An Exploratory Study of a NoSQL Database for a Clinical Data Repository

Francini Hak, Tiago Guimarães, António Abelha, and Manuel Santos(✉)

Algoritmi Research Center, Universidade do Minho, Braga, Portugal
a76665@alunos.uminho.pt, {tsg,mfs}@dsi.uminho.pt, abelha@di.uminho.pt

Abstract. The need to implement a distributed Clinical Data Repository (CDR) at a healthcare facility, rose in large part due to the high volume of data and the discrepancy of their sources. Over the years, Relational Database Management Systems (RDBMS) began to present difficulties in responding to the needs of various organizations when it comes to manipulating a large amount of data and to its scalability. Therefore, it was necessary to explore other techniques to choose the appropriate technology to build the CDR. In this way, NoSQL emerged as a new type of database that is quite useful to work with multiple and different types of data. In addition, NoSQL introduces a number of user-friendly features such as a distributed, scalable, elastic and also fault tolerant system. In this way, Oracle NoSQL Database was the NoSQL solution chosen to develop this case study, using the key-value storage. This article was motivated to propose a CDR architecture based on Oracle NoSQL Database functionalities. A one-single node database was deployed for better comprehension, in order to enhance their features for future implementation.

Keywords: Clinical Data Repository · Key-value · NoSQL · Oracle NoSQL

1 Introduction

Since 1970, Relational Database Management Systems (RDBMS) has been the dominant model for database management. It has been used in most applications to store and retrieve data. However, new applications have been requiring a fast and large amount of data storage due to the advancement of the Internet and the emergence of distributed computing [1]. Thus, a new type of database called NoSQL has emerged to try to meet the new challenges.

NoSQL appeared when organizations realized that RDBMs systems had a fault in terms of scalability, i.e., the adaptability of this system to the growth of resources and users. RDBMS adopt "scaling up" techniques, vertical scalability, focusing only on increasing capabilities on a single machine, such as memory or CPU. Instead, NoSQL databases adopt "scaling out" methods, horizontal scalability, focusing on increasing the number of machines for better performance.

This new data storage technique has led to many properties and processes of a traditional database system undergoing a process of change. For example, transactional

Á. Rocha et al. (Eds.): WorldCIST 2020, AISC 1161, pp. 476–483, 2020.
https://doi.org/10.1007/978-3-030-45697-9_46

properties required to an RDBMS as ACID process (Atomization, Consistency, Isolation and Durability) are not applied to NoSQL systems due to its strong consistency and reliability, resulting from the application of a new model called BASE (Basic Availability, Soft state, Eventual consistency).

Therefore, the main factors that led to the emergence of NoSQL databases were the strictness of relational databases and, consequently, the inadequacy to store a large amount of data [2].

Based on previous studies, a need was identified to create a new Clinical Data Repository that should be able to manage a large amount of data from heterogeneous sources, derived from existing hospital models of information and clinical knowledge. In this sense, this article was motivated to propose an architecture for the new CDR based on Oracle's NoSQL solution, using key-value storage.

This paper is structured in 6 sections. The first and current section exposes the purpose and objectives to be achieved with this document. The second section describes the main concepts involved in this article. The third highlight the methodology and tools used. The proposed architecture is presented on fourth section. The section number five discusses the work developed. Finally, in the last section, final considerations are presented on the study developed so far.

2 Background

2.1 NoSQL Database

NoSQL stands for "Not only SQL" because it represents, more accurately, an approach that combines non-relational databases with the use of relational databases [3]. NoSQL belongs to a group of non-relational data management systems which becomes very useful when it is required to work with a large amount of data or when that same data does not need a relational model and does not need to follow a fixed structure [4].

Sharding technique and horizontal scaling are two characteristics that can be found in NoSQL. The sharding technique consists in the process of storing data on multiple machines which becomes crucial whereas, with the growth of the data, one machine may not be enough for storage or may not correspond to the performance expected [5]. About horizontal scaling, that corresponds to the addition of more machines or setting up a cluster for the software system, which will allow the splitting of data through the sharding technique [6].

Elasticity is also a core feature of NoSQL databases. In times of overload, elasticity is characterized by the adaptation of the system in such situations. It is developed in a horizontal scalability environment and aims to manage and allocate available system resources to balance them when access distribution is exceeded. It is important to note that it has a number of user-friendly features, ranging from being a distributed, scalable and elastic, to its fault-tolerance system, which can be concluded to be of great benefit to the user [2].

In addition, NoSQL databases are divided into four data model types, including [7]:

- Document Databases - stores data in document structure and encodes the information in formats such as JSON. Ex.: MongoDB, CouchDB.

- Graph Databases - emphasizes connections between data, storing related nodes as objects and relationships as edges in graphs to speed up the query. Ex.: Neo4j, GraphDB.
- Key-value Databases - uses a simple data model that matches a single key to a value in data storage. Ex.: Redis, Dynamo, Oracle NoSQL DB.
- Wide Column Stores - column families oriented, is also called Table-Style Databases and stores data through tables that can have a large number of columns. Ex.: Cassandra, HBase.

2.2 Oracle NoSQL Database

Oracle's NoSQL open-source solution was chosen due to key-value approach and for partnership reasons. Oracle NoSQL Database provides data manipulation with some traditional particularities from the NoSQL definition, such as scalability, non-relational database, and elastic storage. It is also characterized by providing flexible schemas, fast load and sharding and replication techniques, using the key-value storage [8].

First release from Oracle NoSQL Database came out in 2011, being its current release from 2019. The database requires a Linux operating system and can be accessed by REST API methods. Oracle NoSQL Database provides Community and Enterprise Edition with a set of characteristics that set them apart. Another feature is the KVLite version which allows simplified database deployment on just one Node.

In this way, the Oracle's solution offers a three-tier architecture (Fig. 1) of presentation, logic and database. The first one, represents the client interface, that is, the output of a request made by the user. The second, is composed by the Oracle NoSQL Driver that consumes Oracle Berkeley Database library [9], which is responsible for controlling functionalities required by the executed process and for the data distribution. The last one, is the database, that stores using the key-value schema.

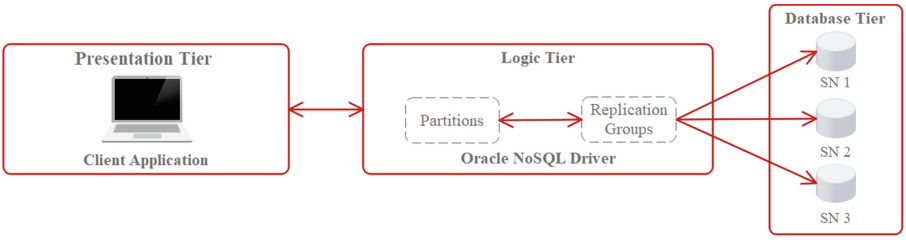

Fig. 1. Three-tier architecture

Following a key-value store, data is hashed in partitions through the primary key, building replication groups, also called shards. Consequently, distributed data are stored in Storage Nodes (SN), that represent a physical machine with its own memory, storage and IP address. Each SN contains Replica Nodes (R) that perform writing and reading functions [10]. Thus, as the number of SN's increases, the better the system will perform.

2.3 Clinical Data Repository

Health care delivery encompasses a complex procedure that involves many different professionals, resources and functions. This results in a large amount of diverse and scattered data. Thus, the concept of Clinical Data Repository (CDR) grows as the need for common sharing storage does. It was firstly described as a "clinical research database that provides a detailed, flexible and quick view of clinical data" in 1995 at University of Virginia [11].

Recently, Gartner [12] described the CDR as a "aggregation of patient-centered granular health data, generally collected from heterogeneous systems and intended to support multiple functions". In this way, the data undergoes a specific organization by analysis, being the CDR understood, in such a way, as Clinical Data Warehouse (CDW).

However, a CDR aims to be distributed and to integrate clinical support rules in order to acquire clinical knowledge, as the opposite of a CDW [13]. In this case, decision support mechanisms are represented by a controlled medical vocabulary and by clinical guidelines previously applied. CDR also promotes internal interoperability and incorporates an architecture the same reasoning and mechanisms as a CDW but focuses mostly on gaining knowledge from clinical data.

3 Methods and Tools

The learning process in each domain is applied through a methodological approach. Therefore, the implementation of the Design Science Research (DSR) encompasses the scope of scientific research in the field of information systems. Collins, Joseph, and Bielaczc [14] state that the insertion of scientific research in the area of information systems becomes crucial due to the need to address theoretical issues and the study of practical cases already carried out in the field.

At the technological level, the study included the practice and theory of using Oracle's NoSQL tool, being its exploration one of the proposed objectives. The use of Oracle's NoSQL solution was crucial to propose a Clinical Data Repository architecture. Thus, this tool has characteristics of a NoSQL database such as horizontal scalability, elasticity and distributed storage, focusing on key-value schema [8].

4 Proposed Architecture

Nowadays, data production and the need to achieve results have been grow exponentially worldwide, especially in healthcare. This advancement has been extremely remarkable in recent years, resulting in improved patient care and a more robust decision-making support.

Afterwards clinical data registration by a health professional, each record is stored in a different source, depending on its context and type. According to adopted approaches in previous studies, the open data model allows to combine knowledge with clinical information, following guidelines and clinical coded terms in a structured way.

This required in-depth research on how to manipulate this data produced by the healthcare organization and to turn it into intelligent and optimized solutions. As it is exposed in Fig. 2, the new CDR requires a solution that can support large amounts of data and flexibility, providing decision support.

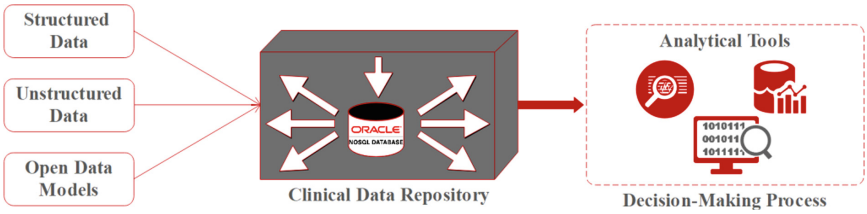

Fig. 2. Process overview

Therefore, the proposed architecture of the Clinical Data Repository is based on Oracle NoSQL Database (Fig. 3), that aims to store and manage data in different structures, combining relational and nonrelational databases. This new technique aims to combat inefficiencies that relational databases brought to these services [15].

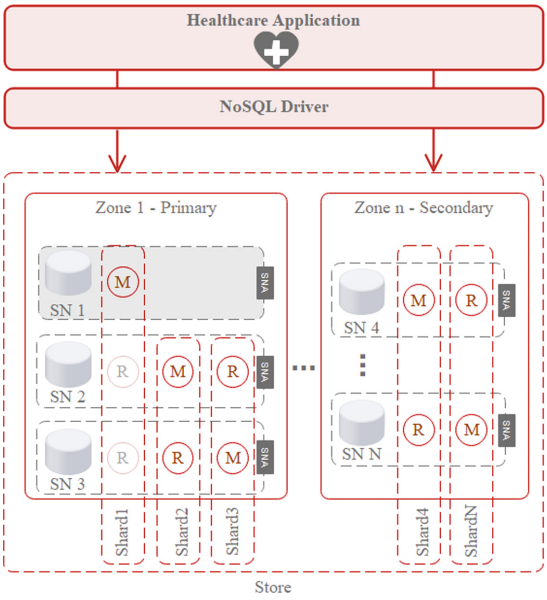

Fig. 3. Clinical Data Repository architecture

Through the exploration of Oracle NoSQL Database some important concepts for its correct performance have been identified. Data storage represented by the last tier of the architecture, is divided in Zones that correspond to physical locations according to

the capacity of the system, could be primary or secondary type. Storage Nodes (SN) are represented within a Zone, corresponding to machines that perform both data writing and reading functions [9].

Increasing the number of SN in the system enables better performance and will decrease system storage latency, as it is formalized in horizontal scalability. In addition, for effective communication between the SN's it is necessary to activate the responsible agent for this function, the Storage Node Agent (SNA), as well as verify its correct operation [8].

According to the proposed architecture based on *N* Storage Nodes and *X* Zones, for this case study only one SN in a Zone was deployed, represented in grey by SN1 in Fig. 3. For a correct activity of SN1, some configuration parameters were established such as the IP address of the respective machine and the communication ports, as well as the system capacity and administrative security system.

Regarding the distribution of data in the cluster, this is done by the Sharding technique that distributes the data uniformly by the Shards in a set of Partitions. This is a fundamental NoSQL method that aims to easily organize and distribute data between machines through the primary key of each record according to key-value, in order to not overload the system.

For effective understanding of the data, each Shard makes up a group of Replication Nodes (R) that perform read functions, the Master Nodes (M) being responsible for writing. The master node always has the most up-to-date value for a given key as opposite of read Replicas that can have slightly older versions [9]. Accordingly, the set of Replication Nodes is called the Replication Factor (RF). In the case applied of implementing one-single node, the formula used to calculate the number of partitions required was as follows [8]:

$$Partitions = 10 \times \frac{Capacity}{Replication\,Factor} = 10 \times \frac{1}{1} = 10$$

According to the topology implemented, the RF number is equal to 1 as well as the system's capacity. Hence, it is possible to state that Shard1 has one Master Node and 10 Partitions enabled. Studying all of these concepts of Oracle NoSQL was crucial to deploy the SN in the proposed architecture, desiring increases the number of nodes in the future to face the requirements for the Clinical Data Repository.

5 Discussion

This article was aimed to explore a solution to propose an architecture for the new Clinical Data Repository. It must be qualified in volume, velocity, scalability and elasticity, that matches with NoSQL concepts. Thus, the Oracle NoSQL Database was the chosen technology for the proposed architecture, with a one-single node deployment.

Furthermore, one of the main features that sparked interest in the Oracle's NoSQL database was the key-value storage. Being the simplest type of NoSQL data models, the key-value is based on array and also comparable to dictionary and hash functions,

mapping a key to a value. A key is a unique identifier and a value the data identified, that is a string of bytes in arbitrary length.

The data model is characterized for being schema free due to the fact that each record can have its own structure as opposed to relational models, giving flexibility to the database. Hereupon, key-value pairs are located in a Distributed Hash Table (DHT), allowing a node to effectively access a value through a key filling up scalable resources [6].

As mentioned before, data distribution is performed by Shards containing a hashed set of records or partitions, stored based on the primary key. Both the key and the value are application-defined, given some loose restrictions according to the NoSQL Driver [9]. In this way, the records inserted in the store are uniformly organized in key-value pairs in partitions.

In Oracle NoSQL, data is stored in particular shards depending on the hashed value of the primary key of the table. Thus, the key or primary key are a combination of major and minor key that the major component identifies the partition which contains a record and what shard is stored, so all of the records with the same major key will be co-located on the same server [10].

With all of this data storage and management mechanism, it is important that the database is configured to track desired performance. This requires that the records do not focus on the same major key, otherwise the system will suffer performance issues as the data is entered.

6 Conclusions and Future Work

The need to explore new solutions capable to support large amounts of heterogeneous data led to the characterization of the NoSQL concept. NoSQL and Big Data concepts are also directly linked when it comes to large amounts of data. In this way, NoSQL meets the requirements proposed by Big Data characteristics such as Volume, Variety and Velocity, the 3Vs that characterize Big Data.

Thereupon, that represents the capacity to handle a large amount of data of various types with different structures, generating and querying data quickly in the store. In this way, the article was developed to address the lack of scalability and speed of a relational database system, leading to the exploration of the NoSQL concept as one of the requirements imposed for the work developed.

As a result, the Oracle NoSQL database was the chosen technology for in-depth study to its functions and data manipulation with key-value store. The proposed architecture for the Clinical Data Repository (CDR) comprehends that structure of the technology.

The study concluded that Oracle's NoSQL tool has adequate functionality for the required implementation, particularly in resource allocation and easier troubleshooting. The key-value data schema is also attractive for future implementation as it has simple and efficient management of data manipulation. Although there are some restrictions on its tool installation, Oracle NoSQL brings high expectations for the implementation of the new Clinical Data Repository.

Future work focuses on building an Oracle NoSQL Database application for the CDR in a multi-node deployment for better system performance. This targets to a deepening of clinical knowledge, improving of care service and supporting the decision-making processes. Business Intelligence techniques for NoSQL database will also be explored as focal points for future work.

Acknowledgments. The work has been supported by FCT – Fundação para a Ciência e Tecnologia within the Project Scope UID/CEC/00319/2019 and DSAIPA/DS/0084/2018.

References

1. Shertil, M., Jowan, S., Swese, R., Aldabrzi, A.: Traditional RDBMS to NoSQL database: new era of databases for big data. J. Humanit. Appl. Sci. **29**, 83–102 (2016)
2. Costa, C., Santos, M.Y.: Big Data: state-of-the-art concepts, techniques, technologies, modeling approaches and research challenges. IAENG Int. J. Comput. Sci. **43**(3), 285–301 (2017)
3. Madison, M., Barnhill, M., Napier, C., Godin, J.: NoSQL database technologies. J. Int. Technol. Inf. Manag. **24**(1), 1–14 (2015)
4. Moniruzzaman, A.B.M., Hossain, S.A.: NoSQL database: new era of databases for big data analytics - classification, characteristics and comparison. Int. J. Database Theor. Appl. **216** (2895), 43–45 (2013)
5. Anand, V., Rao, C.M.: MongoDB and Oracle NoSQL: a technical critique for design decisions. In: Proceedings of the International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS 2016) (2016)
6. Abramova, V., Bernardino, J., Furtado, P.: Experimental evaluation of NoSQL databases. Int. J. Database Manag. Syst. **6**(3), 01–16 (2014)
7. Han, J., Haihong, E., Le, G., Du, J.: Survey on NoSQL database. In: 2011 6th International Conference on Pervasive Computing and Applications, pp. 363–366. IEEE (2011)
8. Oracle: Oracle NoSQL Database: Fast, Reliable, Predictable, pp. 1–38, November 2018
9. Oracle: Oracle® NoSQL Database: Concepts Manual, April 2018
10. Oracle: Oracle® NoSQL Database: Getting Started with Oracle NoSQL Database Key/Value API, August 2019
11. Einbinder, J.S., Scully, K.W., Pates, R.D., Schubart, J.R., Reynolds, R.E.: Case study: a data warehouse for an academic medical center. J. Heal. Inf. Manag. **15**(2), 165–175 (2001)
12. Gartner: Information Technology: Clinical Data Repository (2018)
13. Hamoud, A.K., Hashim, A.S., Awadh, W.A.: Clinical data warehouse: a review. Iraqi J. Comput. Inform. **44**(2), 1–11 (2018)
14. Collins, A., Joseph, D., Bielaczc, K.: Design research: theoretical and methodological issues. Am. Heal. Drug Benefits **3**(3), 171–178 (2004)
15. Kunda, D., Phiri, H.: A comparative study of NoSQL and relational database. Zambia ICT J. **1**(1), 1 (2017)