchbase NoSQL survey include:
 Nearly half of the more than 1,300 respondents indicated they have funded NoSQL projects in the first half of this year. In companies with more than 250 developers, nearly 70% will fund NoSQL projects over the course of 2012.
 49% cited rigid schemas as the primary driver for their migration from relational to NoSQL database technology. Lack of scalability and high latency/low performance also ranked highly among the reasons given for migrating to NoSQL (see chart below for more details).
 40% overall say that NoSQL is very important or critical to their daily operations, with another 37% indicating it is becoming more important.

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

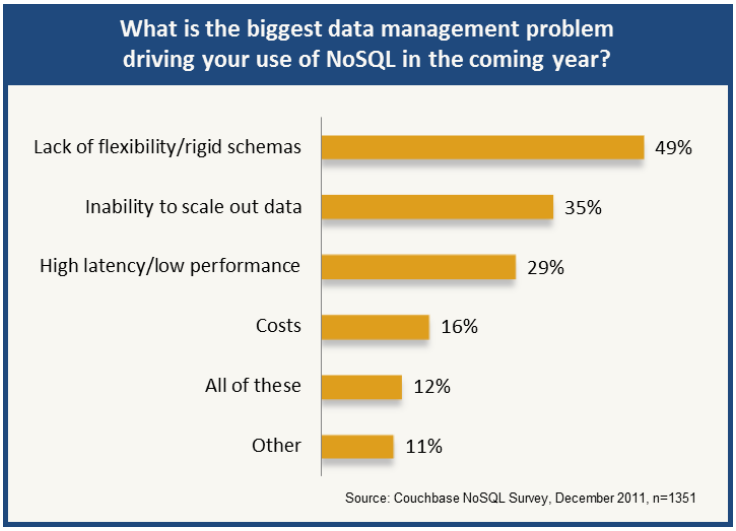

Figure 3: Key problems-driving to NoSQL databases

Organizations that have massive data storage needs are looking seriously at NoSQL. And NoSQL Database expert are highly demanded for most of the developing organizations.

3. PROPOSED WORK

Require:

The number of mappers on each rack

$\{m_1, m_2, \ldots\ldots m_n\}$

A reducer state tuple: $\{r_1, r_2, \ldots\ldots r_n\}$

N ← No of racks

M ← No of mappers

R← no of total reducers

State_tuples[N]←{0,0,…….0}

For i=1 to R do

Minimal ← infinite

For j=0 to N do

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

Traffic = $(M-2m_j).(state\_tuple[j]+1)+m_jR$

If traffic <minimal then

Candidate = j

End if

End for

State_tuple[candidate]++

End for

Return state_tuple

4. RESULTS

Computational and storage requirements of applications such as for Big Data Analytics, Business Intelligence and social networking over peta-byte datasets have pushed sql-like centralized databases to their limits [8]. This led to the development of horizontally scalable, distributed non-relational No-SQL databases. We speculate some of the major (primarily) uses of NoSQL Databeses: Large-scale data processing (parallel processing over distributed systems); Embedded IR (basic machine-to-machine information look-up & retrieval); Exploratory analytics on semi-structured data (expert level); Large volume data storage (unstructured, semi-structured, small-packet structured) NoSQL is a large and expanding field, for the purposes of this paper - characteristics (features and benefits of NoSQL databases); classification (categories four on their features); comparison and evaluation (with a matrix on basis of few attributes- design, integrity, indexing, distribution, system) of different types of NoSQL databases; and current state of adoption of NoSQL databases. This study report motivation to provide an independent understanding of the strengths and weaknesses of various NoSQL database approaches to supporting applications that process huge volumes of data; as well as to provide a global overview of this non-relational NoSQL databases.

5. ACKNOWLEDGEMENTS

Gagangeet Singh Aujla, Chandigarh Engineering College (Guide)
Gagandeep Singh, Chandigarh Engineering College (Co-Guide)

INTERNATIONAL JOURNAL of RESEARCH – GRANTHAALAYAH
A knowledge Repository
Science

6. REFERENCES

[1] http://en.wikipedia.org/wiki/NoSQL
[2] Hecht, R., & Jablonski, S. (2011, December). NoSQL evaluation: A use case oriented survey. In Cloud and Service Computing (CSC), 2011 International Conference on (pp. 336-341). IEEE.
[3] Use relational DBMS, N. (2009). Saying good-bye to DBMSs, designing effective interfaces. Communications of the ACM, 52(9).
[4] Leavitt, N. (2010). Will NoSQL databases live up to their promise?. Computer,43(2), 12-14.
[5] Abadi, D. J. (2009). Data management in the cloud: Limitations and opportunities. IEEE Data Eng. Bull, 32(1), 3-12.
[6] Chang, Fay, et al. "Bigtable: A distributed storage system for structured data."ACM Transactions on Computer Systems (TOCS) 26.2 (2008): 4.
[7] Lakshman, A., & Malik, P. (2010). Cassandra—A decentralized structured storage system. Operating systems review, 44(2), 35.
[8] Konstantinou, I., Angelou, E., Boumpouka, C., Tsoumakos, D., & Koziris, N. (2011, October). On the elasticity of NoSQL databases over cloud management platforms. In Proceedings of the 20th ACM international conference on Information and knowledge management (pp. 2385-2388). ACM.
[9] Russom, P. (2011). big data analytics. TDWI Best Practices Report, 4 th Quarter 2011.
[10] Luhn, H. P. (1958). A business intelligence system. IBM Journal of Research and Development, 2(4), 314-319.
[11] G. DeCandia, et al.,(2007) "Dynamo: amazon's highly available key-value store," in SOSP '07 Proceedings of twenty-first ACM SIGOPS, New York, USA, pp. 205-220. [12] K. Orend, (2010) "Analysis and Classification of NoSQL Databases and Evaluation of their Ability to Replace an Object-relational Persistence Layer," Master Thesis, Technical University of Munich, Munich.
[13] R. Cattell, (2010) "Scalable SQL and NoSQL Data Stores," ACM SIGMODRecord, vol. 39.
[14] Han, J., Haihong, E., Le, G., & Du, J. (2011, October). Survey on NoSQL database. In Pervasive Computing and Applications (ICPCA), 2011 6th International Conference on (pp. 363-366). IEEE.
[15] Tudorica, B. G., & Bucur, C. (2011, June). A comparison between several NoSQL databases with comments and notes. In Roedunet International Conference (RoEduNet), 2011 10th (pp. 1-5). IEEE.
[16] Padhy, R. P., Patra, M. R., & Satapathy, S. C. (2011). RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database.