LUND UNIVERSITY

School of Economics and Management

Department of Informatics

Nosql database: New era of databases for big data solution

Individual assignment for INFN45

Author: Conerlious Sagandira, 97/03/07

Wordcount: 2190

Table of Content

1. Introduction

Although advancements in computing technologies have allowed organizations to gather big datasets that can be used for decision making by using business intelligence tools (Fedusko, Teras & Gregus, 2020). According to the United Nations report on digital economy (2019), there has been a digital revolution underway whereby a variety of tremendous data generated by various sources such as mobile devices, information archives and enterprise applications. To be able to store and monitor the flow of data, most organizations have been traditionally relying on relational databases (RDB) as they are designed for storing structured data in an organized way for easy access and manipulation (Anselma, Peovesana & Terenziani, 2018). However, a few years ago, huge volumes of various data types have been generated at a rapid rate leading the Big Data era (Wani & Jabi, 2020). As a result, RDB are falling behind in the Big Data era as they lack flexibility to handle different data types with unstructured, semi-structured and structured data at the same time (Lee, Tang & Choi, 2013). Moreover, RDB does not perform well when receiving huge amounts of data at a fast speed as it can be very complex to relate all tables with different data types. Furthermore, it can be very expensive to store huge volumes of data using RDB. To ward off these limitations NoSQL databases such as column-oriented, graph and document-oriented databases can be used to mitigate limitations of RDB. MongoDB, MariaDB, Cassandra and Amazon DynamoDB will be used as real-life examples of NoSQL database management systems. Overall, for one to be able to choose a suitable NoSQL database that solves their situation, a CAP theorem is suggested in this paper.

Key words: Relational database, Big Data, NoSQL

1.1 Problem

Due to Big data era, various data types are being generated in huge volumes at a rapid rate leading to RDB failure to handle Big Data (Wani & Jabi, 2020). As there will be a need to receive and store tremendous amounts of data at a rapid rate, RDB lacks flexibility, performance and simplicity to handle complex data (Saeed & Abed, 2020). Thus, there is a need for database solutions that can be used to mitigate limitations of RDB.

1.2 Purpose

This paper aims to address limitations of RDB due to the Big Data era based on the 4Vs *(variety, velocity, volume and veracity).* Moreover, the scope of the paper is to suggest the types of NoSQL databases that can be used to ward off limitations in the Big Data era and show how one can choose a suitable NoSQL database that suits their needs by using the CAP theorem.

1.3 Delimitation

The explanation on how RDB operates will be generalized without explaining specific types of RDB or its management system (RDBMS). CAP theorem will not be elucidated in detail but will only be highlighted as a suggestion for choosing a proper NoSQL database.

1.4 Research question

1. Why are relational databases falling behind in the Big Data era?

2. What types and examples of NoSQL databases management systems that can be used to mitigate limitations of relational databases?

2. Literature review

2.1 Relational databases

According to Codd (1981), “*a relational database (RDBS) is a digital database based on the relational model of data that organizes data into tables which can be linked—or related—based on data common to each other*”. For a relational database to excel, Structured Query Language (SQL) can be used to retrieve and store “real-time” operational data. Traditionally, for good database application, organizations have been relying on RDB because of their Atomicity, Consistency and Isolation (ACID) compliant nature which promotes the integrity of the entire database through atomicity, consistency, isolation and durability (Leavitt, 2010). However, since RDBS are developed with conventional ACID, they tend to be more inclined towards consistency whereas BASE ones such as NoSQL focus on availability over consistency.

2.2 Big Data

According to Mauro, Greico and Grimaldi (2016), Big data is defined as “*the Information asset characterised by such a High Volume, Velocity, Variety and Veracity to require specific Technology and Analytical Methods for its transformation into Value*”. Big Data can be explained as the 4Vs which are: Volume, Variety and Velocity as shown in the diagram *(fig.1)* below.

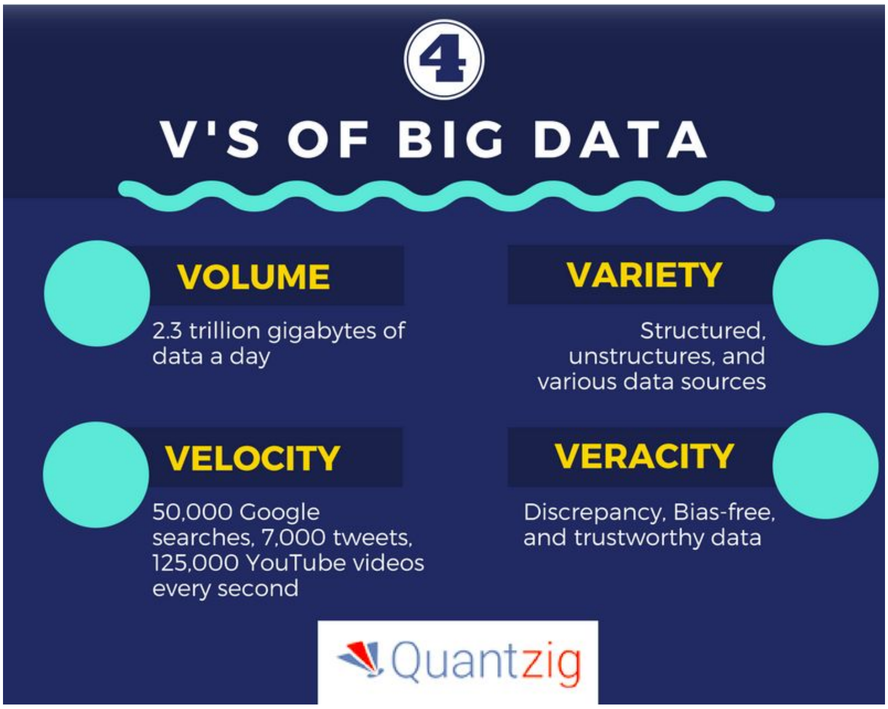
4
V'S OF BIG DATA
VOLUME
2.3 trillion gigabytes of data a day
VARIETY
Structured, unstructures, and various data sources
VELOCITY
50,000 Google searches, 7,000 tweets, 125,000 YouTube videos every second
VERACITY
Discrepancy, Bias-free, and trustworthy data
Quantzig

Fig.1*:* The 4Vs of Big Data
Source: (Quantzig, 2018)

Volume

Volume refers to a high amount or magnitude of data that has to be analyzed. For example, the amount of data that is generated on a daily basis in Europe through credit card transactions is huge (Quantzig, 2018).

Velocity

Velocity refers to the rate at which the data is being received or acted upon (Mauro, Greico & Grimaldi, 2016). For example, the rapid rate at which Facebook and Twitter receives data per second through videos, messages and photos from all its users in the whole world.

Variety

As there are different technologies that can be used to transfer information, various types of data are generated either as videos, messages, text or photos (Mauro, Greico & Grimaldi, 2016). All this type of data can be classified into structured, semi-structured, unstructured and mixed data as they all come from different sources.

Veracity

This is whereby concerns about security of the data which is being generated rapidly are raised. Meaning the trustworthiness of big data depends on if, without inconsistencies, the data is representative and removes prejudices (Quzntzig, 2018).

2.3 Drawbacks of Relational databases in Big Data

Since the invention of RDB by Edgar Codd in 1970, relational databases have been one of the best database solutions organizations could seek for. However, RDB has started falling behind as it fails to cope up with the 4Vs of Big data (Shukla, 2014). Thus, using RDB have drawbacks such as: rational (inflexibility), performance hit complexity and scalability, will be elucidated in connection to its failure to deal with the Big Data 4Vs.

Inflexibility of schemas

As schemas are normalized in RDB, multiple joins are used to connect all the data. However, joins are expensive especially when collecting huge amounts of various data types. Moreover, since RDB are rational, one needs to pre-define the structure of data that is going to be stored (Lee, Tang & Choi, 2013). This means the number of columns, non-null constraints and foreign key relationships of the tables and the data types to be stored has to be known before even getting started.

Performance hit

It is difficult to use RDB in the environments of high-speed data generation such as social media (Facebook and Twitter) since it was designed for steady data retention instead of rapid growth (Shukla, 2014). Although it is possible to receive data with high velocity in RDB, it mostly comes with a tradeoff between flexibility and space (memory).

Complexity

Since RDB makes use of SQL, this means all the data going to be stored has to fit into tables. If the data does not fit, then a structure for that specific data will be needed (Shukla, 2014)..

However, it is difficult to design a database structure of high volume as it can be very complex to follow through each and every column and tables of different data types (Lee, Tang & Choi, 2013).

2.4 NoQSL databases for Big Data solutions

According to Nayak and Poria (2013), NoSQL is defined as *"an approach to database design that can accommodate a wide variety of data models including key-value, document, column-oriented, graph and object-oriented formats"*. NoSQL databases were introduced because there was a need for databases that can be used to store huge amounts of data at a lower price than relational databases (Khazaei, et.al, 2016). For organizations to implement NoSQL these databases, they might need to consider consistency, availability and partition tolerance, the CAP theorem.

2.4.1 Types of NoSQL database management systems and their advantages.

NoSQL databases exist in different forms depending on the type or model it uses. This paper will only focus on some of the existing database model types which are: key-value, column-oriented, graph and document database model. Moreover, real-life examples on each database type will be suggested.

Key-value (Processing data quickly)
Amazon DynamoDB, Redis, Oracle NoSQL and Riak are some of the examples of databases which rely on key-value database models (George, 2013). To be able to store and process data that will be coming at a rapid rate, these databases enable data processing at a fast rate as well as matching with the speed at which the data is being generated (Sadalage & Fowler, 2013).

Column-oriented (Low cost)
As it can be extremely expensive to store huge amounts of data in RDBs, MariaDB, MonetDB and Apache Kudu are NoSQL databases designed to store tremendous volumes of data at a very low cost compared to RDB (George, 2013). NoSQL databases use row keys to extract columns of data which is cheaper than the traditional way of retrieving data from RDB (Sadalage & Fowler, 2013)
.

Graph (Mass storage)
To be able to store all the data, which is being generated at high volumes, more database space *(memory)* will be also needed (Sadalage &Fowler, 2013). By using graph database model, various data types can be stored since there will be no need to apply complex queries to identify the relationships between data points (George, 2013). Organizations also choose various types of database management systems such as OrientDB, Cassandra, ArangoDB and Neo4j.

.

Document database (Easy to expand)

As complex as it can be to maintain a database that receives high volume of data with various data types at a rapid rate, there are databases which are based on document model such as MongoDB, IBM Cloudant and Apache CouchDB can be used as they come with all the flexibility needed to maintain data of such nature (Big Data). The model uses a flexible schema that applies complex querying for high performance (George, 2013).

2.4.2 Implementation of NoSQL database management systems.

A CAP theorem can be used during NoSQL implementation. The main idea behind the CAP theorem is that a distributed system cannot fulfill all the three needs at the same time (George, 2013). The diagram *fig 2* shows how a CAP theorem can be used to choose a NoSQL database management system that suits one's specific needs as a way to mitigate RDB drawbacks.

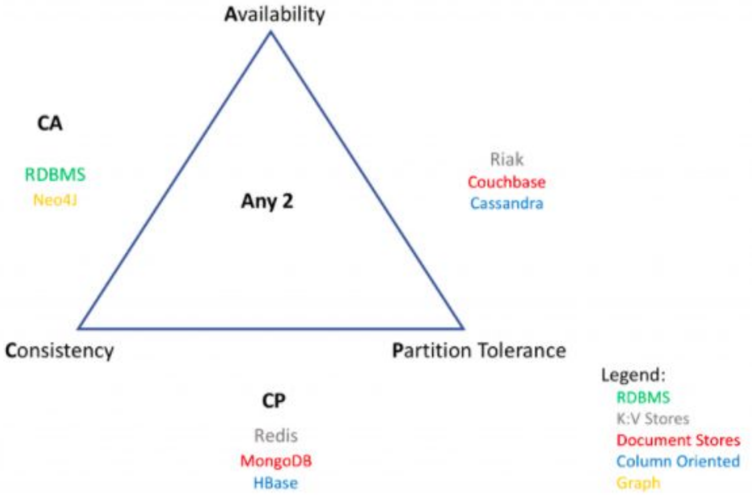

Fig. 2: CAP theorem
Source: (Han, 2011)

Consistence

This refers to the property that the correct response to each request is returned by each server, that is, a response that is appropriate to the requested specification of the service. The precise importance of quality depends on the kind of service (Gilbert & Lynch, 2012)

Availability

Refers to the property that eventually receives a response to each request. Meaning, a quick reaction is obviously preferable to a slow reaction, but even requiring an eventual reaction is sufficient to generate problems in the sense of the theorem (Gilbert & Lynch, 2012).

Partition tolerance

Despite an arbitrary number of messages being dropped (or delayed) by the network between nodes, the system continues to run. The CAP theorem means that one has to choose between consistency and availability in the presence of a network partition. (Gilbert & Lynch, 2012).

3 Discussions

The *Table 1* below addresses limitations or drawbacks of RDB and highlights NoSQL database solutions that can be used to ward off RDB drawbacks in the Big Data era as well as real-life examples of NoSQL database management systems are highlighted.

Table 1: Drawbacks of RDB and NoSQL database solutions.

RDB drawbacks	**NoSQL database solutions**	**Example of NoSQL DBMS**	**NoSQL database deliverables**
Poor performance	Key value	Amazon DynamoDB, Redis, Oracle NoSQL and Riak	Processes data quickly
Expensive	Column-oriented	MariaDB, MonetDB and Apache Kudu	Low cost
Complex	Graph	OrientDB, Cassandra, ArangoDB and Neo4j	More storage
Inflexible	Document	MongoDB, IBM Cloudant and Apache CouchDB	Easy to expand

Table 1 Shows that RDB lacks flexibility, complexity, they are expensive, and they don't perform well when subjected to the 4Vs of Big Data. To solve this, key-value, column-oriented, graph and document databases can be applied to mitigate each drawback. Moreover, Amazon DynamoDB, MariaDB, Cassandra and MongoDB are some of the examples of NoSQL databases that can be implemented in organizations.

Conclusion

As RDB lacks flexibility, high performance, simplicity and is expensive to handle complex data. This paper suggests the use of NoSQL databases to mitigate drawbacks of RDB in the Big Data era through the use of literature review. Moreover, it has been found that there are various NoSQL database models that can be used to mitigated or ward off various limitations of RDB such as: column-oriented databases can be used to store high volume of data at low cost. Real-life examples of column-oriented databases are MariaDB, MonetDB and Apache Kudu. Moreover, graph database models can be used to store unlimited amounts of data and still maintain high performance, thus, organizations can use databases such as Cassandra, ArangoDB and Neo4j databases. Document databases such as MongoDB, IBM Cloudant and Apache CouchDB can be used for more flexibility and Amazon DynamoDB, Redis, Oracle NoSQL and Riak are some of key-value databases that can be used for processing data at high speed. Overall, this paper has suggested the use of CAP theorem when it comes to NoSQL database implementation to ensure right choice which meets business needs.

4 References

Andrea De Mauro, Marco Greco , Michele Grimaldi (2016) ' A formal definition of Big Data based on its essential features', *Library review ,* 65(3), pp. 122-135.

Anselma. L, Peovesana. L, Terenziani. P (2018) *An AI approach to the Temporal indeterminacy in Relational database* , 16 edn., America : Springer.

Christopher. Austin & Fred. Kusumoto (2016) 'The application of Big Data in medicine ', *International Cardiac electrophysiology ,* 47(1), pp. 51-59.

Edgar. F. Codd (1981) 'Relational Database: A Practical Foundation for Productivity ', *Relational Database:,* 1(1), pp. 1-9.

Excellence for Research in Adaptive Systems,, 3(9), pp. 668 [Online]. Available at: https://www.mdpi.com/2079-9292/9/4/668 (Accessed: 8 December 2020).

Fedushko. F, U. Taras & M. Gregus (2020) 'Real-Time High-Load Infrastructure Transaction Status Output Prediction Using Operational Intelligence and Big Data Technologies', *Center of Information Systems.*

George. S, (2013) 'NOSQL - NOTONLY SQL', *International Journal of Enterprise Computing and Business Systems ,* 2(2), pp. 1-11

Hadjigeorgiou. C, (2013) 'RDBMS vs NoSQL: Performance and Scaling Comparison', *Computer science ,* 6(22), pp. 1-45.

Khazaei. H, Fokaefs. M, Zareian. S, Beigi-Mohammadi. N, Ramprasad, M Shtern, P Gaikwad, & M Litoiu (2016) 'How do I choose the right NoSQL solution? A comprehensive theoretical and experimental survey', *Center of Excellence for Research in Adaptive Systems,,* 8(34), pp. 185-216

Leea. K. K, Tangb. W, Choia. K (2020) 'Alternatives to relational database: Comparison of NoSQL and XML approaches for clinical data storage', *Computer Methods and Programs in Biomedicine,* 110(1), pp. 99-109.

Neal Leavitt (2010) 'Will NoSQL Databases Live Up to Their Promise?', *in Computer,* 43(2), pp. 12-14.

Polding(2018) *Databases:Evolution and Change,* Available at: https://medium.com/@rpolding/databases-evolution-and-change-29b8abe9df3e (A ccessed: 4 December 2020).

Saeed. N, Abed. A (2020) 'Big Data with Column Oriented NOSQL Database to Overcome the Drawbacks of Relational Databases', *Advanced Networking and Applications,* 11(5), pp. 4423-4428.

Seref Sagiroglu; Duygu Sinanc (2016) 'Big data Review', *International Conference on Collaboration Technologies and Systems (CTS),* (), pp. 42-47

United Nations (2019) *DIGITAL ECONOMY REPORT*, Geneva: UNICAF.

Wani. M. A, Jabi. S (2020) 'Big Data: Issues, Challenges, and Techniques in Business Intelligence', *Big Data Analytics ,* 5(2), pp. 613-628.

Yishan Li; Sathiamoorthy Manoharan (2013) 'A performance comparison of SQL and NoSQL databases', *IEEE conference ,* 4(2), pp. 8-33.

Seth Gilbert & Nancy Lynch (2012) 'Perspectives on the CAP Theorem', *IEEE,* 45(2), pp. 30-36.

Pramod J. Sadalage & Martin Fowler (2013) *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*, 13 edn., United States: Pearson Education, Inc.

Quantzig (2018) *4Vs of Big Data: Everything You Need To Know,* Available at: https://www.quantzig.com/blog/4-vs-big-data (Accessed: 2 february 2021)