

V. V. Hnatushenko¹,
orcid.org/0000-0003-3140-3788,
Vik. V. Hnatushenko²,
orcid.org/0000-0001-5304-4144,
N. L. Dorosh²,
orcid.org/0000-0003-4184-3648,
N. O. Solodka³,
orcid.org/0000-0002-7545-4969,
O. A. Liashenko³,
orcid.org/0000-0002-9983-5504

1 – Dnipro University of Technology, Dnipro, Ukraine,
e-mail: vygnat@ukr.net

2 – Ukrainian State University of Science and Technology,
Dnipro, Ukraine

3 – Ukrainian State University of Chemical Technology, Dni-
pro, Ukraine

NON-RELATIONAL APPROACH TO DEVELOPING KNOWLEDGE BASES OF EXPERT SYSTEM PROTOTYPE

Purpose. Use of a non-relational database management system is proposed while developing a database of a prototype of expert system with using a semantic model of the knowledge.

Methodology. The study compares traditional relational approach with the proposed non-relational one in terms of the formation of certain queries. The following indices are used to compare efficiency of two management systems for the databases: particular query set (in MySQL and Cypher languages); runtime for the specified record size (i.e. their processing speed); ease of understanding; and software support of the queries.

Findings. It has been identified that the graph model is a more expedient solution in the process of designing semantic networks and their development where complex hierarchical relationships between objects have to be stored and processed. Architecture of the graph database has been applied in terms of the specific example. A prototype of an expert system has been developed to demonstrate the capabilities of the created system of logical inference. The classifier of sciences was chosen as an example in the subject area.

Originality. A prototype of the expert system, using the proposed non-relational approach, has been designed involving modern service-oriented architecture (SOA). The abovementioned helped separate the database from the inference engine and the user interface, facilitate perception as well as update and code debugging. Service-oriented architecture makes the system more flexible and robust.

Practical value. The developed software is meant to develop both simple expert systems and medium-complex ones.

Keywords: *semantic network, database, graph model, Neo4j, SOA*

Introduction. In the modern information society, there is an increasing need for the use of the latest technologies in various fields to improve data processing and storage methods which are increasingly unstructured, poorly structured, or require processing of complex hierarchical relationships between objects. The latter is relevant for the development of expert systems. This feature of the data causes a number of problems for the application of traditional approaches based on the relational data model to the construction of data warehouses that are integrated into software systems. Traditional relational databases lose performance when storing such data.

Expert systems are applicable to solve problems involving practices of professionals. Intellectuality of expert systems is a potential to function with the formalized specialist knowledge in the context of so-called database. Information acquisition from the analysis of processes of problems, solved by specialists, and their formalization is the model of the expert behavior in the context of the specific field of knowledge. Inference procedures and decision procedures are applied for the problem-solving within the expert systems.

Usually, the structure of expert systems consists of the following:

- subsystem of knowledge acquisition being responsible for its accumulation, transfer, and transformation from the expert information;
- database containing the knowledge used to formulate rules;
- interference machine consisting of procedures and algorithms able to make conclusions;
- user interface for user-computer communication;
- explanation subsystem substantiating the decision made by the system.

Literature review. The problem topicality is substantiated by the reviews [1] describing the current research and developments in the field of expert systems. Nowadays, the expert systems are used by business [2], medicine [3], agriculture [4], education [5], defense industry [6] and by other engineering applications and by many other spheres.

Development process of the expert system applies data and information accumulated by the knowledge base. Its implementation depends upon the selected classical model of knowledge representation: semantic, logical, rule-oriented, frame-based model. Moreover, fuzzy-logic expert systems are

also widely used [3]. Use of apparatus of artificial neural networks is the specific tendency in the field of expert system development [7, 8].

Semantic network, being the most visual and practical one for software implementation, is the most popular model to represent knowledge in the context of expert systems.

There are many technologies to represent semantic data [5, 9, 10], but they cannot take into consideration database representation in the form of graphs. Graphs are applied productively in terms of image classification problems in the process of imaging and pattern recognition [11–13].

As a rule, graph algorithm is implemented in the intellectual analysis [14, 15]. Graph capability to store data as well as their relations is also used in algorithms of process detection [16, 17]. The listed scientific sources demonstrate the great interest of many researchers in the use of graph models in terms of various problems. Graph databases are satisfiable for the link analysis; hence, they are of utmost interest to be used for the development of expert systems in terms of which graph data structure is quite natural.

Purpose. Subject of the research is a process of inference within the expert system on the basis of a semantic network within the use of a graph data model.

The research objectives are as follows:

- to carry out comparative analysis of the efficiency of a traditional relational database and a graph database in terms of the developed test case;
- to apply the graph database for the elaboration of interference mechanism in the context of expert systems;
- to select a strategy for the system architectural design;
- to develop a prototype of expert system.

Proposed approach. Comparison of a relational database and a graph database. The expert system being elaborated has been developed with the use of a four-stage semantic model of knowledge representation involving a methodology of the software lifecycle: selection of a model for data storage while database developing; design of the expert system; prototype implementation; and the system test.

Traditionally, relational data models dominate among the means of data storage. Eventually, developers of software applications faced problems, to solve which the relational databases are not sufficient as for the query speed and ease of display of the data structure. Moreover, from time to time, software implementation of the queries in the SQL language turns out to be a problem. For instance, if it is required to implement multilevel database connections within a relational database, it will be a multilevel chain of multiaspect requests of JOIN-based connections.

A relational base stores data in columns and lines. Nevertheless, the columns and lines differ from the data available in the real world. More like, the data exist as objects as well as connections between them. Right within EC, connections between the data may be more valuable than the data themselves. Relational model turns out to be not flexible enough to develop complex queries, and to process various hierarchical and multilevel connections between the objects.

So-called Not only SQL (NoSQL) data storages have been developed as an alternative to the relational bases. One of the branches of such databases is a graph one where data are stored in a form of a graph rather than in a tabular form. Nodes and connections are the basic components of the graph model. The graph bases are more preferable while data handling.

The matter is that just connections between objects are quite important especially when it is required to backtrace the connections several levels deep. At the same time, relational databases calculate connections during querying through the complicated JOIN operations, graph data base save the connections together with data in the model.

The listed advantages and the current tendencies to solve certain problems on the basis of information saving in the form

graphs as well as the problems of optimum processing of the relations and their connections stimulated to select the graph model of data storage in the process of expert system development for which such a graph data structure is natural. Selection of the graph databases for semantic network implementation within expert system may simplify algorithm complexity owing to the proper architecture.

A number of papers concern review, analysis, relational databases and nonrelational databases comparison as well as examples of their usage [18, 19]. The research stood clear of design and development of expert systems with the use of semantic networks as the model of knowledge representation.

In terms of formulation of one and the same queries with the use of different data models, the following problem becomes topical: whether a graph model is more expedient solution while designing and developing semantic networks where complex hierarchical connections between objects should be stored and processed.

Neo4j graph database, developed on the basis of a high-level object-oriented Java language, has been used to implement a semantic model. Currently, Neo4j is the most popular graph databases with proper processing capabilities: own query language and data manipulation (Cypher); and own data storage format adapted specially for graph information representation. To compare with the relational DBMS, such an approach makes it possible to apply extra optimization while handling data being of a more complex structure.

Neo4j architecture has been designed to optimize such processes as management, storage, and bypassing nodes and connections. In the context of Neo4j, relations are important objects being the connections between essences. The relational model is not flexible enough to create complex queries as well as handle various hierarchical and multilevel relationships between objects. It is known that in the relational database, JOIN productivity decreases exponentially with the increased relation number. Instead, in the Neo4j the operation is performed as a linear navigation algorithm from one node to another.

The two DBMSs were tested using a PC with the following characteristics: Intel(R) Core(TM) i5-3210M CPU @ 2.50 GHz processor, 64-bit operating system, x64 processor, 6 Gbyte RAM.

Comparison with the help of MySQL involved making a TEST_TABLE and completing it. Identical data were applied to complete a base developed in Neo4j.

The following indices were used to compare efficiency of the two DBMSs: certain query set (in the MySQL and Cypher languages), their processing speed, ease of understanding, and ease of software implementation of the queries. Among other things, two queries were considered: a root node search (Q1 query) and multilevel node sampling (Q2 query). The test results have helped identify the time to execute two types of queries in terms of 1000 recordings (nodes). Tables 1 and 2 demonstrate test results for the queries.

Table 1

Time to execute Q1 query

Database	Query Structure	Query execution time (msec)
MySQL 5.6(x64)	SELECT out_node FROM `TEST_TABLE` WHERE out_node NOT IN (SELECT in_node FROM `TEST_TABLE`);	0.2
Neo4j 3.2.3	MATCH (n) WHERE NOT (n)<-[](-) RETURN n	3.0

Table 2

Time to execute Q2 query

Database	Query Structure	Query execution time (msec)
MySQL 5.6(×64)	<pre>SELECT out_node, in_node FROM (SELECT * FROM TEST_TABLE ORDER BY out_node, in_node) nodes_sorted, (SELECT @pv := '1') initialisation WHERE find_in_set(out_node, @pv) AND length(@pv := concat(@pv, ',', in_node)) AND in_node != 1000;</pre>	1.2
Neo4j 3.2.3	<pre>MATCH result = ({ value:0 })-[*]- >({ value:1000 }) RETURN result</pre>	3.0

The presented data indicate that there is not always an undeniable advantage in speed between relational and non-relational databases. Examples of the formation of certain queries using two different data models have shown that the databases managed to get efficiently through the test queries in terms of the specified data amount. In the context of the amount of data being tested, processing speed turned out to be comparable for the two databases. The considered number of recordings (nodes) cannot achieve the data amount where advantages of graph databases become obvious. However, choosing a graphical storage model to implement a semantic network in an expert system can simplify the complexity of algorithms due to its architecture which is designed to optimize processes such as management, storage as well as bypassing nodes and connections.

Service-oriented structure in the process of an expert system prototype development. There are two basic approaches to develop complex information systems: monolithic architecture and service-oriented architecture (SOA). For a long time, so-called “monolithic architecture” dominated. In terms of the approach, the system is a monolith located physically in terms of one machine, which is initiated in one process, and performs all business operations of the system.

To compare with the monolithic system, capable of horizontal scale only, a system with service-oriented architecture is based upon the use of the distributed, loosely connected replaced components equipped by the standardized interfaces to interact according to the standardized protocols. Hence, it consists of a set of independent services focused on the proper task. If the components implement their specific function, then it means they can be applied in the context of other systems.

In the strict sense, it is the requirement for successful expert systems to separate a database from a logic inference machine as well as from a user interface. The abovementioned makes it possible to supplement the database by the facts and rules from various resources with no changes for users.

As it is known, a modular principle is applied to develop expert systems, i.e. database and inference mechanism are different components. SOA solves connectivity problem at the level of the software component through the standardized protocols and interfaces. Moreover, if the system components are independent in fact, then their boundaries are well defined indeed (as it in the structure of expert system. Hence, SOA use is especially efficient.

Owing to the component distribution between different server nodes as well as to independent interaction language, it becomes possible to use absolutely alternative programming languages, interaction tools, and data storage. That makes it possible to select the best and the most convenient solutions and to make experiments, e.g. with new methods of data storage.

Among other things, if there is a solution to transfer data from a graph database to relational or document-oriented, or to a hybrid database, then nonavailability of SOA approach would stipulate the necessity for the third-party software programmers to develop a code querying directly the database which would destruct the databases component. SOA use helps databases setup for the interface takes place independently. Thus, transfer from one databases type to another interface can change and destruct nothing.

Obviously, SOA makes application more robust: even if one of the components fails, it will not result in the crash of the whole application. For instance, a server with admin panel may shutdown; nevertheless, database server and a server of a user application will operate as usual.

When SOA is applied, the program is the distributed system messaging according to a specific protocol. Currently, REST is a silent standard in the process of SOA development applied by the paper.

REST advantages:

- possibility to call remote function as well as possibility to link information to the service (as distinct from RPC);
- transfer of data as well as their state while calling the service (as distinct from SOAP);
- possibility to use JSON format of data coding (as distinct from SOAP applying XML).

Consequently, REST approach advantage in the process of SOA development is obvious.

The following tools were used while developing the expert system: REST API (for the system component communication); databases Neo4j (for the database storage); CodeIgniter framework (for the admin panel); and Knockout framework (for a User Interface component).

Figs. 1 and 2 represent structures of a monolithic architecture, and SOA architecture. Use of SOA (Fig. 2) makes it possible to modify every of the three parts, shown in Fig. 1, with no interface changes. Fig. 2 demonstrates a partial case of SOA architecture used to develop an inference machine while creating the expert system.

Hence, service-oriented architecture of the developed product helps facilitate understanding, debugging, and code modifying if appropriate. Moreover, it is of the necessary flexibility required by expert systems.

Results. Development of an expert system prototype helps avoid numerous reworks while creating a run-time version of the expert system and use a potential of a preliminary comprehensive testing. Test data of the science classifier have been applied as an example. A prototype of expert system has been developed as a result of the research. The prototype supports expediency and efficiency of the applied SOA concept according to which the program has individual components – administrative component and user one. An administrator may

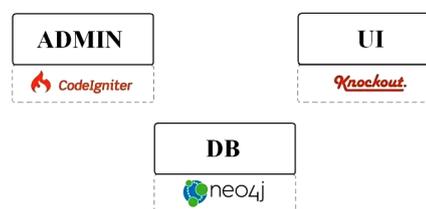


Fig. 1. Common monolithic architecture

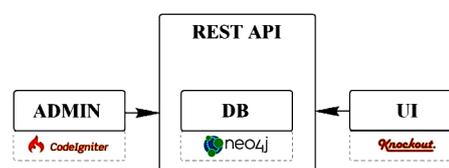


Fig. 2. SOA architecture

create statements to form database of the expert system prototype (Fig. 3).

Implementation of the program provides the opportunity for a user to obtain a name of the specific knowledge domain from a more abstract level. The data are arranged in the order

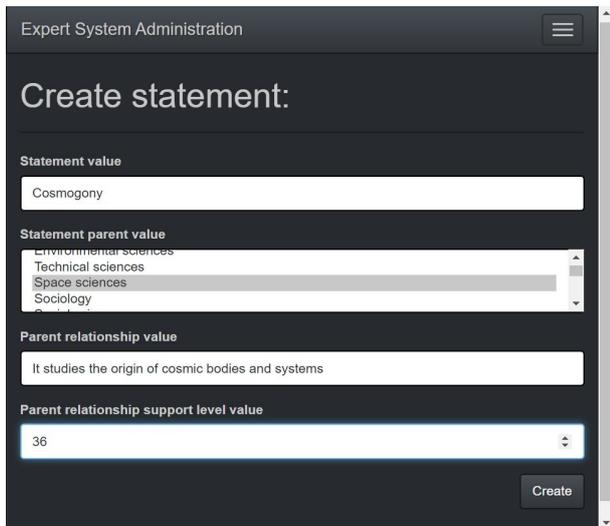


Fig. 3. Administrator interface

of choice to a more special area. The search results in certain scientific discipline.

As a result of the introduction of statements within the administrator panel, relevant data of the science classifier are generated automatically in the graph databases Neo4j (Fig. 4). Fig. 5 demonstrates a user interface panel.

While selecting one of the proposed possible answers, a user obtains a new corresponding set of statements specifying previous ones. During the search stage, preliminary query results are stored in the history being available to the user. Moreover, along with the results selected previously, support level value is also displayed which can be interpreted as a reliability level of the obtained results.

Conclusions. The paper considered a possibility to apply graph database for the development of knowledge base of expert system.

The developed test examples have been applied to compare a traditional database of a relational type with a graph database. The test examples have helped determine that the both databases managed to get efficiently through the test queries in terms of the specified data amount. Search time in terms of a relational database and a graph database differ nominally. In the context of the amount of data being tested, processing speed turned out to be comparable for the two databases.

Non-relational graph DBMS (database managing systems) should be used for processing and storing large

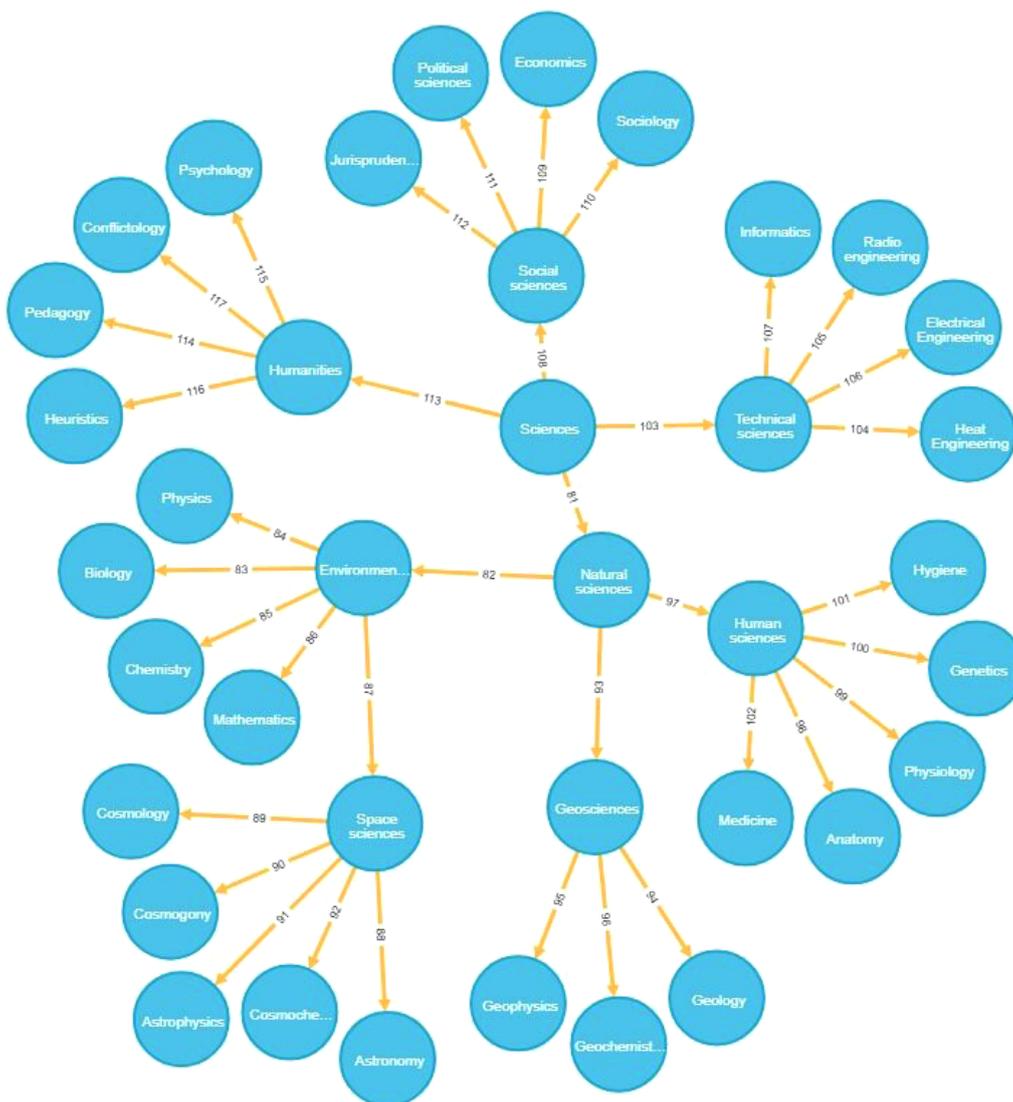
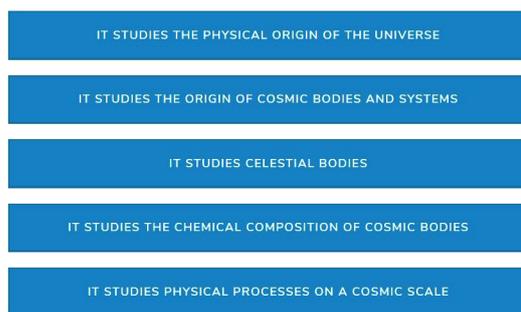


Fig. 4. Graphic presentation of data in terms of Neo4j DBMS

Choose one of the proposed options

SPACE SCIENCES



History:

SL - Support level

- Sciences
- They study natural systems their processes and phenomena, mean Natural sciences
SL - 25
- They learn the form and motion of matter, mean Environmental sciences
SL - 5
- They study Space Development, mean Space sciences
SL - 92

Fig. 5. User interface panel

amounts of data with a flexible structure where data is closely connected in relationships that can go deeper into several levels, if hard disk resources are less important and the search should be performed quickly. For small systems that have a clear fixed data structure, it is better to use a relational database.

The key features and advantages of the use of service-oriented architecture instead of the classic monolithic solution in the process of the expert system development have been considered.

As a result, an expert system prototype has been developed with the use of the graph database and service-oriented architecture. In the long view, the developed software is meant to develop both simple expert systems and medium-complex ones. Moreover, it demonstrates a potential to have a functional of an expert system shell.

References.

1. Deepthi, M. B., & Sreekantha, D. K. (2017). Application of expert systems for agricultural crop disease diagnoses – A review. *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*, 222-229. <https://doi.org/10.1109/ICICCT.2017.7975192>.
2. Yang, Y., Fu, C., Chen, Y.-W., Xu, D.-L., & Yang, S.-L. (2016). A belief rule based expert system for predicting consumer preference in new product development. *Knowledge-Based Systems*, 94, 105-113. <https://doi.org/10.1016/j.knosys.2015.11.012>.
3. Arani, L. A., Sadoughi, F., & Langarizadeh, M. (2019). An expert system to diagnose Pneumonia Using Fuzzy Logic. *Acta Informatica Medica*, 27, 103-107. <https://doi.org/10.5455/aim.2019.27.103-107>.
4. Lasso, E., & Corrales, J. C. (2016). Expert system for crop disease based on graph pattern matching: a proposal. *Revista Ingenierías Universidad de Medellín*, 15(29), 81-98. <https://doi.org/10.22395/rium.v15n29a5>.
5. Yunianta, A., Yusof, N., Bramantoro, A., Haviluddin, H., Othman, M., & Dengen, N. (2016). Data mapping process to handle semantic data problem on student grading system. *International Journal of Advances in Intelligent Informatics*, 2(3), 157-166. <https://doi.org/10.26555/ijain.v2i3.84>.
6. Azimirad, E., & Haddadnia, J. (2016). A new model for threat assessment in data fusion based on fuzzy evidence theory. *International Journal of Advances in Intelligent Informatics*, 2(2), 54-64. <https://doi.org/10.26555/ijain.v2i2.56>.
7. Batista, L., Silva, G., Araujo, V., Rezende, T., Guimarães, A., Campos Souza, P., & Araujo, V. (2018). Fuzzy neural networks to create an expert system for detecting attacks by SQL Injection. *The International Journal of Forensic Computer Science*, 13(1), 8-21. <https://doi.org/10.5769/IJ201801001>.
8. Hnatushenko, V., & Zhernovyi, V. (2019). Complex Approach of High-Resolution Multispectral Data Engineering for Deep Neural Network Processing. *Lecture Notes in Computational Intelligence and Decision Making, ISDMCI 2019*, 1020, 659-672. https://doi.org/10.1007/978-3-030-26474-1_46.
9. Yuninta, A., Barukab, O. M., Yusof, N., Dengen, N., Haviluddin, H., & Othman, M. S. (2017). Semantic data mapping technology to solve semantic data problem on heterogeneity aspect. *International Journal of Advances in Intelligent Informatics*, 3(3), 161-172. <https://doi.org/10.26555/ijain.v3i3.131>.
10. Hnatushenko, V., Zhernovyi, V., Udovik, I., & Shevtsova, O. (2021). Intelligent System for Building Separation on a Semantically Segmented Map. *2nd International Workshop on Intelligent Information Technologies and Systems of Information Security (IntellITSIS-2021)*, 2853. Retrieved from <http://ceur-ws.org/Vol-2853/keynote1.pdf>.
11. Chen, Z., Jiang, B., Tang, J., & Luo, B. (2017). Image Set Representation and Classification with Attributed Covariate-Relation Graph Model and Graph Sparse Representation Classification. *Neurocomputing*, 226, 262-268. <https://doi.org/10.1016/j.neucom.2016.12.004>.
12. Yuan, H., Li, J., Lai, L. L., & Tang, Y. Y. (2018). Graph-based multiple rank regression for image classification. *Neurocomputing*, 315, 394-404. <https://doi.org/10.1016/j.neucom.2018.07.032>.
13. Zhu, R., Dornaika, F., & Ruichek, Y. (2019). Joint graph based embedding and feature weighting for image classification. *Pattern Recognit*, 93, 458-469. <https://doi.org/10.1016/j.patcog.2019.05.004>.
14. Velampalli, S., & Jonnalagedda, M. V. (2017). Graph based knowledge discovery using MapReduce and SUBDUE algorithm. *Data Knowl. Eng.*, 111, 103-113. <https://doi.org/10.1016/j.datak.2017.08.001>.
15. Zhang, Q., Song, X., Yang, Y., Ma, H., & Shibusaki, R. (2019). Visual graph mining for graph matching. *Computer Vision and Image Understanding*, 178, 16-29. <https://doi.org/10.1016/j.cviu.2018.11.002>.
16. Sarno, R., Sungkono, K. R., Johanes, R., & Sunaryono, D. (2019). Graph-based algorithms for discovering a process model containing invisible tasks. *Intelligent Networks and Systems Society*, 12(2), 85-94. <https://doi.org/10.22266/ijies2019.0430.09>.
17. Sarno, R., & Sungkono, K. R. (2019). A survey of graph-based algorithms for discovering business processes. *International Journal of Advances in Intelligent Informatics*, 5(2), 137-149. <https://doi.org/10.26555/ijain.v5i2.296>.
18. Liashenko, O., & Dorosh, N. (2021). Technologies Of Software Development Based on Non-Relative Databases. *Scientific and Technical International Conference: Information Technology in Metallurgy and Machine Engineering*, 334-337. <https://doi.org/10.34185/1991-7848.itmm.2021.01.041>.
19. Paul, S., Mitra, A., & Koner, C. (2019). A Review on Graph Database and its representation. *2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC)*, 1-5. <https://doi.org/10.1109/ICRAECC43874.2019.8995006>.

Нереляційний підхід при розробці бази знань прототипу експертної системи

В. В. Гнатушенко¹, Вік. В. Гнатушенко²,
Н. Л. Дорош², Н. О. Солодка³, О. А. Ляшенко³

- 1 – Національний технічний університет «Дніпровська політехніка», м. Дніпро, Україна, e-mail: vvgnat@ukr.net
- 2 – Український державний університет науки та технологій, м. Дніпро, Україна
- 3 – Український державний хіміко-технологічний університет, м. Дніпро, Україна

Мета. Застосування нереляційної системи управління базами даних при розробці бази знань прототипу експертної системи з використанням семантичної моделі представлення знань.

Методика. Проведено порівняння традиційного реляційного підходу й запропонованого нереляційного на прикладах формування певних запитів. Для порівняння ефективності двох систем управління базами даних використані наступні показники: певний набір запитів (мовами MySQL і Cypher), час виконання для заданого розміру записів (швидкість їх обробки), простота розуміння та програмної реалізації запитів.

Результати. З'ясовано, що графова модель є більш доцільним рішенням при проектуванні та створенні семантичних мереж, де потрібно зберігати та обробляти складні ієрархічні зв'язки між об'єктами. На конкретному прикладі застосована архітектура графової бази даних. Для демонстрації можливостей створеної системи логі-

чного висновку розроблено прототип експертної системи. В якості предметної області для прикладу було обрано класифікатор наук.

Наукова новизна. Прототип експертної системи, що використовує запропонований нереляційний підхід, спроектований із застосуванням сучасної сервісно-орієнтованої архітектури (Service-oriented architecture, SOA). Це дозволило відокремити базу знань від машини логічного виводу та інтерфейсу користувача, полегшити сприйняття, зміну й налагодження коду. Сервісно-орієнтована архітектура робить систему більш гнучкою та стійкою до збоїв.

Практична значимість. Розроблене програмне забезпечення призначене для створення простих і середніх за складністю експертних систем.

Ключові слова: *семантична мережа, база даних, графова модель, Neo4j, SOA*

The manuscript was submitted 23.06.21.