

Review of the current state of a problem of processing big data

Aiman N Moldagulova* , Azamat Zhubandykov, Askar Mustafin

International Information Technology University, Almaty, Kazakhstan

**Corresponding author's e-mail: a.moldagulova@iitu.kz*

Received 1 May 2015, www.cmmt.lv

Abstract

The problem around the processing of large amount data sets is solved within the Big Data paradigm. Big data is important in many diverse areas, such as science, social media, enterprise and etc. This paper refers to various ways to store data to define the differences between traditional storage systems and current approaches to dealing with large data sets. Technologies such as MapReduce, NoSQL and processing of event streams in real time are discussed.

Keywords: Big Data, MapReduce, Hadoop, NoSQL, MongoDB

1 Introduction

As it is well known, IT- industry has it's own buzzwords - words that everyone usually hear, use them in their daily life, but only some of us really know what is hidden behind these words and how to use them correctly. Nowadays, one of the most well-known development in this field of study is «Big Data».

Big Data figures unusual size (capacity over traditional databases) or generated at an incredible rate, such as gathered from social networking providers. Big data is often unstructured - they cannot clearly fit into a pre-defined structure of the database. The main advantage of big data is that you can first collect and then analyze these data to obtain statistics, structure and relationships, without knowing in advance what to look for.

Big Data Analytics allows moving from traditional database queries to analyze what is happening, and to analyze the options for product development planning. It will be useful for large companies.

Hundreds of gigabytes/ terabytes/ megabytes of data are not valuable in themselves. Their main purpose is to help to understand the past and predict the future.

Three characteristics which define Big Data are volume, variety and velocity. They have created the need a new class of capabilities to augment the way things is done today to provide better line of site and controls over our existing knowledge domains and the ability to act them.

2 Big Data

Big Data information technology is a series of accesses, tools and modes for processing of structured and unstructured data and big amounts of the significant variously perceived by man for results that are effective in conditions of continuous growth, the distribution on multiple nodes computer network, formed in late 2000-s alternatives to traditional database management systems and solutions class of Business Intelligence. This series consist means massively parallel processing vaguely structured data, primarily decisions category NoSQL, algorithms MapReduce, software frameworks and libraries project Hadoop.

The concept of big data means working with the amount

of information and varied composition, and quite often updated and located in different sources in order to increase strength, create new products and improve competitiveness. Consulting company Forrester gives brief statement: Big Data combines engineering and technology that extracts meaning from the data in the extreme limit of practicality.

3 Data capacity

In the next eight years, the number of data in the world will be 40 zetabytes, which is equivalent to 5200 gigabytes (GB) for each person on the planet, according to researches of IDC Digital Universe, published in December 2012. 40 zetabytes equivalent to 40 trillion GB, which is 57 times more than the number of grains of sand on the beaches on the entire surface of the Earth. According to forecasts, the amount of data in the world will double every two years until 2020 [1].

Most of the data that will be produced in the period from 2012 to 2020, will generate not people, but machines in interaction with each other and other data networks. These include, for example, intelligent sensors and devices which can communicate with external devices.

Analysts also expect that in the future most of the digital information to be stored in the cloud. If the cloud now accounts for about 5% of global IT expenses, by 2020 40% of all information in the digital universe will be "tied" to cloud systems. However, the cloud will be produced mainly handling and processing of the data, but directly stored in the cloud will only be 15% of the information [2].

4 Hadoop

Hadoop is a project of Apache Software Foundation, an open source set of tools, libraries, and software framework for the development and execution of distributed applications running on clusters of hundreds or thousands of nodes. It is used to implement search and contextual mechanisms of many heavily websites, including, for Yahoo! and Facebook. Developed in Java within the computing paradigm MapReduce, according to which an application is divided into big quantity of identical elementary tasks feasible on the cluster nodes and naturally reducible to the final result.

Hadoop is a framework with open source for creating and running distributed applications that process large amounts of data [3]. Distributed computing is a broad and multi-faceted region, but Hadoop has several important distinguishing features, namely:

- Availability: Hadoop running on large clusters assembled from standard equipment, or cloud computing, for example on the basis of service Elastic Compute Cloud (EC2), offered by Amazon;
- Reliability: as Hadoop to run on standard hardware, its architecture is designed to allow frequent failures. Most failures can be treated so that the characteristics of the cluster will gradually deteriorate;
- Scalability: Hadoop scales linearly, i.e. by increasing the amount of data is sufficient to add new nodes to the cluster;
- Simplicity: Hadoop allows a user to create quickly effectual parallel code.

As of 2014 the project consists of four modules - Hadoop Common (middleware software - a set of infrastructure software libraries and utilities used for other modules and related projects), HDFS (Distributed File System), YARN (system for scheduling and managing the cluster) and Hadoop MapReduce (programming platform and executing distributed MapReduce-computing), earlier in Hadoop includes a number of other projects, became independent projects within the Apache Software Foundation [4].

Hadoop MapReduce is a software framework for distributed computing program within the paradigm MapReduce. Application developer for Hadoop MapReduce is necessary to implement the core handler that on each compute node in the cluster will provide initial conversion pairs "key - value" pairs in the intermediate set of "key - value" (a class that implements the interface Mapper, called higher-order functions on Map), and handler, which reduces the intermediate set of pairs in the final, reduced set (convolution, a class that implements the interface Reducer). Frame passes the input convolution sorted conclusions from basic handlers, mixing consists of three phases - shuffle (shuffle, the selection for you output section), sort (sorting, grouping by key findings from distributors - sorting required in the case where different sets of atomic handlers return the same key, wherein the sorting rules in this phase can be defined programmatically and using any particular internal structure of the key) and actually reduce (convolution list) - get the result set. For some types of convolution processing is required and returns a frame in this case the set of sorted pairs received basic handlers.

Hadoop MapReduce job allows a user to create both basic handlers, and with contractions written without using Java: Hadoop utility streaming lest to use as base handlers and parcel of any executable file that runs the standard input-output operating system (e.g. utilities shell UNIX) There is also a SWIG-compatible application programming interface Hadoop pipes on C++. Also, in the Hadoop distributions include the implementation of various specific base handlers and convolutions, most typically used in

distributed processing [5].

In the first versions of Hadoop MapReduce included scheduler (JobTracker), starting with version 2.0, this function was moved to YARN, and since this version module is implemented on top of Hadoop MapReduce YARN. Software interfaces for the most part retained, but no full backwards compatibility (e.g. to run programs written for earlier versions of API, to work in YARN generally require their modification or refactoring, and only under certain restrictions are possible options reverse binary compatibility).

5 NoSQL

The concept of NoSQL (Not Only SQL or No SQL) became famous from 2009. Then it was the development of web-based technologies and social services spurred many new approaches to the storage and processing of data. The developers of these applications are faced with the tasks for which traditional relational databases were either too expensive or are not productive enough. In addition, the rejection of universal popularizers "harvesting" (RDBMS) in favor of specialized steel making startups and those who have to work in scenarios so-called Big Data [6].

We must understand that NoSQL-solutions do not necessarily mean a change and a complete rejection of the DBMS. As usual, the tool must be selected by the task, and not vice versa. When people talk about NoSQL, typically list the following advantages.

Scalability. Horizontal scaling existing traditional database is usually time-consuming, expensive and effective only up to a certain level of challenge. At the same time, many NoSQL-based solutions designed to scale horizontally and do it "on the fly." Therefore, this procedure is usually simpler and more transparent in NoSQL, than in the DBMS.

Database performance on one node and not in the cluster is also an important parameter. For many tasks, such properties of traditional DBMS, both transactional isolation changes reliability within a single node or even the relational model is not always needed in full. Therefore, the rejection of these properties (all or some) allows NoSQL sometimes achieve better performance on one node than traditional solutions.

Reliable operation in circumstances where failure or network unavailability of iron - a common occurrence, is one of the many properties of solutions NoSQL. The main way to ensure it - it's replication. Replication itself is not a unique feature of NoSQL, but here, as when zooming, it plays an important role the efficiency and ease of making changes to an existing installation. Going to work in database replication mode - this is a simple task for most NoSQL-making.

Simplicity of administration and development is also an important argument in favor of NoSQL-technologies. A quantity of problems associated with scaling and replication, representing a significant challenge and requires extensive specialist expertise in traditional DBMS NoSQL in a matter of minutes. Setup and configuration tasks, the very use of NoSQL-making is usually much easier and less time-consuming than in the case with the DBMS.

Therefore NoSQL-systems have become the obvious choice for many startups, where the speed of the development and implementation is key.

In contrast to the relational model that preserves the

logical business entity application to various physical tables to normalize, NoSQL storage with these entities operate as a cohesive objects, as shown in (Figure 1).

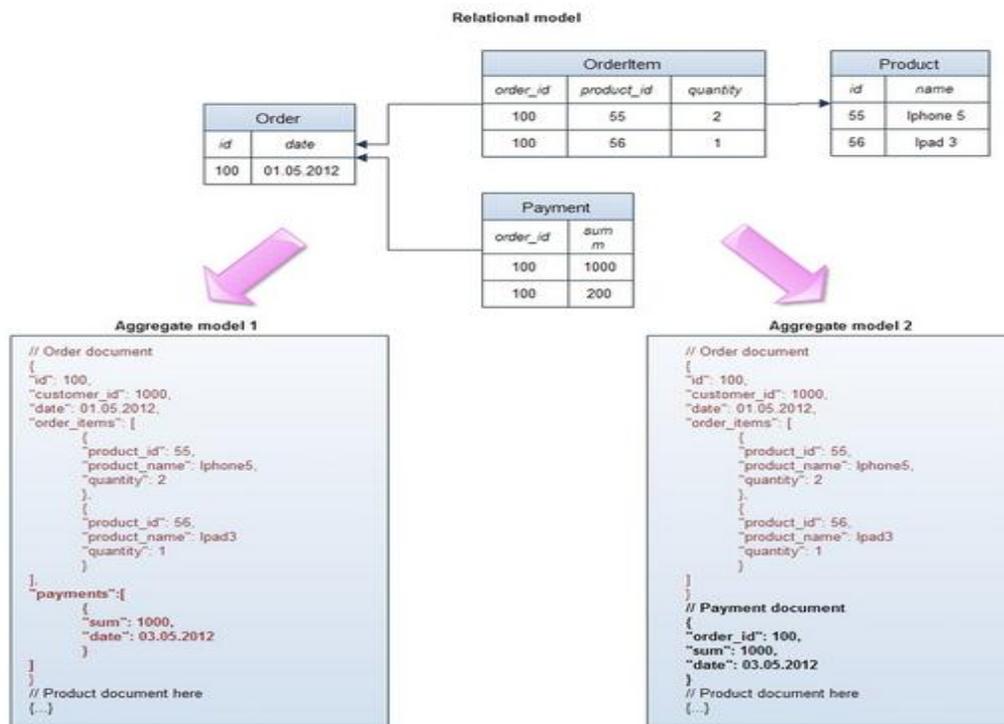


FIGURE 1 Aggregate model (source: Pramod J. Sadalage, Martin Fowler)

This example illustrates conceptual units for standard relational model of e-commerce "order - order item - payments - product." In both cases, the order is combined with the position of one entity, with each position keeps a link to the product and some of its attributes, such as name (such denormalization is necessary not to request the product to the extraction of the order - the main rule of distributed systems - at least "Join" between objects). In one aggregate payments combined with the order and are an integral part of the object, in the other - in a separate object. This demonstrates the main design rule data structure NoSQL databases - it must comply with the requirements and the application to be as optimized for the most frequent requests.

6 NoSQL vs SQL

NoSQL doesn't require schemas like SQL does meaning it can process information much quicker. With SQL, schemas (another word for categories) had to be predetermined before information was entered. That made dealing with unstructured information extremely difficult because companies never knew just what categories of information they would be dealing with. NoSQL doesn't require schemas so it can handle unstructured information easier and much quicker. Also, NoSQL can handle and process data in real-time. Something SQL doesn't do.

Another advantage to NoSQL computing is the scal-

ability it provides. Unlike SQL, which tends to be very costly when trying to scale information and isn't nearly as flexible, NoSQL makes scaling information a breeze. Not only is it cheaper and easier, but it also promotes increased data gathering. With SQL companies had to be very selective in the information they gathered and how much of it they gathered. That placed restrictions on growth and revenue possibilities. Because of NoSQL's flexibility and scalability, it promotes data growth. That's good for businesses and it's good for the consumer.

NoSQL is also extremely valuable and important for cloud computing. One of the main reasons we've seen such a rise in big data's prominence in the mainstream is because of cloud computing. Cloud computing has drastically reduced the startup costs of big data by eliminating the need of costly infrastructure. That has increased its availability to both big and small business. Cloud computing has also made the entire process of big data, from the gathering stages to analyzing and implementing, easier for companies. Much of the process is now taken care of and monitored by the service providers. The increased availability of big data means that companies can better serve the general public [7].

7 MongoDB

MongoDB (from "humongous") is a cross-platform document-oriented database. Classified as a NoSQL database, MongoDB eschews the traditional table-based relational

database structure in favor of JSON-like documents with dynamic schemas (MongoDB calls the format BSON), making the integration of data in certain types of applications easier and faster.

MongoDB has a high read / write speed and scalability but safety and integrity of the data is not as good. Mongo has an excellent implementation of replication, which is

fairly easy to install and set up or sharing (the ability to spread data across multiple servers), which is also quite easy to install.

The following Table 1 presents the various SQL terminology and concepts and the corresponding MongoDB terminology and concepts.

SQL Terms/Concepts	MongoDB Terms/Concepts
Database	database
Table	collection
Row	document or BSON document
Column	field
Index	index
table joins	embedded documents and linking
primary key	primary key
Specify any unique column or column combination as primary key.	In MongoDB, the primary key is automatically set to the <code>_id</code> field.
aggregation (e.g. group by)	aggregation pipeline
	See the SQL to Aggregation Mapping Chart.

Figure 2 shows a sharp distinction between performance of MongoDB and MySQL [8]. Significantly low number of threads and records are represent low latency of MySQL with comparing to MongoDB, once number of threads and records amplifies the latency of MongoDB conversely decreases. This suggests that for large volumes if data is better to use MongoDB (Figure 2).

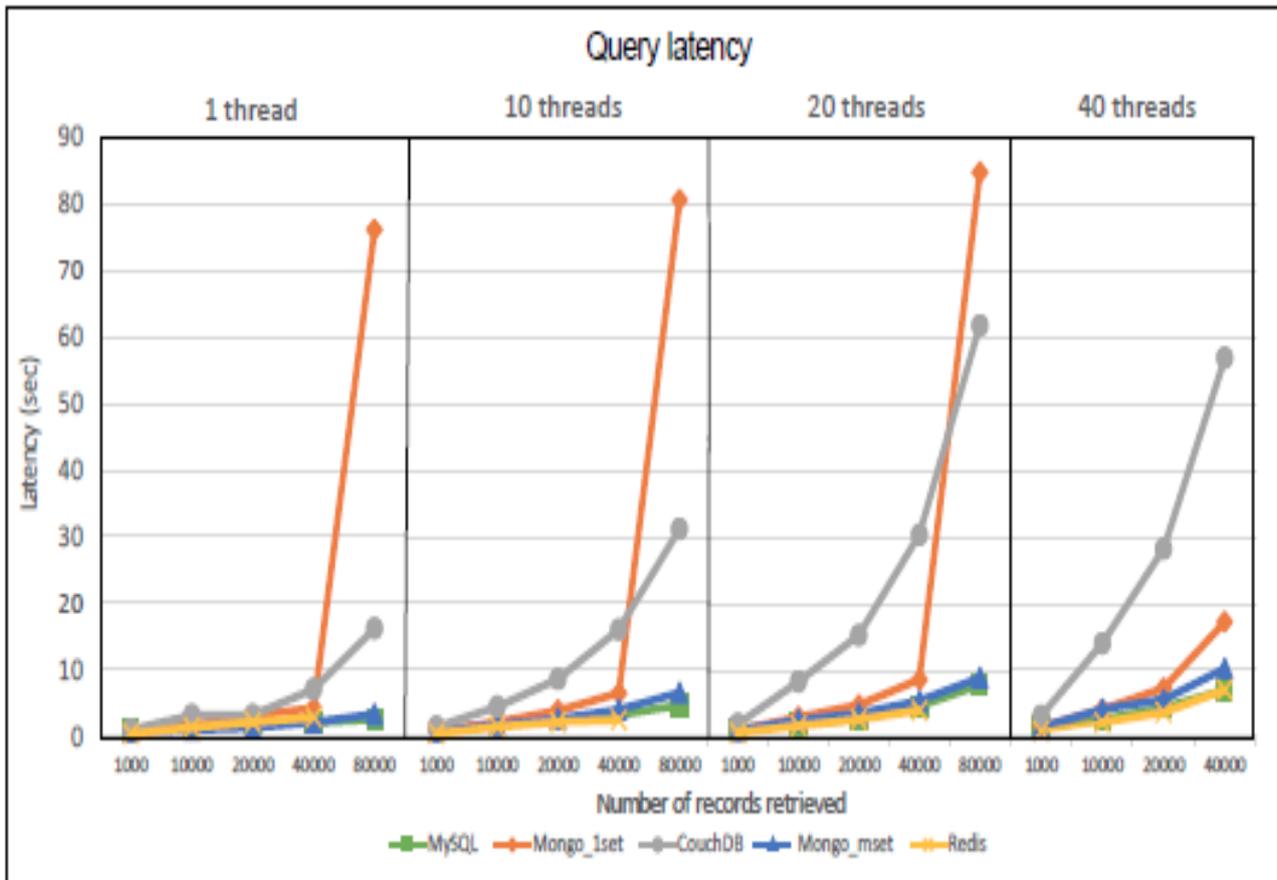


FIGURE 2 Performance of MongoDB and MySQL

8 Conclusion

In this paper problems which delivered big data before the relational DBMS were considered. We specified that that the relational database isn't capable to cope with quickly

growing data streams. Also considered some approaches of processing of big data. We also specified that all instruments of processing of big data decide effectively the class of tasks.

References

[1] The Digital Universe in 2020: Big Data Bigger Digital Shadows and Biggest Growth in the Far East 2012 <http://www.emc.com/-leadership/digital-universe/2012iview/big-data-2020.htm>

[2] Big data: The next frontier for innovation competition and productivity McKinsey Global Institute 2011 www.mckinsey.com/mgi/publications/

[3] *Hadoop Releases* <http://hadoop.apache.org/releases.html>

[4] *What is the Hadoop Distributed File System (HDFS)?* <http://www-01.ibm.com/software/data/infosphere/hadoop/hdfs/>

[5] Dean J, Ghemawat S 2004 MapReduce: Simplified Data Processing on Large Clusters OSDI'04: Sixth Symposium on Operating System Design and Implementation San Francisco

[6] Pramod J Sadalage, Fowler M 2012 NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence Addison-Wesley 2012.

[7] *Allouche G 2014 NoSQL vs. SQL: An Overview* <http://mike2.openmethodology.org/blogs/information-development/2014/10/17/nosql-vs-sql-an-overview/>

[8] Thi Anh Mai Phan 2013 Cloud Databases for Internet-of-Things Data Technical University of Denmark Informatics and Mathematical Modelling http://nordsecmob.aalto.fi/en/publications/-theses2013/thesis_mai/

Authors	
	<p>Aiman Moldagulova, 1961, Kazakhstan</p> <p>Current position, grades: Associate professor, Candidate of Physical and Mathematical Sciences University studies: Kazakh State University Scientific interest: Big Data, Data mining, Process Mining, Machine Learning Publications: more than 40 Experience: 32 years</p>
	<p>Azamat Zhubandykov, 1991, Almaty, Kazakhstan</p> <p>Current position, grades: web programmer in private company University studies: Master degree in IITU, Almaty Scientific interest: Analysis of big data Publications: 1</p>
	<p>Askar Mustafin, 1991, Almaty, Kazakhstan</p> <p>Current position, grades: IOS programmer in private company University studies: Master degree in IITU, Almaty. Scientific interest: Analysis of big data Publications: 1</p>