The use of NoSQL in business intelligence

Andrei Karpau
Institute of Technology Blanchardstown
Dublin

October 11, 2016

Abstract

The growth of the amount of global data results in new challenges that the technologies must cope with. This is also relevant for the business intelligence (BI) tools and methodologies, especially in scenarios of big and unstructured data. Distributed computing and NoSQL help to deal with these challenges. This paper looks into the main stages of the BI process, reviews the issues that occur in the context of big data and provides the overview of how the NoSQL technology can be used for solving them. Finally, it concludes with the latest achievements in this area and the problems that are not yet solved.

Keywords: business intelligence, NoSQL, data warehouse, ETL, OLAP

1 Introduction

Business intelligence focuses on the easy interpretation of the large volumes of data in order to allow the user to provide better business and strategic decisions. Duan and Xu (2012, p. 679) define it as the process of transforming raw data into useful information so that it yields business benefits. Jordan and Ellen (2009, p. 16) argue that BI provides the information for supporting business decision making, where the information results from the data supplemented by the correct business context. Actually, data gathering, preprocessing, storing and analyzing play the central role in the business intelligence process. The core components of traditional BI are: ETL tools, enterprise data warehouse (DW) and business analytics tools (Muntean and Surcel, 2013, p. 114).

Nowadays, there is a huge growth of the amount of the global data in the world. This increase consists of not only video and audio but also of emails, spreadsheets, financial and other business related data. Muntean and Surcel (2013, p. 114) indicate that enterprises have hundreds of internal and external data sources and as a result 80% of organizational data are unstructured or semi-structured. Since the traditional BI process works mainly with relational database management systems (RDBMS), the processing of the unstructured and big data can sometimes lead to difficulties. One of the possible solutions is the use of NoSQL databases. This technology is already used by many tech companies for processing, storing and analyzing big data. Quick reading and writing, support of mass storage, easy expandability and low cost are the main advantages of NoSQL systems (Han et al., 2011, p. 364). So, this paper aims to provide the literature review of the usage of NoSQL technology during the business Intelligence process and to examine the current state of knowledge on this subject.

1.1 Contribution and Structure

The paper addresses the use of NoSQL databases and distributed technologies in business intelligence. The paper researches the issues that occur during the main stages of BI process when dealing with big data and makes a review of existing solutions and methodologies that are based on NoSQL concept. The paper is structured as follows: in Section 2 there is the overview of NoSQL technology and the reasons of relevance; Section 3 contains the review of NoSQL usage during the BI process; and finally Section 4 provides the conclusions.

2 Survey on NoSQL

The continuous development of Internet and the need of storing and processing the big amounts of data detect the new challenges that the industry should take into consideration. Among them are high concurrency of reading and writing operations, the efficiency of saving big data, storage availability and scalability, limited capacity, slow reading and writing (Han et al., 2011, p. 363). Traditional SQL databases are not able to cope with these challenges because of its atomicity, consistency, isolation, and durability (ACID) properties. By contrast, NoSQL or Not only SQL generally provide BASE properties (Basically Available, Soft state, Eventually consistent). The idea is that by getting rid of ACID properties the higher performance and scalability can be achieved (Cattell, 2011, p. 12). Actually, NoSQL databases do not aim to eliminate the traditional RDBMS, but try to use both concepts in order to achieve high scalability, availability, performance and flexibility. Currently, there are many different NoSQL DBMS that vary by their data model. Han, Haihong and Le (2011, p. 364) highlight three mainstream data models: key-values stores, column-oriented stores and document stores. Key-values databases store the associative arrays that consist of key and corresponding value. The document DBMS use documents being encoded in a standard format such as JSON, YAML or XML. The column-oriented stores contain the key-value pairs that are saved in different columns of the table.

Generally, NoSQL systems provide the different approach than the traditional RDBMS do. Relational databases were developed to follow the ACID properties, which make it difficult to work with massive amounts of unstructured data. Contrastingly, NoSQL systems follow the BASE rules, which make it more flexible. Therefore, NoSQL databases is a more efficient choice for cases when it is required to store big and badly structured data, maintain high-speed operations on it, support flexible schemes and allow horizontal scalability.

3 NoSQL in business intelligence process

3.1 NoSQL in ETL

ETL is the first stage in BI process when the data from many sources is extracted, transformed and loaded to final target database that usually is data warehouse. When working with a lot of different data sources including the NoSQL databases extract and transform phases can cause difficulties. Vassiliadis and Simitsis (2009, p. 1100) state that with the evaluation of technology and broader use of Internet the interest moves from the traditional relational format of sources to the multiple types of data. Thus, modern ETL applications should be able to deal with XML, spatial, multimedia data and so on. The same relates to the NoSQL sources that more and more frequently are used for storing such information. As a result, solutions for data integration such as traditional ETL are no longer keeping pace with variety of sources and platforms that are often deployed in the cloud including Hadoop and NoSQL (McKendrick, 2014, p. 6).

During the last couple of years there is a tendency to integrate business intelligence platforms with NoSQL sources. Sometimes such solutions are cloud based and can work in distributed manner. For example, Microsoft provides Azure Data Factory Service that among all supports the extraction from Azure DocumentDB, Cassandra and MongoDB as well as transformations using Hive, Pig, MapReduce and Hadoop Streaming (Pelluru, 2016). Another ETL tool Oracle Data Integrator currently has the capabilities to connect to Hive, Sqoop and HBase systems (Oracle, 2016, pp. 7-8). The solutions for working with NoSQL databases are also provided by open source BI platforms. The most popular among them are Jaspersoft and Pentaho. They both have the integration with NoSQL databases such as MongoDB and Cassandra and can process big data using map reduce functions defined within Hadoop platform (Duda, 2012, pp. 33-34). The use of map reduce framework for data preprocessing provides the distributed computing capabilities and as a result an effective way for dealing with high workloads. Although, more and more ETL platforms become capable to work with NoSQL, most of the solutions cannot connect to the full range of existing NoSQL systems and often are coupled only to one vendor. A methodology proposed by Sahiet and Asanka (2015, p. 73) extracts the data through public APIs and then loads it into the staging area where the transformations and cleansing take place. After it, during the loading phase the transformed data is loaded into the data warehouse. This allows to reduce the processing in source NoSQL database, the performance of issues and the impact on DW as well as decouples the application from the concrete vendors. However, this methodology has not been released as an enterprise level application yet. Moreover, even though the methodology provides some flexibility in data preprocessing and platform choice, it still needs to deal with special API for each NoSQL data source on data extraction phase. Consequently, there are some ETL tools and concepts that provide the extraction of data from NoSQL databases and the transforming of it using the distributed frameworks. It leads to better performance and high capabilities when processing the big amount of unstructured data. Nevertheless, there is still the lack of tools that produce ETL operations with the full range of sources and NoSQL databases.

3.2 NoSQL in data warehousing

After the ETL process is done, the data is usually loaded into the data warehouse. DW is a database that is highly optimized for further processing and analysis that finally provides the information for business decisions. Such a processing is named online analytical processing (OLAP). Traditionally, data warehouses are implemented in the relational database management systems. That become an issue when dealing with the unusual volumes of data, which require distributed storage environment and the use of parallel treatment (Dehdouh, 2016, pp. 166-167). NoSQL databases fit good for this scenario. However, there is still no well-defined methodology for transforming multidimensional conceptual data model used in data warehouses into the model that is optimized for NoSQL databases.

The first step that is made by researchers is the creation of benchmark that is able to measure the performance of NoSQL data warehouse. Dehdouh, Boussaid and Bentayeb (2014b, p. 288) provide Columnar NoSQL Star Schema Benchmark that allows generating synthetic data and queries set to evaluate column-oriented NoSQL data warehouse. Another benchmark proposed by Chevalier, El Malki, Kopliku, Teste and Tournier (2015, p. 485) extends the simple Star Schema Benchmark to NoSQL databases. Thus, it allows the evaluation of data warehouses based on both relational and NoSQL databases. The next step is the creation of approach for migration between these types of databases. Yangui, Nabli and Gargouri (2016, p. 256) highlight two types of approaches: the migration from relational databases to NoSQL ones (indirect approaches) and the transformation of the multidimensial conceptual model to NoSQL logical one (direct approaches). The indirect approaches need all data to be stored in relational database at the beginning. Therefore, this limits the use of this concept on the big data sets. In contrast, according to the direct approaches data warehouse is built in NoSQL database. Dehdouh, Bentayeb, Boussaid and Kabachi (2015, pp. 469 & 474) propose the direct concept and provide three models named NLA (Normalized Logical Approach), DLA (Denormalized Logical Approach), and DLA-CF (Denormalized Logical Approach by using Column Family), which afterwards are evaluated by NoSQL Star Scheme Benchmark. These approaches differ in the structure and attributes used during the mapping. DLA and DLA-CF show better performance than NLA, however NLA uses less disk space. Yangui, Nabli and Gargouri (2016, p. 264) propose another direct approach that consist of transformation rules that ensure the translation from DW scheme to two logical NoSQL models (columns oriented and document ori-

ented). Finally, this concept has been tested on Cassandra (columns oriented) and MongoDB (document oriented). The experiments have shown that MongoDB with hierarchical transformation is more suitable for OLAP queries.
The analytical processing of the data stored in the DW is another important step in the BI process. Thus, some researches focus on the OLAP in NoSQL. Dehdouh, Bentayeb, Boussaid and Kabachi (2014a, pp. 3828 & 3830) argue that even though the data cubes computation is needed for representing decision making, column-oriented NoSQL DBMS do not have OLAP operators. The research provides new CN-CUBE operator for NoSQL databases that computes the OLAP cube in three phases: extraction of the attributes, hashing the dimension columns positions and the final aggregation. Dehdouh (2016, pp. 170-171, 175-177) describes the concept of MC-CUBE operator that is based on MapReduce paradigm. It performs the cube aggregation in four phases by executing seven MapReduce jobs. The final experiments have been made on Hadoop and Hbase environment on both single-node and multi-node clusters. The last one has shown the benefits due to storage and workloads distribution.

4 Conclusion

Nowadays, more and more data is produced in the world. As a result, companies have to cope with a lot of unstructured data and different data sources during the BI process. One of the most effective solutions for this problem in industry is the use of distributed computing and NoSQL databases. Actually, these technologies can be used in different stages of BI process as well. Modern ETL tools often have the possibility to collect data from different NoSQL databases. Some of them are able to use distributed computing technologies. However, most of the tools are not capable to work with the full variety of sources without the use of special drivers and components. Another important part of current investigations is the creation of data warehouses on top of NoSQL databases. The latest researches provide the concepts of benchmarks for measuring the performance of NoSQL data warehouses, transformation rules for converting the traditional DW scheme to NoSQL one and the conceptual models for building DW in NoSQL databases. When using them with the OLAP operators for NoSQL systems it can achieve the benefits especially in performance and storage. Although, there is still no system that aggregates the whole BI process and uses distributed technologies on each stage, much research has been done in this area and the latest achievements show that NoSQL technology together with distributed computing such as MapReduce provide the efficient solutions when dealing with big data in business intelligence context.

References

Cattell, R. (2011). Scalable sql and nosql data stores. *Acm Sigmod Record*, 39(4):12–27.

Chevalier, M., El Malki, M., Kopliku, A., Teste, O., and Tournier, R. (2015). Benchmark for olap on nosql technologies comparing nosql multidimensional data warehousing solutions. In *2015 IEEE 9th International Conference on Research Challenges in Information Science (RCIS)*, pages 480–485. IEEE.

Dehdouh, K. (2016). Building olap cubes from columnar nosql data warehouses. In *International Conference on Model and Data Engineering*, pages 166–179. Springer.

Dehdouh, K., Bentayeb, F., Boussaid, O., and Kabachi, N. (2014a). Columnar nosql cube: Agregation operator for columnar nosql data warehouse. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3828–3833. IEEE.

Dehdouh, K., Bentayeb, F., Boussaid, O., and Kabachi, N. (2015). Using the column oriented nosql model for implementing big data warehouses. In *Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*, pages 469–475. The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).

Dehdouh, K., Boussaid, O., and Bentayeb, F. (2014b). Columnar nosql star schema benchmark. In *International Conference on Model and Data Engineering*, pages 281–288. Springer.

Duan, L. and Da Xu, L. (2012). Business intelligence for enterprise systems: a survey. *IEEE Transactions on Industrial Informatics*, 8(3):679–687.

Duda, J. (2012). Business intelligence and nosql databases. *Information Systems in Management*, 1(1):25–37.

Han, J., Haihong, E., Le, G., and Du, J. (2011). Survey on nosql database. In *Pervasive computing and applications (ICPCA), 2011 6th international conference on*, pages 363–366. IEEE.

Jordan, J. and Ellen, C. (2009). Business need, data and business intelligence. *Journal of Digital Asset Management*, 5(1):10–20.

McKendrick, J. (2014). Data integration in the era of big data. *Database Trends and Applications*, 28(1):4–9.

Muntean, M. and Surcel, T. (2013). Agile bi-the future of bi. *Informatica Economica*, 17(3):114–124.

Oracle (2016). White paper: Oracle data integrator 12c new features.

Pelluru, B. S. (2016). Introduction to azure data factory service, a data integration service in the cloud. Retrieved Oktober 4 2016, from https://azure.microsoft.com/en-us/documentation/articles/data-factory-introduction/.

Sahiet, D. and Asanka, P. D. (2015). Etl framework design for nosql databases in data warehousing. *International Journal of Research in Computer Applications and Robotics*, 3(11):67–75.

Vassiliadis, P. and Simitsis, A. (2009). Extraction, transformation, and loading. In *Encyclopedia of Database Systems*, pages 1095–1101. Springer.

Yangui, R., Nabli, A., and Gargouri, F. (2016). Automatic transformation of data warehouse schema to nosql data base: Comparative study. *Procedia Computer Science*, 96:255–264.