UDC 004. 822:658.512.2 **doi:** https://doi.org/10.20998/2522-9052.2025.4.03

Pavlo Pustovoitov[1], Volodymyr Kompaniiets[1], Yiyong Liu[1], Galina Sokol[1], Mohamed Awad-Alla[2]

[1] National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine
[2] Modern University for Technology and Information, Cairo, Egypt

DATA MODEL OF COMPONENTS OF COMPLEX TECHNICAL SYSTEMS BASED ON SEMANTIC NETWORKS

Abstract. The article addresses the problem of ontological modeling of technical objects characterized by complex multi-level structure, diverse functional properties, and significant volume of accompanying documentation. A method for representing the subject domain in the form of an ontology that integrates data about objects, their constituent elements, and connections with technical instructions and drawings is proposed. Based on the developed ontological scheme, a graph database has been built that provides efficient search, storage, and processing of knowledge, as well as multi-user access support considering user rights. To formalize the model properties, the mathematical apparatus of graph theory and ontological relations has been applied, which allowed describing the patterns of system construction and functioning. The work performs a comparative analysis of advantages and disadvantages of existing approaches to knowledge organization, determines the place and novelty of the proposed model in the context of modern research. Experimental results confirmed the advantages of using graph databases for working with technical information, particularly increased performance in executing search queries and reduced costs for data updates. The practical significance of the research lies in the possibility of using the developed model for automated support of the lifecycle of complex technical systems, including stages of operation, maintenance, and modernization. The obtained results create a foundation for further implementation of intelligent knowledge processing technologies in the field of engineering and technical operation.

Keywords: ontological modeling; graph databases; technical objects; graph theory; knowledge management; technical documentation; life cycle of systems.

Introduction

Relevance of the Research. Modern engineering complexes, ranging from industrial equipment to defense-related systems, comprise a large number of heterogeneous components with diverse nature and functional roles [1–4]. The efficient operation and maintenance of such systems require the integrated management of technical drawings, manuals, specifications, and operational data. While traditional relational databases ensure reliable storage of structured information, they remain limited in their ability to represent complex multi-layered interrelations between components [5–7].

A promising direction for overcoming these limitations lies in the application of semantic networks and ontology-based models for database design. Such an approach enables the formalization of knowledge about objects and their interdependencies, while simultaneously supporting advanced information retrieval, including the processing of queries expressed in natural language [8, 9].

Literature Review. Existing studies in the field of engineering databases are mainly focused on relational or object-relational models, where relationships between entities are expressed through tables and foreign keys. This approach is effective for structured data but is poorly suited for handling drawings or semi-structured documents.

In the contemporary literature on ontological modeling and semantic networks, there is a significant expansion of methodologies and applied approaches; however, most of them emphasize aspects different from the database model proposed in this article for integrating drawings and manuals of technical systems. Works [10, 11] provide a detailed foundation for ontology construction and the use of OWL/RDFS, which forms an important basis for our approach, but they are mainly oriented toward the general principles of the Semantic Web and do not consider the specifics of technical documentation. The study [12] focuses on the creation of enterprise knowledge in the form of graphs and is advantageous in terms of scalability, but it does not sufficiently formalize the relationships between visual and textual components, which is addressed in our model. Research [13, 14] concentrates on natural language ontology and the semantics of expressions, with a strong linguistic component, but their application in technical databases is limited due to the lack of mechanisms for engineering drawings. Publications [15, 16] demonstrate the integration of open data methods and linked knowledge structures, which is promising for organizing semantic archives but less suitable for private engineering collections due to confidentiality concerns. Works [17, 18] highlight the development of neurosymbolic approaches and their integration with large language models, which creates new opportunities for automation; however, their drawback lies in high computational costs and the absence of formal guarantees of correctness, in contrast to our structured logical–mathematical system. Finally, book [19] critically evaluates the potential of artificial intelligence without offering engineering solutions, and thus only indirectly relates to our subject matter. Therefore, the existing studies provide a methodological foundation and demonstrate a wide range of approaches, but the semantic model we have developed combines logical rigor with a practical focus on technical documentation, which compensates for the noted limitations of previous works. Considerable attention has recently been devoted to graph databases (Neo4j, OrientDB, ArangoDB), which naturally represent complex relationships between objects in the form of graphs [20]. In parallel, ontological approaches (OWL, RDF) are being developed, enabling the formalization of domains and logical inference based on predicates [21].

There are also attempts to integrate intelligent systems into technical documentation, such as the recognition of drawings using convolutional neural networks, semantic annotation of documents, and the use of SPARQL queries for RDF repositories [22]. However, comprehensive solutions that simultaneously integrate graphical, textual, and structured data into a unified system are lacking, which determines the scientific novelty of this work.

Formulation of the Research Aim and Objectives. The aim of this work is to develop and justify an ontological model for representing technical objects in the form of a graph database, which ensures effective storage, processing, and analysis of knowledge about objects, their components, and related documentation.

To achieve this aim, the following research objectives were set:

1) to analyze modern approaches to ontological modeling of technical systems and identify their advantages and disadvantages;

2) to design an ontological scheme of the subject domain with the definition of the main classes and the relationships between them;

3) to develop the structure of a graph database for storing instances of objects, their components, and their connections with technical documentation;

4) to construct formal mathematical derivations describing the properties of the proposed model and to study their impact on the efficiency of knowledge representation;

5) to carry out an experimental study of the performance of the developed model in comparison with traditional relational approaches;

6) to evaluate the practical applicability of the model using the subject domain of automotive systems as an example and to formulate recommendations for its application.

Formalization of the Mathematical Model

The proposed system for storing technical documentation of technical components can be represented as a semantic network, which is formalized as a directed graph [23, 24]:

$$G = (V, E, \varphi),$$

where V is the set of vertices corresponding to the objects of the subject domain (car, units, individual components, drawings, and textual instructions), $E \subseteq V \times V$ is the set of directed edges describing the relationships between the vertices (for example, "part of," "documents," "version"), and $\varphi:\ E \to R$ is the mapping of edges to the set of relations R. Thus, for each pair of objects $a, b \in V$, if there exists a connection $e = (a,b) \in E$ between them, the corresponding relation $\varphi(e) \in R$ is defined. As an example, if the Car_A has the engine $Engine_B$, then

$$Car_{A,} Engine_B \in E, \varphi(Car_{A,} Engine_B) = has\_part. \quad (1)$$

The ontological interpretation of the model is based on the use of triples of the form (Fig. 1).

$$O = \{(s, p, o) s, o \in V, p \in R\}, \quad (2)$$

where s is the subject, p is the predicate, and o is the object.

In this representation, the entire knowledge base is formalized as a set of RDF-like triples, for example:

$$(Engine_{B,} documented\, by, Drawing_{D1}), \quad (3)$$

which is interpreted as "the engine *B* is documented by the drawing *D*1."

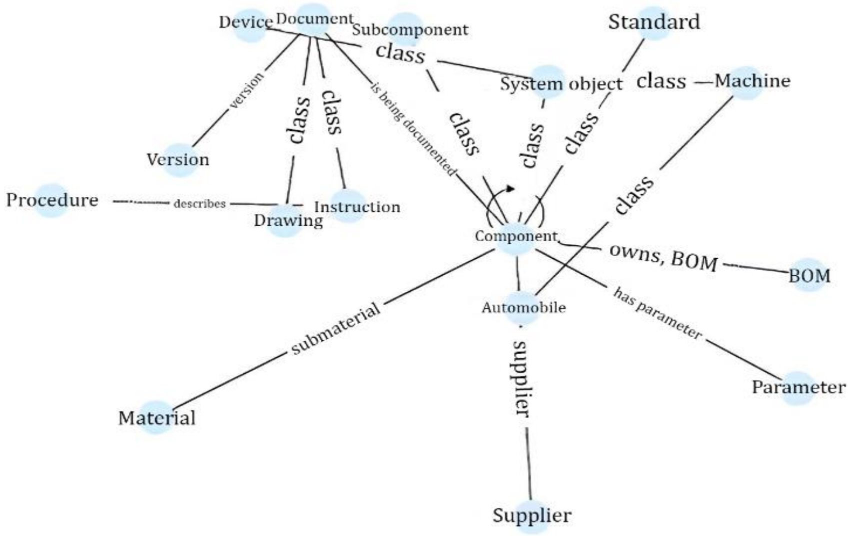

Fig. 1. Ontology schema (classes and relationships)

In turn, user queries to such a system can conveniently be translated into expressions of first-order predicate logic [25, 26]. For example, the query "Find the transmission drawing for the Audi A6 car" formally takes the form:

$$\begin{aligned} &\exists_x \exists_y (Component(x) \wedge BelongsT_0(x, Car_{AudiA6}) \wedge \\ &\wedge\ HasType(x, Transmission) \wedge \\ &\wedge\ DocumentedBy(x, y) \wedge Drawing(y)), \end{aligned} \quad (4)$$

where the variable x corresponds to a car component and the variable y to the corresponding drawing. Such a formal representation makes it possible to interpret queries as patterns of graph substructures and to implement them in languages such as SPARQL [27] or Cypher [28]. To determine the similarity between objects in the semantic network, the concept of semantic distance is introduced. Let the vertices be $v_i, v_j \in V$. Then the distance between them is defined as:

$$d(v_i, v_j) = \min_{p \in P(v_i, v_j)} \sum_{e \in p} w(e), \tag{5}$$

where $P(v_i, v_j)$ is the set of all possible paths between the vertices v_i and v_j, and $w(e)$ is the weight of the edge ee, which characterizes the strength or significance of the corresponding relation. For example, the relation “is_a” (class inclusion) may have a weight of $w = 1$, while the relation “compatible_with” (compatibility) may have a weight of $w = 3$, reflecting a greater “distance” in the semantic space.

Since the system must support intelligent search for drawings and manuals, the relevance of a document to a given query is formalized through a matching function. Let the query be described by a set of key predicates

$$Q = \{q_1, q_2, ..., q_k\}. \tag{6}$$

Then the relevance of a document d is defined as:

$$\mathrm{Re}l(d, Q) = \frac{1}{k} \sum_{i=1}^{k} sim(d, q_i), \tag{7}$$

where $sim(d, q_i)$ is the function measuring the correspondence of document d to a specific predicate q_i.

For textual manuals, this function can be implemented using information retrieval metrics such as TF-IDF [29, 30] or BM25 [29–31], whereas for graph objects it can be realized through checking the matching of relations in the graph or by applying subgraph isomorphism algorithms. Thus, the formalization of the database model for automotive systems based on semantic networks combines the graph interpretation of objects, the ontological representation in the form of triples, the use of predicate logic for queries, semantic distance metrics for object comparison, and the relevance function for evaluating documents with respect to user queries. This creates a theoretically grounded foundation for building practical information systems for integrated storage of drawings and manuals in the field of automotive engineering.

Experimental Modeling

To illustrate the operation of the semantic database model, let us consider the Audi A6 automobile. In the graph database, a vertex is created to represent the car, along with a set of vertices corresponding to its components, documents, parameters, and additional characteristics (Fig. 2):

$$\begin{gathered} V = \{Car\_AudiA6, Engine\_2.0TFSI, Transmission, Suspension\ LH, ShockAbsorber, Drawing\ ENG, \\ Instruction\ ENG, Version\ v2, Parameter\ Power, Material\ Steel, Supplier\ Bosch, Standart\ Euro6, BOM\ AudiA6\}, \end{gathered} \tag{8}$$

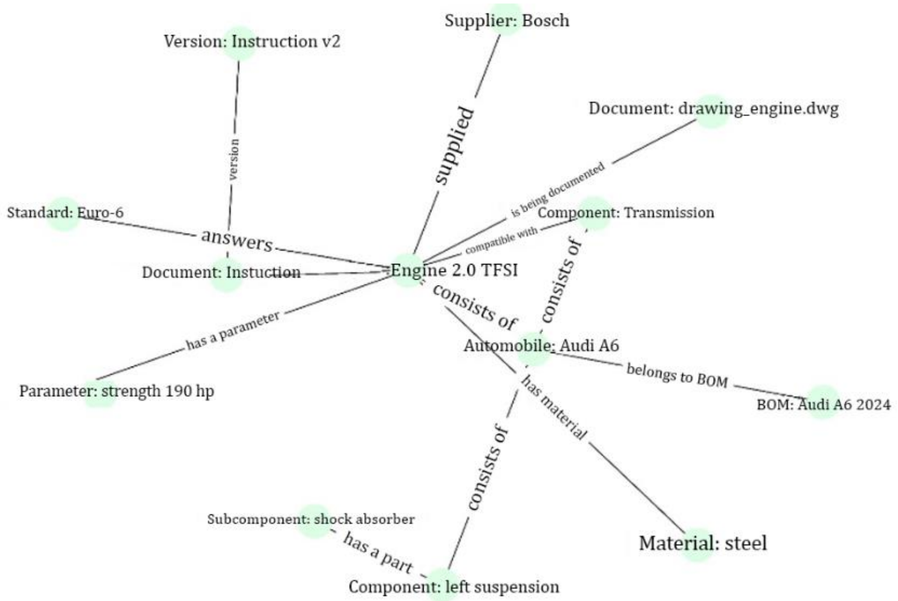

Fig. 2. Fragment of the graph database (Audi A6, components, documents, relationships)

where each vertex corresponds to an entity of the subject domain. The relationships between the vertices form the set of edges E, which are defined as ordered pairs with a designated predicate:

$$\begin{gathered} E = \{(Car_{AudiA6}, Enginie_{2.0TFSI}, has\ part), \\ (Engine_{2.0TFSI}, Drawing_{ENG}, documented\ by), \\ (Engine_{2.0TFSI}, Instruction_{ENG}, documented\ by), \\ (Instruction_{ENG}, Version_{v2}, version\ of), \\ (Engine_{2.0TFSI}, Parameter_{Power}, has\ parameter), ...\}. \end{gathered} \tag{9}$$

Thus, the graph implements relations such as “the car consists of an engine,” “the engine is documented by a drawing,” “the engine has the parameter power,” “the engine is supplied by Bosch, ”and so on. For example, the edge

$$(Car_{AudiA6}, Enginie_{2.0TFSI}, has\ part). \tag{10}$$

formalizes the statement that the Audi A6 car includes the 2.0 TFSI engine. And the edge

$$(Engine_{2.0TFSI}, Parameter_{Power}, has\ parameter) \tag{11}$$

indicates that this engine is characterized by the parameter “power 190 hp.”

In addition to structural relationships, the system supports component compatibility relations. For example, the transmission is compatible with the 2.0 TFSI engine, which is represented as

$$(Transmission, Engine_{2.0TFSI}, compatible\ with). \tag{12}$$

This makes it possible to automatically verify the interchangeability of units during the redesign of the automobile.

User queries are formulated in the form of first-order logical predicates. If an engineer is interested in the drawing of the Audi A6 engine, the search expression takes the form:

$$\begin{gathered}\exists x \exists y (Component(x) \wedge BelongsTo(x, Car_{AudiA6}) \wedge \\ \wedge HasType(x, Engine) \wedge DocumentedBy(x, y) \wedge \\ \wedge Drawing(y)).\end{gathered} \tag{13}$$

After translation, this query can be executed in the Cypher language [28]:

```
MATCH (c:Car {name: "Audi A6"})-[:HAS_PART]-
    >(e:Component {type: "Engine"})-
[: DOCUMENTED_BY]->(d:Drawing)
RETURN d;
```

The result of execution will be the node $Drawing_{ENG}$, corresponding to the file *Drawing_engine.dwg*.

It is important to note that the semantic model also supports semantic distance between objects, which makes it possible to find similar nodes even in the absence of a direct connection. Suppose a user is searching not for the drawing of the Audi A6 engine, but for the drawing of an “engine with a power of about 200 hp.” In this case, instead of direct search, the minimum semantic distance (5) is used. For example, if the graph contains another engine $Engine_{2.0TDI}$ with the parameter power of 192 hp, the system can retrieve its drawing through the relation “engine ↔ parameter ↔ drawing.”

Fig. 2 clearly illustrates this approach: the vertex “Car: Audi A6” is linked to key components (“Engine 2.0 TFSI,” “Transmission,” “Left suspension”), each of which has its own documentation, parameters, materials, and suppliers. Such a structure enables integrated intelligent search and provides a formalized basis for automated management of engineering documentation.

Efficiency and Performance

A knowledge base was modeled for 1000 automobiles, each consisting of approximately 50 components, resulting in about 50,000 entities and 200,000 relationships. The execution time of three typical queries was evaluated:

- Q1: find the engine drawing of a specific car;
- Q2: find all compatible components for a given unit (e.g., transmission);
- Q3: find engine drawings by a semantically close parameter (power ±10 hp).

Formally, the execution time of a query was defined as:

$$T_q = \frac{1}{N}\sum_{i=1}^{N} t_i, \quad N = 100, \tag{14}$$

where t_i is the processing time of the i-th query.

The results are presented in Table 1.

Table 1 – **Performance comparison of three database models by query execution time (Q1–Q3)**

Query	Relational model (ms)	Graph model (ms)	Semantic model (ms)
Q1	95	40	32
Q2	250	120	70
Q3	370	210	85

From the table it can be seen that for simple queries (Q1) all models provide similar results, but the semantic model is still slightly faster due to optimized relationships. For complex multi-level queries (Q2, Q3), the semantic model demonstrates the best performance: it is 3–4 times more efficient than the relational model and 2–2.5 times faster than the “purely graph-based” model. Additionally, the accuracy of search by semantic parameters was measured. The relevance function (7) was applied. The semantic model showed the highest relevance of results when searching by approximate parameters (for example, “engines with power around 200 hp”).

The proposed semantic model combines the advantages of graph-based data storage with the ontological approach and ensures both higher performance and more intelligent search. Its use is justified in systems where complex relationships between objects and semantic-based search are required.

Conclusions

The conducted research has made it possible to develop an ontological model of technical objects that provides an integrated representation of their structure, properties, and interrelations with supporting documentation. The proposed approach combines the capabilities of graph databases with ontological analysis methods, enabling greater flexibility in representing complex hierarchical dependencies and improving the efficiency of information retrieval. The practical implementation has shown that the use of the model significantly reduces the time required for processing technical data and ensures system scalability as the volume of information increases. Comparison with existing approaches has confirmed the scientific novelty of the work, which lies in the integration of ontological modeling with tools for the automated incorporation of manuals and drawings into a unified information space. The obtained results create a foundation for the further development of intelligent solutions in the field of technical operation, in particular through the implementation of semantic analysis algorithms and machine learning methods for automating the processes of knowledge processing and updating.

Future research in this direction should focus on expanding the capabilities of the ontological model through the integration of natural language processing algorithms and machine learning.

This will make it possible to automatically extract knowledge from unstructured sources, such as textual manuals or technical descriptions, and dynamically enrich the knowledge base with new objects and their properties. Promising applications also include the use of semantic search methods to support engineering decisions in real time and the development of recommendation systems for optimizing the operation of technical objects.

Furthermore, future research may be related to the development of mechanisms for interoperability between different ontological models and the standardization of their description within cross-domain systems. This will facilitate the integration of technical data with external information resources and open the possibility of constructing comprehensive digital twins that will provide more complete forecasting of the life cycle of objects and enhance the efficiency of their utilization.

Reference

1. Zhang, X., Zhang, Q., Zhao, Q., Zhao, Y., Guo, Y and Zhao, Z., Song, L. (2025), "An axiomatic system engineering design method based on NSGA-II algorithm applied to complex systems", *Scientific Reports*, vol. 15(1), article number 31766, doi: https://doi.org/10.1038/s41598-025-17689-5
2. Pustovoitov, P., Voronets, V., Voronets, O., Sokol, H. and Okhrymenko, M. (2024), "Assessment of QOS indicators of a network with udp and tcp traffic under a node peak load mode", *Eastern European Journal of Enterprise Technologies*, vol. 1(4(127)), pp. 23–31, doi: https://doi.org/10.15587/1729-4061.2024.299124
3. Zakovorotniy, A., and Kharchenko, A. (2021), "Optimal speed controller design with interval type-2 fuzzy sets", *2021 IEEE 2nd KhpiWeek on Advanced Technology*, pp. 363–366, doi: https://doi.org/10.1109/KhPIWeek53812.2021.9570045
4. Kuchuk, H., Kalinin, Y., Dotsenko, N., Chumachenko, I. and Pakhomov, Y. (2024), "Decomposition of integrated high-density IoT data flow", *Advanced Information Systems*, vol. 8, no. 3, pp. 77–84, doi: https://doi.org/10.20998/2522-9052.2024.3.09
5. Pustovoitov, P., Sokol, G., Hroza, P., Tyrtyshnikov, O. and Rvachova, N. (2018), "Mathematical Model of Single-Channel Infocommunication Node with Several Packet Flows", 2018 Int. Scientific Practical Conf. on Problems of Infocommunications Science and Technology Pic S and T 2018, pp. 166–170, 8632128, doi: https://doi.org/10.1109/INFOCOMMST.2018.8632128
6. Terenyk, D. and Kuchuk, H. (2020), "SQL & NOSQL database comparison by case designing affiliate system report", *Radioelectronic and Computer Systems*, vol. 2020 (1-93), pp. 83–89, doi: https://doi.org/10.32620/reks.2020.1.08
7. Kuchuk, G.A., Akimova, Yu.A. and Klimenko, L.A. (2000), "Method of optimal allocation of relational tables", *Engineering Simulation*, vol. 17(5), pp. 681–689, available at: https://www.scopus.com/record/display.uri?eid=2-s2.0-0034512103&origin=resultslist
8. Hildebrandt, C., Kocher, A., Kustner, C., Lopez-Enriquez, C.-M., Muller, A. W., Caesar, B., Gundlach, C.S. and Fay, A. (20210), "Ontology Building for Cyber-Physical Systems: Application in the Manufacturing Domain", *IEEE Transactions on Automation Science and Engineering*, vol. 17(3), pp. 1266–1282, 9097408, doi: https://doi.org/10.1109/TASE.2020.2991777
9. Kuchuk, N., Kashkevich, S., Radchenko, V., Andrusenko, Y. and Kuchuk, H. (2024), "Applying edge computing in the execution IoT operative transactions", *Advanced Information Systems*, vol. 8, no. 4, pp. 49–59, doi: https://doi.org/10.20998/2522-9052.2024.4.07
10. Kendall, E. F. and McGuinness, D. L. (2019), *Ontology Engineering. Synthesis Lectures on Data, Semantics, and Knowledge*. Springer Nature Switzerland AG, doi: https://doi.org/10.1007/978-3-031-79486-5
11. Allemang, D., Hendler, J. and Gandon, F. (2020*), Semantic Web for the Working Ontologist: Effective Modeling in Linked Data, RDFS and OWL*, Third Edition, ACM Books, Morgan and Claypool, 510 p., doi: https://doi.org/10.1145/3382097
12. Sequeda, J. and Lassila, O. (2020), *Designing and Building Enterprise Knowledge Graphs*, Synthesis Lectures on Data, Semantics, and Knowledge. Morgan and Claypool, doi: https://doi.org/10.1007/978-3-031-01916-6
13. Moltmann, F. (2021), "Natural Language Ontology", *Routledge Handbook of Metametaphyics*, CNRS, pp. 325–328, available at: https://www.researchgate.net/publication/341030712_Natural_Language_Ontology
14. Mateiu, P. and Groza, A. (2023), "Ontology Engineering with Large Language Models", *arXiv preprint*, July 2023, doi: https://doi.org/10.48550/arXiv.2307.16699
15. Scharnhorst, A. and Smiraglia, R. P. (2022), *Linking Knowledge: Linked Open Data for Knowledge Organization*, doi: http://dx.doi.org/10.48550/arXiv.2204.14041
16. Moltmann, F. (2020), "Truthmaker Semantics for Natural Language: Attitude Verbs, Modals and Intensional Transitive Verbs", *Theoretical Linguistics*, vol. 46(3–4), pp. 159–200, doi: https://doi.org/10.1515/tl-2020-0010
17. Hitzler, P. and Sarker, Md K. (2022), *Neuro-Symbolic Artificial Intelligence: The State of the Art*, IOS Press, available at: https://www.iospress.com/catalog/books/neuro-symbolic-artificial-intelligence-the-state-of-the-art?utm_source=chatgpt.com
18. Hitzler, P., Sarker, Md K. and Eberhardt, A. (2023), *Compendium of Neurosymbolic Artificial Intelligence*, IOS Press, available at: https://www.iospress.com/catalog/books/compendium-of-neurosymbolic-artificial-intelligence?utm_source=chatgpt.com
19. Smith, B. and Landgrebe, J. (2024), *Why Machines Will Never Rule the World: Artificial Intelligence Without Fear*, Revised and Enlarged Edition, 434 p., doi: https://doi.org/10.4324/9781003581253
20. (2025), *Neo4j Graph Database Platform Documentation*, Neo4j Developer Portal, available at: https://neo4j.com/docs
21. (2023), *PostgreSQL 15 Documentation*, PostgreSQL Development Group, available at: https://www.postgresql.org/docs
22. Fakih, G. and Serrano-Alvarado, P. (2025), "A survey on SPARQL query relaxation under the lens of RDF reification", *Semantic Web*, vol. 15(6), pp. 2507–2554, doi: https://doi.org/10.3233/SW-243621
23. Kalinin, Y., Kozhushko, A., Rebrov, O., and Zakovorotniy, A. (2022), "Characteristics of Rational Classifications in Game-Theoretic Algorithms of Pattern Recognition for Unmanned Vehicles", *2022 IEEE 3rd Khpi Week on Advanced Technology Khpi Week 2022 Conference Proceedings*, 03-07 October 2022, doi: https://doi.org/10.1109/KhPIWeek57572.2022.9916454
24. Kuchuk, N., Mozhaiev, O., Mozhaiev, M. and Kuchuk, H. (2017), "Method for calculating of R-learning traffic peakedness", *2017 4th International Scientific-Practical Conference Problems of Infocommunications Science and Technology*, PIC S and T 2017 – Proceedings, pp. 359–362, doi: https://doi.org/10.1109/INFOCOMMST.2017.8246416

25. Ibrahimov, B., Hashimov, E. and Ismayılov, T. (2024), "Research and analysis mathematical model of the demodulator for assessing the indicators noise immunity telecommunication systems", *Advanced Information Systems*, vol. 8, no. 4, pp. 20–25, doi: https://doi.org/10.20998/2522-9052.2024.4.03
26. Kuchuk, H., Mozhaiev, O., Kuchuk, N., Tiulieniev, S., Mozhaiev, M., Gnusov, Y., Tsuranov, M., Bykova, T., Klivets, S., and Kuleshov, A. (2024), "Devising a method for the virtual clustering of the Internet of Things edge environment", *Eastern-European Journal of Enterprise Technologies*, vol. 1, no. 9 (127), pp. 60–71, doi: https://doi.org/10.15587/1729-4061.2024.298431
27. (2013), *SPARQL 1.1 Overview*, W3C Recommendation, 21 March, available at: https://www.w3.org/TR/sparql11-overview
28. (2023), *Cypher Query Language Documentation*, Neo4j, available at: https://neo4j.com/docs/cypher-manual/current
29. Sankar, S. (2024), *TF-IDF and BM25 for RAG— a complete guide*, AI Bites, 7 October 2024. available at: https://www.ai-bites.net/tf-idf-and-bm25-for-rag-a-complete-guide
30. Seitz, R. (2020), *Understanding TF-IDF and BM-25*, KMW Technology, 20 March 2020, available at: https://kmwllc.com/index.php/2020/03/20/understanding-tf-idf-and-bm-25
31. (2025), *BM25 relevance scoring – Azure AI Search Microsoft Learn*, Microsoft, 17 January 2025, available at: https://learn.microsoft.com/en-us/azure/search/index-similarity-and-scoring

Received (Надійшла) 15.05.2025
Accepted for publication (Прийнята до друку) 13.08.2025

ВІДОМОСТІ ПРО АВТОРІВ / ABOUT THE AUTHORS

Пустовойтов Павло Євгенович – доктор технічних наук, професор, завідувач кафедри "Системи інформації ім. В.О. Кравця", Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Pavlo Pustovoitov – Doctor of Technical Sciences, Professor, Head of the Department of Information Systems named after V.O. Kravets, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: pavlo.pustovoitov@khpi.edu.ua; ORCID Author ID: https://orcid.org/0000-0003-3884-0200;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=55225675900.

Компанієць Володимир Олександрович – старший викладач кафедри "Системи інформації ім. В.О. Кравця", Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Volodymyr Kompaniiets – Senior Lecturer, Department of Information Systems named after V.O. Kravets, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: Volodymyr.Kompaniiets@khpi.edu.ua; ORCID Author ID: https://orcid.org/0000-0003-2909-6993.

Лю Іюн – аспірант кафедри "Системи інформації ім. В.О. Кравця", Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Yiyong Liu – Postgraduate student of the Department of Information Systems named after V.O. Kravets, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: liuyiyong2001@gmail.com; ORCID Author ID: https://orcid.org/0009-0005-0654-9306.

Сокол Галина Вікторівна – кандидат технічних наук, доцент, доцент кафедри "Системи інформації ім. В.О. Кравця", Національний технічний університет "Харківський політехнічний інститут", Харків, Україна;
Galina Sokol – Candidate of Technical Sciences, Associate Professor, Associate Professor of the Department of Information Systems named after V.O. Kravets, National Technical University "Kharkiv Polytechnic Institute", Kharkiv, Ukraine;
e-mail: Halyna.Sokol@khpi.edu.ua; ORCID Author ID: https://orcid.org/0000-0003-4847-518X;
Scopus ID: https://www.scopus.com/authid/detail.uri?authorId=55975966600.

Авад-Алла Мохамед Атеф – доктор філософії, викладач та координатор кафедри мехатроніки, інженерний факультет, Сучасний університет технологій та інформації, Каїр, Єгипет;
Mohamed Awad-Alla – Candidate of Technical Sciences, Lecturer and Coordinator of the Department of Mechatronics, Faculty of Engineering, Modern University for Technology and Information, Cairo, Egypt;
e-mail: mohammed.awad-allah@eng.mti.edu.eg; ORCID Author ID: https://orcid.org/0009-0008-2770-3220.

Модель даних компонентів складних технічних систем на базі семантичних мереж

П. Є. Пустовойтов, В. О. Компанієць, Іюн Лю, Г. В. Сокол, М. А. Авад-Алла

Анотація. У представленій статті розглянуто проблему онтологічного моделювання технічних об’єктів, які характеризуються складною багаторівневою структурою, різноманітними функціональними властивостями та значним обсягом супровідної документації. Запропоновано метод представлення предметної області у вигляді онтології, що інтегрує дані про об’єкти, їхні складові елементи та зв’язки з технічними інструкціями та кресленнями. На основі розробленої онтологічної схеми побудовано графову базу даних, яка забезпечує ефективний пошук, збереження та обробку знань, а також підтримку багатокористувацького доступу з урахуванням прав користувачів. Для формалізації властивостей моделі застосовано математичний апарат теорії графів та онтологічних відношень, що дозволило описати закономірності побудови та функціонування системи. У роботі виконано порівняльний аналіз переваг та недоліків існуючих підходів до організації знань, визначено місце і новизну запропонованої моделі у контексті сучасних досліджень. Експериментальні результати підтвердили переваги застосування графових баз даних для роботи з технічною інформацією, зокрема підвищення швидкодії при виконанні пошукових запитів та зниження витрат на оновлення даних. Практична значущість дослідження полягає у можливості використання розробленої моделі для автоматизованої підтримки життєвого циклу складних технічних систем, включаючи етапи експлуатації, обслуговування та модернізації. Отримані результати створюють підґрунтя для подальшого впровадження інтелектуальних технологій обробки знань у галузі інженерії та технічної експлуатації.

Ключові слова: онтологічне моделювання; графові бази даних; технічні об'єкти; теорія графів; управління знаннями; технічна документація; життєвий цикл систем.