EFFECTIVENESS OF SELECTED DATABASE MODELS IN MANAGEMENT INFORMATION SYSTEMS

Edyta SZCZEPANIUK[1*], Hubert SZCZEPANIUK[2]

[1] Polish Air Force Academy, National Security and Logistics Faculty; e.szczepaniuk@law.mil.pl, ORCID: 0000-0002-6707-2987
[2] Warsaw University of Life Sciences, Faculty of Economic Sciences, Warsaw; University of Information Technology, Faculty of Information Technology, Warsaw; hubert_szczepaniuk@sggw.pl, ORCID: 0000-0001-7613-4281
* Correspondence author

Abstract: In this paper, the authors attempt to identify the universal factors of the effectiveness of selected database models used in Management Information Systems. Selection of an appropriate database model at the stage of design and implementation directly determines the effectiveness of the Management Information Systems within an organisation. The relational database model and database models based on NoSQL paradigms, including in particular graph databases and document-oriented databases, will be evaluated. NoSQL-class database models demonstrate properties in terms of effective application for business purposes.

Keywords: effectiveness of databases, Management Information Systems, NoSQL databases.

1. Introduction

The architecture of Management Information Systems (MIS) implemented in organisations with a view to improving the information and management processes usually involves, at the architectural level, an adequate data model. An effective management process requires efficient acquisition and processing of information resources coming from the organisation and its environment. Information needs of today's enterprises are often fulfilled under the conditions of processing heterogeneous datasets, the expanding Big Data phenomenon and the development of the process management concept. The entirety of the indicated processes poses new requirements on the contemporary Database Management Systems (DBMS).

The relational data model is a popular technology used in the implementation of business solutions. However, there are alternative data models, called NoSQL-class databases, which include graph databases, key-value databases and document-oriented databases. In a DB-Engines January 2018 ranking of the most popular database systems, the first four places were occupied by relational database models (from first to fourth: Oracle, MySQL, Microsoft SQL

Server and PostgreSQL), the fifth place belonged to the document-oriented MongoDB, places six and seven were again taken by a relational data model (DB2 and Access, respectively), Cassandra took the eighth place with its column data model, while the ninth place was taken by Redis as a key-value type database (DB-Engines, 10 January 2018).

Selection of the database system architecture directly determines the structure of the database model, the process of implementing a database access layer and the effectiveness of the entire system in terms of performing access and manipulation operations on the database. Given the availability of a variety of architectural solutions and the fundamental significance of an adequate data model in information systems implemented in organisations, the need for research on the effectiveness of applying a variety of databases for business solutions is justified.

2. Methodology

The subject of this research is the data models applied in information systems implemented in organisations. The aim of the research is to identify the fundamental factors of efficiency of data models applied in information systems which are implemented in modern organisations under the conditions of processing heterogeneous datasets. The effectiveness of a database model will be understood in this study in terms of the adequacy of the database model to the expected information needs of the organisation.

The adopted methodology involves an interdisciplinary research approach in the field of information technology and management, including a theoretical analysis and synthesis of the subject literature and the use of elements of programming, system analysis and mathematical modelling.

3. Databases in Management Information Systems

One of the main objectives of MIS is to streamline the information processes within an organisation. The thusly defined objective requires an appropriate database model and database management system (DBMS) to be applied. The market of database solutions is mainly dominated by relational databases. However, other database systems which can form the basis of MIS are also available on the market. These include first and foremost the NoSQL-class databases. It should be noted that the leading NoSQL-class solutions are mostly key-value and graph databases.

The theoretical formal model of relational databases was designed by E.F. Codd, who had based his considerations on the mathematical concept of relation calculated from an extended Cartesian product (Codd, 1972). Codd assumed that the extended Cartesian product x for n D_1, D_2, ..., D_n sets and the r relation determined from its subset have the following form (Codd, 1972, p. 5):

$$x\ (D_1, D_2, \dots, D_n) = \{(d_1, d_2, \dots, d_n): d_j \in D_j, dla\ j = 1,2, \dots, n\}, \quad (1)$$

r is the relation on sets D_1, D_2,...D_n if it is a subset $x(D_1, D_2, \dots, D_n)$.

In the relational model proposed by E.F. Codd, the database is a set of individual relations. The theoretical formal model of relational databases is conducive to maintaining the consistency of data stored in the database.

In practice, relational database modelling involves determining the scheme of tables consisting of columns of a specific value type out of the set of types supported by the Database Management System. Tables are a representation of the individual relationship in the database. Data is stored in the rows of individual tables (tuples, records). Each table has a specific name and a primary key, defined on the non-empty set of columns, which uniquely identifies each record in the table. Tables included in the database may be subsequently connected to the relationships of the following types: one – one (1 – 1), one – many (1 – n), many – many (m – n).

Currently, relational database management systems are one of the most popular groups of the solutions used in business applications. The set of such solutions may equally include the commercial software, namely, Oracle Database, Microsoft SQL Server and the software based on the open-source licenses, such as MariaDB.

Another type of the analysed databases is the graph database, based on the mathematical graph structure. Graph is one of the elementary abstractions in IT and is widely used within this field. Databases are one the effective applications of the concept of graphs. Graph representation in the database is based in particular on the use of directed multi-graphs and refers to the concept of nodes and edges in the graph. Attributes may be assigned both to the nodes and edges of the graph. Model of a directed multi-graph has been shown in Figure 1.

For the multi-graph shown in Figure 1, it is possible to define a set N of the graph nodes, $N = \{1,2,3,4\}$ a set of binary relation R defined on the N set, as a set of graph edges $R = \{(1,2), (1,3), (2,4), (4,3), (4,4)\}$ and a set of attributes assigned to the graph edges $A = \{e_1, e_2, \dots, e_5\}$.

Representation of the database in the graph-based concept may use the nodes of graph as data entities and the edges of graph as relations. Such assumption enables data storing in the graph representation. It is possible to effectively perform database operations by using mathematical algorithms for processing such graphs.

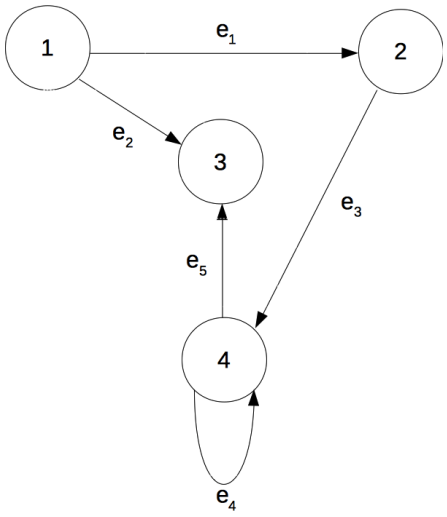

Figure 1. Directed multi-graph with attributes.

Neo4J is one of the leading software in the field of the graph database management systems, which is currently available on the market and used in the information systems of business solutions. It should be noted, that it is difficult to identify a uniform standard of queries in the case of graph database, unlike the relational data model in which there is a standard database queries in the form of SQL query language and its extensions. This means that there exists a variety of standards for the query language, depending on the specific implementation of the graph database management systems. Providing an example, the graph database management system Neo4J uses the Cypher Query Language (CQL).

Syntax of an exemplary query for a graph database in the Cypher Query Language, which searches reports, made by Jan Kowalski since 2018, may be presented as follows:

```
MATCH (employ: Employee {name: "Jan Kowalski"}) - [: AUTHOR] -> (report: Report)
WHERE report.year <= 2018
RETURN report
```

Figure 2. The query structure for the graph database in the Cypher language.

CQL language operates mainly on clauses such as MATCH, WHERE and RETURN, which refer to the vertices, edges and attributes pf the directed multi-graph, representing the database. On the other hand, the SQL language operates on the SELECT, FROM, WHERE, GROUP BY, HAVING and JOIN clauses in terms of access to the data relational model.

The advantage of graph databases are relationships expressed in the form of a set of attributes of the directed edges of the graph. Description of a relationship in a directed multi-graph allows the provision of an additional layer of semantics for the data representation and facilitates the process of machine learning, e.g., algorithms of the artificial intelligence or algorithms of the data-based knowledge extraction.

Another kind of the analysed databases is the document-oriented solution, which assumes that the data is stored in the form of documents identified by keys. Documents describing the data may have any specific structure. The database may include many documents of the

differently specified structures. Such approach creates opportunities to adjust the data model to the information needs of an organisation at various stages of the system implementation in a flexible and appropriate manner.

Data storing documents may have a variety of formats, depending on the specific implementations of the Database Management System. XML and JSON are the most popular standards for recording documents. Exemplary data document encoded in the XML standard has been shown in figure 3.

```
<person>
        <name>Jan</name>
        <surname>Kowalski</surname>
        <city>Warszawa</city>
</person>
```

Figure 3. Data document encoded in XML.

On the other hand, the analogous data document encoded in the JSON standard may have the form as shown in figure 4.

```
{
        „name”: „Jan”,
        „surname”: „Kowalski”,
        „city”: „Warszawa”
}
```

Figure 4. Data document encoded in the JSON standard.

Encoding data documents in XML and JSON indicates that the document-oriented databases facilitate the integration with modern programming technologies, e.g., in terms of eliminating the need for mapping objects to the relational data model.

According to the DB-Engines ranking, the most popular document-oriented Database Management System is MongoDB (DB-Engines, 10.01.2018). MongoDB is a management system of databases, of which the data document structure is formatted using the JSON standard.

Graph technologies of databases are successfully used in situations where there is a need for processing the dynamically changing, heterogeneous datasets. According to the official server of MongoDB database, it was used, inter alia, by New York Times (MongoDB, 01.10.2018).

4. Discussion

Information systems of modern organisations operate in the environment of large, structurally diverse and changeable datasets. At the same time the information systems must provide adequate information and adequate data model allowing the accomplishment of

business processes of the organisation. Despite the growing requirements regarding the information systems being implemented in contemporary organisations, majority of them still use the architecture of relational database management systems. It is advisable to discuss the individual dimensions of the database management systems, oriented on the relational and the alternative NoSQl model, in terms of efficiency factors against business applications.

Table 1.
Comparison of the relational model with the NoSQL model

Feature	Relational model	NoSQL model
Formal model	One formal model based on assumptions of the relation assigned on the expanded Cartesian product of sets.	Many formal models include, inter alia, graph databases, key-value databases and document-oriented databases.
Data model	Database contains a finite set of tables consisting of tuples. The following is a set of primary and foreign keys and the relations are defined between tables (1-1, 1- n , m - n).	Depending on the formal model, database may contain associative tables (key-value database type); implementation of the graph model; a set of documents, e.g., in the JSON or XML standard for the document-oriented database.
Data schema	Data static schema modelled at an early phase of the designed information system.	Dynamic data schema, easier editing of the data model at the later stages of the life cycle of the information system.
Query language	The generally accepted standard of the query structured SQL and its PL / SQL and Transact-SQL extensions.	A variety of the database query languages depending on the specific implementation of the data management system.

Note: based on (NoSQL Databases Explained, 01.10.2018).

Both the concept of formal model and the data model is critical to the efficiency of the entire MIS in an organisation. Analysing the data models shown in Table 1 it is possible to distinguish the following set of efficiency factors, relevant when choosing a database management system for a particular MIS project:

- adjusting the data model – the chosen architecture should allow the correct mapping a set of business processes in the organization in the data model,
- data schema flexibility – ability to expand the data schema with new structures when the organisation is being developed or the concept of business processes is being changed,
- database performance – access and data manipulation operations should be realised within a fixed time horizon,
- safety – the database system should comply with the assumptions set out in the safety information policy of the organisation,
- ability to integrate – integration ease with other information systems in the organisation and technologies used in the implementation,
- database management system purchase costs – the market offers commercial systems, both payable and free-of-charge, which are distributed on open licences,

When considering the adjustment and flexibility of the data model, it is necessary to take into account the possibility of mapping and supporting business processes in a given organisation. Relational data model may prove to be a good choice for organisations which process the structured datasets and a small number of business processes that are repetitive and well-characterised. Database which is based on the NoSQL assumptions may be a better choice for organizations which process large, heterogeneous datasets of a dynamic and changing structure.

Results of an experiment comparing the efficiency of a relational database (RBD) and the graph database (GBD) for the specified data model have been described in the article titled „Neo Technology Commercializes Next Generation Graph Based Database" (Cubrilovic, 27.09.2018). In the quoted experiment the database performance was evaluated by the defined query response time through the database engine. Both databases reflected the structure of one thousand people, with each person having fifty friends assigned (Cubrilovic, 27.09.2018). A query was determined both for RBD and GBD, which were to return the data of each the friend from a defined set of people. Query response time amounted to 2000 ms for the RDB and 2 ms for GBD, implanted in the Neo4j system (Cubrilovic, 09.27.2018). In the next step of the experiment, a set of people was increased up to one million and a number of connections was increased by an order of magnitude, whilst the query execution time for the graph database was not changed (Cubrilovic, 09.27.2018).

Query response time is of fundamental importance for end-users of the MIS implemented in the organisation. By way of illustration, the time to generate reports concerning employees of various organisation departments and the company's management depends on query execution time to the database. Therefore, the right selection of the database system translates directly into the efficiency of information processes in the organisation.

Object-relational mapping, referred to as ORM, is a frequent problem encountered in the field of integration capabilities of relational database management systems with modern programming technologies. Modern programming languages such as Java or C# are object-oriented, whereas relational databases store data in the form of a finite set of tables. In such a situation it may be necessary to translate the data from tabular form into objects in a particular programming language, and vice versa. Although modern database management systems are a formalised, stable and safe technology, translation of the object-oriented application logic into the relational database may be problematic and time-consuming. It should be taken into account that the object-oriented application logic, e.g., in the Java programming language may additionally implement mechanisms that support inheritance, polymorphism, interfaces and collections. Range of technologies, e.g., Java language may use the Hibernate technology was created to solve the problem of object-relational mapping. The ORM mapping issue may be reduced by using the NoSQL databases, especially the document-oriented ones, which may map the specific objects in the data documents corresponding to the logic of the object-oriented program.

5. Summary

Management Information Systems at the level of the database management subsystem architecture may be based on a variety of database systems. Selection of an appropriate database system directly determines the effectiveness of the Management Information Systems within an organisation.

Relational databases are the most common class of database systems in the business solutions. However, not always, the choice of the relational data model at the MIS design and implementation phase is an optimal solution. Modern economic organisations operate in conditions of processing the large, rapidly changing and diverse datasets. It is possible to use the databases based on NoSQL paradigms to handle the processing of such sets. Graph database should be distinguished in this respect, because thanks to mathematical searching operations of the graph they provide highly efficient algorithms of access and data manipulation. Document-oriented database is another database model, which shows high potential for businesses. A key advantage of such a database model is a dynamic structure of the data schema, allowing for flexible processing of heterogeneous and changeable datasets. In addition, the document-oriented databases offer API for modern programming languages eliminating the need for the object-relational mapping.

According to the authors, the efficiency factors, which should be considered when choosing a database model at the phase of designing and implementing MIS include: data model adjustment, data schema flexibility, database performance, safety, integration capability and the purchase costs of the Database Management System (DBMS).

References

1. Codd, E.F. (1972). *Relational Completeness of Data Base Sublanguages.* San Jose: IBM Research Laboratory. Retrived from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.90.5965&rep=rep1&type=pdf.
2. DB-Engines (2018.01.10). Available online https://db-engines.com/en/ranking.
3. MongoDB (2018.01.10). Available online https://www.mongodb.com/post/38162455327/new-york-times-runs-mongodb?c=f80679ca74.
4. Mugglestone, P. (2018.01.10). *Graph Processing with SAP HANA 2*. Available online https://blogs.sap.com/2016/12/01/graph-processing-with-sap-hana-2/.
5. What is NoSQL? (2018.01.10). Available online https://www.mongodb.com/nosql-explained.

6. Wrembel, R. (2018.01.10). *Relacyjny model danych*. Available online http://wazniak.mimuw.edu.pl/images/0/04/BD-2st-1.2-w02.tresc-1.1.pdf.
7. Zakrzewski, P. (2018.01.10). *Grafowe bazy danych*. Available online https://prezi.com/38oxxwd59yei/grafowe-bazy-danych/.