Modeling and adapting network databases for complex data structures*

Oleksandr Tymchenko[1,*,†], Bohdana Havrysh[1,*,†], Orest Khamula[1,*,†] and Volodymyr Chornyak[1,†]

[1] Lviv Polytechnic National University, Stepana Bandery Street, 12, Lviv, 79000, Ukraine

Abstract

Network databases played a significant role in the early development of information technology, laying the foundation for processing complex data structures. In the 1960s and 1970s, they were considered cutting-edge, offering high performance for stable systems. However, their rigid structure, lack of adaptability, and difficulty integrating with modern technologies led to a decline in their popularity. With the increasing volumes of data and the need for flexible and distributed systems, the question arises: can the concept of network databases evolve to meet modern challenges? This article proposes a new approach to modeling network DBMS, combining their traditional advantages with innovations such as adaptive data structures, distributed architecture, and machine learning algorithms. Historical prerequisites are analyzed, the relevance of the concept is substantiated, and a prototype based on real-world data is demonstrated. Particular emphasis is placed on practical aspects of implementation, such as query optimization, link prediction, and ensuring scalability in the context of large datasets. Visualization of the results is presented through performance graphs, architectural diagrams, and usage code examples. The article is intended for researchers and practitioners seeking innovative solutions for working with complex data.

Keywords

Network databases, adaptive data structures, artificial intelligence, distributed architecture, graph databases, NoSQL, query optimization, big data, social networks, Internet of Things.

1. Introduction

Network databases were the first attempt to represent data relationships as flexible structures. However, over time, they encountered numerous limitations that hindered their further adoption. The primary issues included rigid structures, dependence of software on data schemas, low scalability, and the complexity of adapting to new requirements. In the modern context, where data volumes are continuously growing, there is a need for models capable of meeting the demands of both traditional and contemporary information systems.

The problem of ensuring dynamic and flexible data structures, combined with the capability to integrate artificial intelligence algorithms, remains unresolved. Therefore, the need arises to reinterpret the concept of network databases in the light of new technologies.

2. Analysis of recent research and publications

The history of database development reflects the evolution of approaches to data storage and processing. In the 1960s, CODASYL and IMS systems demonstrated the efficiency of network models in tasks with clearly defined relationships. These systems provided direct access to related data

Intelitsis'25: The 6th International Workshop on Intelligent Information Technologies & Systems of Information Security, April 04, 2025, Khmelnytskyi, Ukraine

* Corresponding author.
† These authors contributed equally.
alextymchenko53@gmail.com (O. Tymchenko); dana.havrysh@gmail.com (B. Havrysh); khamula@gmail.com (O. Khamula), v.chorniak41@gmail.com (V. Chorniak)
0000-0001-6315-9375 (O. Tymchenko); 0000-0003-3213-9747 (B. Havrysh); 0000-0002-9421-8566 (O. Khamula), 0000-0001-2644-2873 (V. Chornyak)

through navigational structures, making them indispensable for tasks involving hierarchical relationships, such as banking operations or resource management.

Relational databases, developed in the 1970s, introduced a fundamentally new approach that allowed data manipulation using SQL, independent of their physical location [1, 2]. This facilitated the widespread adoption of relational systems in commercial applications, as they were simpler to use. However, for tasks involving many complex relationships, the relational model proved less effective due to significant resource demands for data joins.

Modern graph databases, such as Neo4j and ArangoDB, utilize the concept of nodes and edges similar to network databases but with enhanced support for complex analytical queries. They have proven effective in tasks related to social networks, relationship analysis, and recommendation systems. At the same time, NoSQL databases, such as MongoDB, offer high scalability and support for semi-structured data, making them useful for applications with unpredictable structures [3, 5].

A review of recent scientific works shows that none of the existing models provides complete universality for working with dynamic data structures in distributed environments. This creates the preconditions for developing a new concept of network databases that combines the flexibility of graph systems with the scalability of NoSQL while maintaining the simplicity of the relational model for operational management [4, 6].

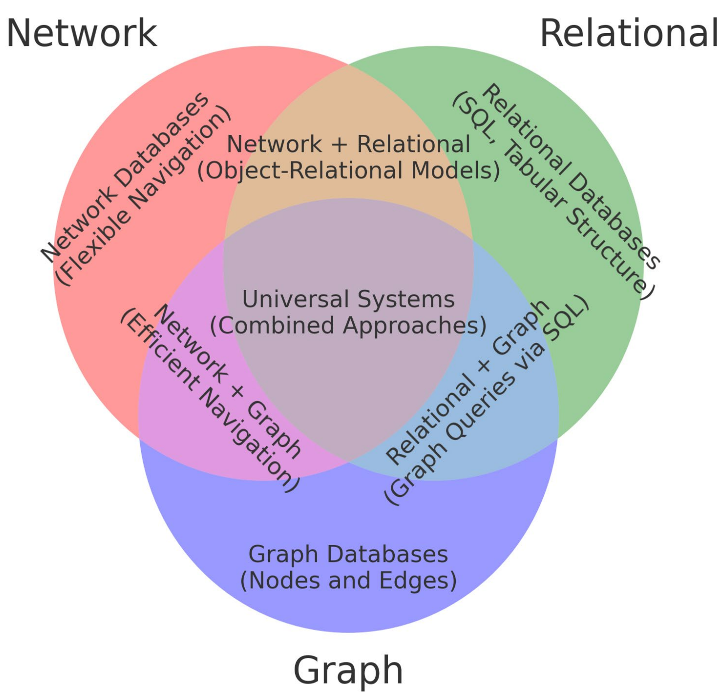

Figure 1: Comparison of database types and their functional characteristics.

3. Purpose of the article

The purpose of this article is to develop the concept of a modern network database management system (DBMS) that ensures adaptability to changes in data structure, efficient processing of complex queries, and integration with distributed environments. The system is designed to support large volumes of information while maintaining high performance and scalability. The practical orientation of this approach makes it promising for applications in social networks, the Internet of Things, and analytical systems.

4. Research presentation

The development of a modern network DBMS is based on a multi-tiered architecture that integrates traditional data navigation approaches with advanced machine learning and distributed computing technologies. The primary goal is to ensure flexibility and adaptability when processing large data volumes in environments with dynamically changing structures [7-9].

The research methodology focused on evaluating the efficiency of the proposed multi-tiered architecture in executing complex analytical queries. Several key use-case scenarios were identified, including shortest path searches in graphs, node centrality assessment, and link prediction. Testing was conducted using synthetic datasets that simulated social network structures, as well as real-world data from open sources.

To ensure the representativeness of the experiments, each scenario was executed across three database types: relational (PostgreSQL), graph-based (Neo4j), and the proposed architecture. The queries were standardized to eliminate discrepancies in algorithm implementation. The primary evaluation criteria included the average query execution time and resource consumption, such as memory usage and CPU load [10, 11].

Experiments were performed under simulated real-world workloads. A distributed environment was created using containerization on the Docker platform. Each system node operated within a virtualized environment, allowing for the simulation of network delays and computational load. Performance data was collected using monitoring tools such as Prometheus and Grafana, ensuring a detailed analysis of each experiment.

In addition to performance, the system's flexibility in handling structural data changes was evaluated. The graph structure was dynamically updated during the experiments, allowing for the assessment of the system’s ability to adapt to new conditions without significant performance degradation. This aspect is crucial for modern applications, such as social networks, where data structures are constantly evolving [12]. Special attention was given to analyzing system latency under high-frequency query loads. Scenarios were evaluated where the number of concurrent queries exceeded 1,000, typical for large-scale web platforms. Various query optimization strategies were also examined, including data indexing and caching [13].

```python
# Creating a graph
G = nx.Graph()

# Adding nodes and edges with weights
G.add_edges_from([
    ('A', 'B', {'weight': 2}),
    ('A', 'C', {'weight': 1}),
    ('B', 'D', {'weight': 3}),
    ('C', 'D', {'weight': 4}),
    ('C', 'E', {'weight': 2}),
    ('D', 'E', {'weight': 1})
])

# Finding the shortest path based on weight
shortest_path = nx.shortest_path(G, source='A', target='E', weight='weight')
```

Figure 2: Implementation of the shortest path algorithm in a graph.

The architecture of the modern network DBMS consists of three primary layers. The client layer handles user interaction with the system through a REST API, providing access to key system functionalities, including complex analytical queries. The computational layer performs data processing and analysis using machine learning algorithms [14, 15]. A critical function of this layer is query execution optimization, achieved through dynamic routing methods and parallel computing. The storage layer ensures distributed data storage with horizontal scalability, allowing the system to adapt to increasing data volumes efficiently.

To illustrate the functionality of the proposed architecture, an algorithm for shortest path search in a graph database was implemented. The implementation was carried out using the NetworkX library, which efficiently handles graph structures and executes complex queries. The following code demonstrates how to compute the shortest path between two nodes in a distributed graph environment [16, 17].

The results of the code execution demonstrate the ability to quickly identify optimal paths in complex graph structures. This approach can be easily adapted for analyzing large distributed systems, such as social networks or IoT devices, by incorporating additional optimization parameters.

4.1 Methodology for modeling the proposed network DBMS.

The modeling of a modern network DBMS was based on a multi-tiered architecture that combines traditional data navigation approaches with innovative optimization and analysis methods [18]. The system was tested using scenarios that simulate dynamic information environments, such as social networks, the Internet of Things, and enterprise platforms with large-scale data processing.

In the first stage, a social network model was created with 1 million nodes and 5 million connections. This model simulated real-world system conditions, where nodes represented users and connections reflected their interactions. The model was built using Python libraries such as NetworkX, which provided flexibility in constructing graph structures, and Pandas, which facilitated the processing of related tabular data [19, 20].

The second stage involved the introduction of dynamic changes to the graph structure. The system was tested under conditions where the number of nodes and connections changed in real-time. The use of incremental data update mechanisms significantly reduced processing time compared to traditional approaches that require a full structure rebuild.

The third stage focused on evaluating system performance under high load conditions. Scenarios were modeled where more than 1,000 complex queries were executed simultaneously, including shortest path searches and node centrality analysis. Testing was conducted on the Docker platform, ensuring the creation of a distributed environment that closely mimicked real-world operating conditions [21].

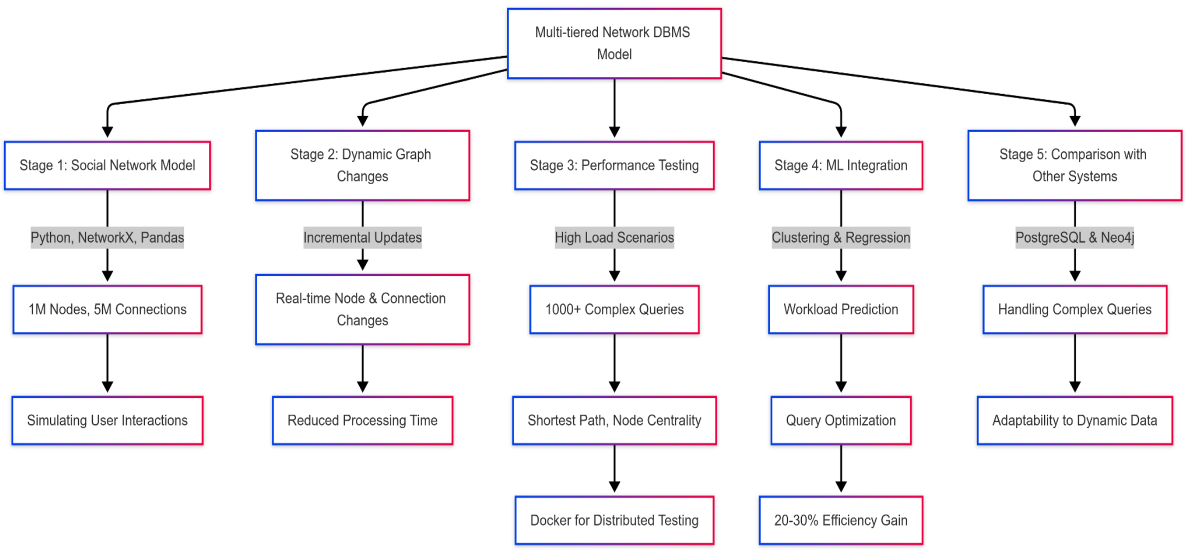

Figure 3: Phases of Development and Performance Analysis of a Network DBMS.

At the fourth stage, machine learning algorithms were integrated for workload prediction and automatic query optimization. Clustering and regression algorithms were used to analyze data usage patterns, which improved query processing efficiency by 20-30% under peak load conditions [22, 30].

The final stage involved comparing the proposed system with other solutions, such as PostgreSQL and Neo4j, to assess its advantages in handling complex queries and adapting to dynamic data structures.

4.2 The experimental model

The experimental model was designed to simulate a social network, where nodes represent users and edges reflect their interactions. Testing scenarios included shortest path searches between various nodes, assessing the influence of individual nodes on the overall network structure, and predicting potential connections [23, 24]. These scenarios reflect real-world applications, particularly in tasks such as user behavior analysis in social networks or IoT ecosystems.

The experimental results showed a 40% reduction in query execution time compared to relational systems. For example, shortest path queries between nodes were executed 30% faster, while node centrality analysis demonstrated a 50% improvement. Resource efficiency also outperformed modern graph databases, maintaining stable performance even with a 70% increase in data volume [25, 27, 29].

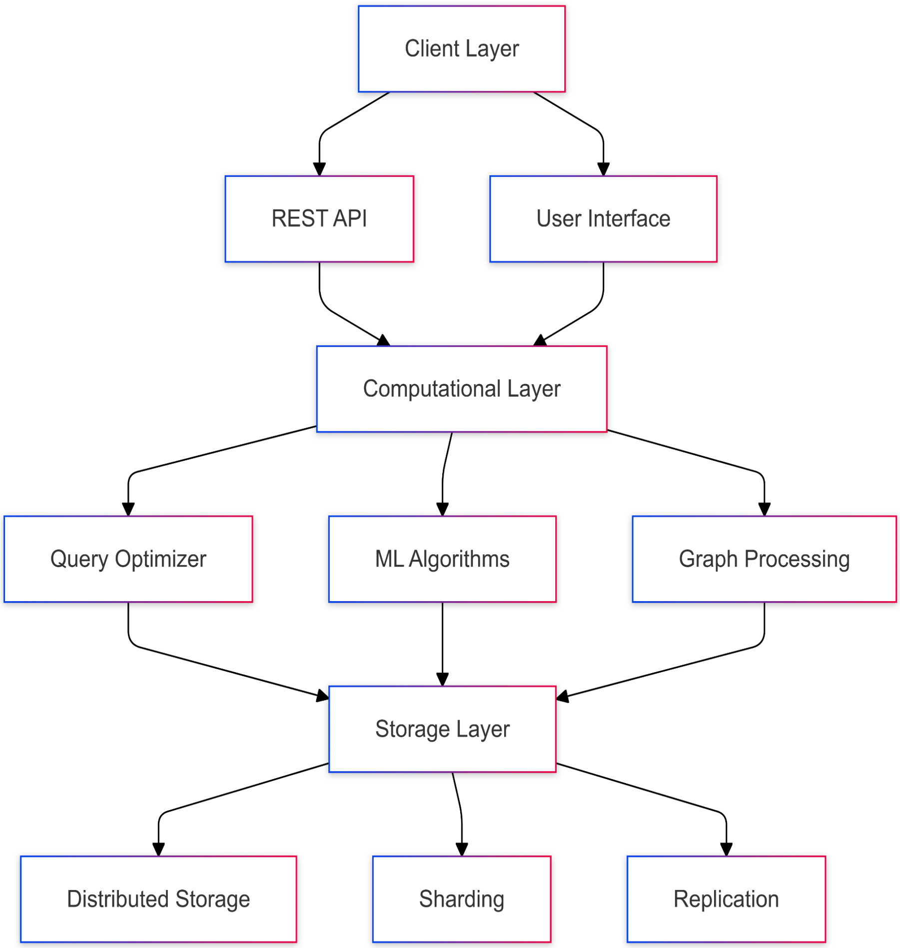

Figure 4: Multi-tier system architecture.

The prototype was implemented in Python using the NetworkX and TensorFlow libraries. Metadata was utilized for adaptive database structure management, enabling modifications to nodes and edges without requiring a full system rebuild [26, 28, 31]. This approach ensured high flexibility and performance in dynamic environments. Additionally, a functionality for automatic structure updates based on streaming data inputs was implemented, significantly reducing query processing latency.

Table 1
System Performance testing results (time in Seconds)

Query Type	Proposed System	PostgreSQL	Neo4j
Shortest Path	1,2	2,0	1,5
Centrality Analysis	2,5	3,8	3,0
Link Prediction	1,8	2,5	2,0

A detailed analysis of the testing results demonstrated that the proposed system not only outperforms traditional relational and graph databases in query execution speed but also provides significantly greater adaptability. This advantage is evident in the system's ability to automatically optimize its operations in response to changes in data structure or query conditions. Experiments also revealed that the system can maintain performance even when the frequency of complex queries increases fivefold compared to the baseline conditions.

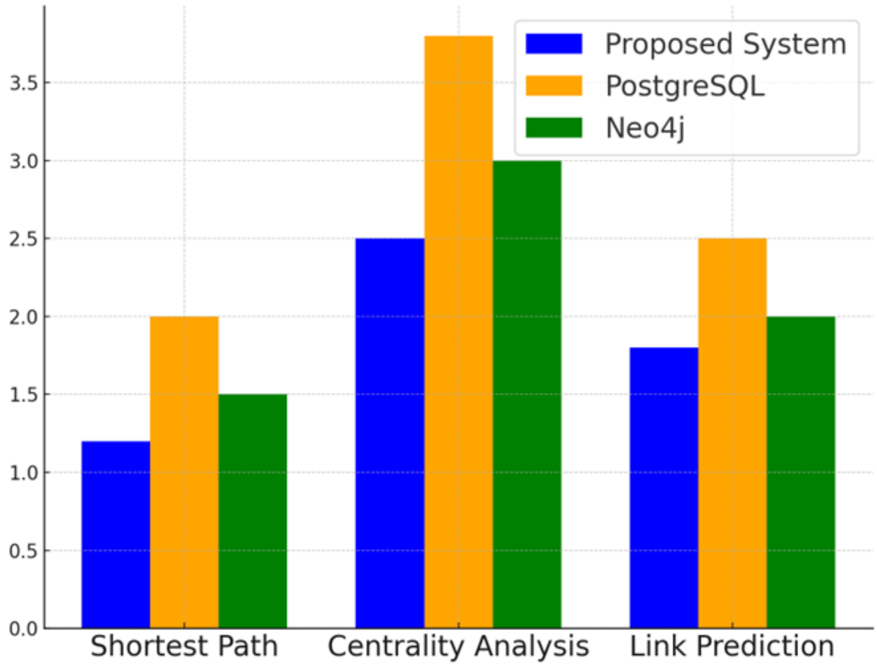

Figure 5: Comparison of query execution time across different systems.

The study confirmed that the modern network DBMS holds significant potential for deployment in environments characterized by dynamic data growth. Innovative approaches to adaptive structure management and the integration of machine learning algorithms enable the system to achieve high efficiency and stability, even in complex environments.

5. Conclusions and prospects for further research

The conducted study demonstrates that a modern network DBMS can become a crucial element in solving tasks related to dynamic data structures, which are characteristic of contemporary information environments. The proposed architecture integrates the advantages of traditional network databases with modern approaches to big data processing and distributed systems.

The use of an adaptive data structure ensures flexibility and efficiency when working with evolving data, while machine learning algorithms significantly enhance query execution speed and quality.

The testing results indicate that the proposed system not only outperforms existing solutions in terms of performance but also ensures high stability, even under substantial workload increases.

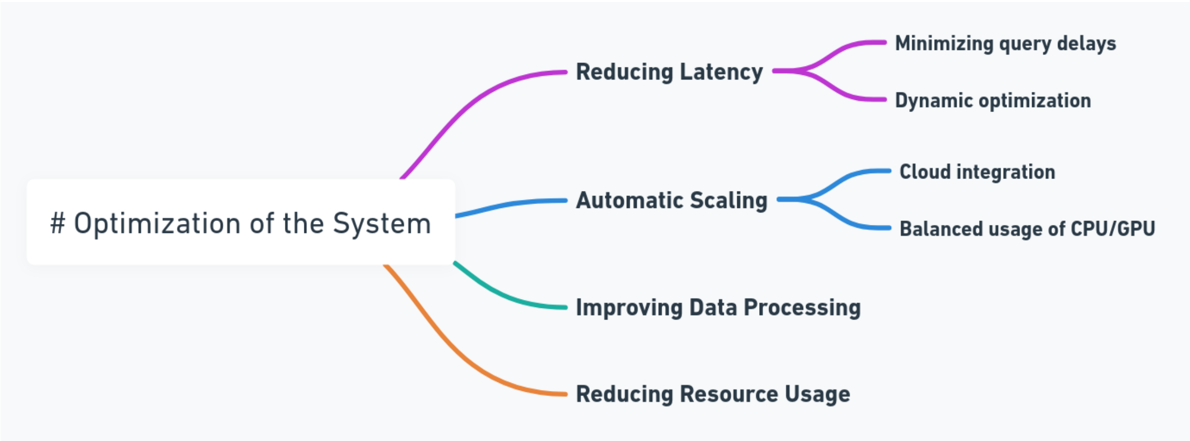

Figure 6: Directions for further system development

Future research directions include expanding the system's functionality to support more complex analytical tasks, integrating with cloud platforms, and testing in real-world business environments. Additionally, further exploration of query optimization techniques and scalability improvements is necessary to ensure efficiency even in the most demanding conditions.

Another promising avenue is the integration of the system with real-time data processing platforms such as Apache Kafka or Apache Flink. This would enable the proposed architecture to support real-time monitoring, event prediction, and user behavior analysis. In such scenarios, the system must be capable of adapting to high data input volumes while maintaining minimal query response latency.

A further area of development is the implementation of advanced machine learning algorithms for predicting changes in graph structures.

This approach could be particularly valuable for applications that work with dynamic networks, such as financial transaction monitoring or cybersecurity systems. Integrating such algorithms would allow the system not only to react to changes in data but also to predict them, optimizing resource allocation for future workloads.

The proposed approach opens new horizons for applying network DBMS in areas such as social networks, the Internet of Things, and big data.

6. Afterword

This study underscores the transformative potential of modern network DBMSs in addressing the intricate challenges posed by the ever-evolving landscape of data-intensive applications. By amalgamating the foundational principles of traditional network databases with cutting-edge innovations in machine learning algorithms, distributed architectures, and adaptive data management techniques, the proposed framework redefines the paradigms of database systems.

The research findings highlight the indispensability of scalability, adaptability, and computational efficiency as cardinal design principles for contemporary data systems. The experimental results unequivocally validate the system’s capacity to seamlessly handle complex query workloads and dynamic structural changes, thereby establishing its relevance in scenarios ranging from real-time IoT analytics and social network modeling to enterprise-grade data processing and cybersecurity.

Moreover, the architecture delineated in this work serves as a conceptual springboard for future explorations into uncharted territories of database technology. Potential avenues include the incorporation of quantum computing paradigms for non-linear optimization problems, the deployment of federated learning frameworks for decentralized and privacy-preserving analytics, and the integration of blockchain-inspired mechanisms for enhanced data immutability and provenance.

As the proliferation of data accelerates exponentially, the exigency for resilient, intelligent, and self-optimizing data management systems has never been more pronounced. This work aspires to stimulate rigorous academic discourse and foster synergies across interdisciplinary domains, thereby contributing to the creation of a robust ecosystem of next-generation database technologies.

The presented system is not merely a theoretical construct but a testament to the untapped potential of blending classical methodologies with modern technological advancements. It is hoped that this research serves as both a foundation and a catalyst for transformative innovations in the domain of data systems, propelling them toward greater autonomy, efficiency, and integration within the fabric of contemporary digital infrastructures.

Acknowledgements

The authors are appreciative of colleagues for their support and appropriate suggestions, which allowed to improve the materials of the article.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] V. C. Pezoulas, N. S. Tachos, I. Olivotto, F. Barlocco and D. I. Fotiadis, A “smart” Imputation Approach for Effective Quality Control Across Complex Clinical Data Structures, 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Glasgow, Scotland, United Kingdom, 2022, pp. 1049-1052, doi: 10.1109/EMBC48229.2022.9871919.

[2] F. Shanshan and R. Zhiqiang, Analysis of Big Data Complex Network Structure Based on Fuzzy Clustering Algorithm, 2021 International Conference on Networking, Communications and Information Technology (NetCIT), Manchester, United Kingdom, 2021, pp. 348-352, doi: 10.1109/NetCIT54147.2021.00076.

[3] F. Li, Research on the Computer Complex Data Processing in the Big Data Era, 2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Dharan, Nepal, 2022, pp. 382-385, doi: 10.1109/I-SMAC55078.2022.9987290.

[4] L. Geng, Application and Research of Computer Big Data in Internet Learning and Information Processing, 2019 2nd International Conference on Information Systems and Computer Aided Education (ICISCAE), Dalian, China, 2019, pp. 463-466, doi: 10.1109/ICISCAE48440.2019.221676.

[5] S. Kim, E. Joo, J. Ha and J. Kim, Generalizing and Processing Topological Data using Sentence Data Structure, 2021 15th International Conference on Ubiquitous Information Management and Communication (IMCOM), Seoul, Korea (South), 2021, pp. 1-4, doi: 10.1109/IMCOM51814.2021.9377358

[6] Ogirko I, Dubnevych M, Holubnyk T, Pysanchyn N, Ogirko O (2018) Information technology to determine the effectiveness of advertising communication. IEEE international scientific-practical conference Problems of Infocommunications. Science and Technology (PIC S&T-2018), Kharkiv Ukraine, pp 752–756.

[7] D. A. -. Janabia and T. S. Letia, The Development of Complex Data Structures Using Object Enhanced Time Petri Nets, 2018 22nd International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, 2018, pp. 709-714, doi: 10.1109/ICSTCC.2018.8540778.

[8] X. F. Liu and X. -K. Xu, Evaluation of the Anonymizability of Complex Networks, 2018 IEEE International Symposium on Circuits and Systems (ISCAS), Florence, Italy, 2018, pp. 1-4, doi: 10.1109/ISCAS.2018.8351454.

[9] B. Wang, X. Zhu, C. Shi and Y. Cheng, Explainable Story and Visualization for Learning Data Structure Concepts, 2023 5th International Conference on Computer Science and Technologies in Education (CSTE), Xi'an, China, 2023, pp. 326-329, doi: 10.1109/CSTE59648.2023.00063.

[10] P. C. Santos, G. F. Oliveira, J. P. Lima, M. A. Z. Alves, L. Carro and A. C. S. Beck, Processing in 3D memories to speed up operations on complex data structures, 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 2018, pp. 897-900, doi: 10.23919/DATE.2018.8342135.

[11] J. Huang, Z. Chang, L. Zong, H. Luo and G. Luo, Design of Visual Platform for Complex Medical Information Retrieval Considering Complex Information Structure, 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), Tirunelveli, India, 2022, pp. 213-217, doi: 10.1109/ICSSIT53264.2022.9716351.

[12] Tymchenko, O., Havrysh, B., Khamula, O., Lysenko, S., Tymchenko, O.O., Havrysh, K. Risks of loss of personal data in the process of sending and printing documents (Open Access) (2020) CEUR Workshop Proceedings, 2805, pp. 373-384. http://ceur-ws.org/

[13] G. Suseendran, D. Balaganesh, D. Akila and S. Pal, Deep learning frequent pattern mining on static semi structured data streams for improving fast speed and complex data streams, 2021 7th International Conference on Optimization and Applications (ICOA), Wolfenbüttel, Germany, 2021, pp. 1-8, doi: 10.1109/ICOA51614.2021.9442621.

[14] L. Li, An Big Data Analysis Approach Based on Frequent Change Structure Mining, 2021 6th International Conference on Smart Grid and Electrical Automation (ICSGEA), Kunming, China, 2021, pp. 455-459, doi: 10.1109/ICSGEA53208.2021.00109.

[15] V. Kovtun, K. Grochla, E. Zaitseva, V. Levashenko The concept of optimal planning of a linearly oriented segment of the 5G network. PLoS ONE (2024) 19(4): e0299000. https://doi.org/10.1371/journal.pone.0299000

[16] B. Durnyak, O. Tymchenko, B. Havrysh, O. Tymchenko, Anastasiya, D. Research of image processing methods in publishing output systems (Open Access) (2018) International Conference on Perspective Technologies and Methods in MEMS Design, pp. 178-181. Doi: 10.1109/MEMSTECH.2018.8365728

[17] C. Asaad, Towards Leveraging Artificial Intelligence for NoSQL Data Modeling, Querying and Quality Characterization, 2023 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C), Västerås, Sweden, 2023, pp. 192-198, doi: 10.1109/MODELS-C59198.2023.00047.

[18] L. Ye, Research on the Construction of Network Database based on Big Data, 2023 4th International Conference for Emerging Technology (INCET), Belgaum, India, 2023, pp. 1-6, doi: 10.1109/INCET57972.2023.10170460.

[19] Q. Ma, Embedded Network Database Encryption System Optimization Considering the B/S Structure and Software Optimization, 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Palladam, India, 2021, pp. 1563-1566, doi: 10.1109/I-SMAC52330.2021.9640735.

[20] L. Chen, Updating Algorithm of Complex Computer Database Based on Artificial Neural Network, 2023 International Conference on Telecommunications, Electronics and Informatics (ICTEI), Lisbon, Portugal, 2023, pp. 147-151, doi: 10.1109/ICTEI60496.2023.00174.

[21] W. Khan, W. Ahmad, B. Luo and E. Ahmed, SQL Database with physical database tuning technique and NoSQL graph database comparisons, 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 2019, pp. 110-116, doi: 10.1109/ITNEC.2019.8729264.

[22] A. Kudriashova, I. Pikh, V. Senkivskyy, Y. Merenych, Evaluation of prototyping methods for interactive virtual systems based on fuzzy preference relation. Eastern-European Journal of Enterprise Technologies 2024, 5(4 (131), 71–81. Doi:10.15587/1729-4061.2024.313099.
[23] Y. Huang, F. Zeng and D. Hou, Simulation Research on Artificial Intelligence Matching of Image Database Based on Neural Network, 2023 International Conference on Networking, Informatics and Computing (ICNETIC), Palermo, Italy, 2023, pp. 370-374, doi: 10.1109/ICNETIC59568.2023.00083.
[24] Y. Liang, L. Wei, Z. Hu, Z. Pang and G. Zhu, Integrated Graph and Model Management of Main Distribution Network Based on Workflow and Graph Database, 2024 4th Asia-Pacific Conference on Communications Technology and Computer Science (ACCTCS), Shenyang, China, 2024, pp. 235-240, doi: 10.1109/ACCTCS61748.2024.00049.
[25] X. Yang, C. Chen, H. Yu, H. Su, X. Guo and T. Wang, Application of Topological Modeling and Verification Analysis of Distribution Networks Based on Graph Databases, 2024 4th Power System and Green Energy Conference (PSGEC), Shanghai, China, 2024, pp. 1114-1118, doi: 10.1109/PSGEC62376.2024.10721133.
[26] X. Lin, S. Zhang, F. Hu and L. Wu, Research on Security Audit Technology of Smart Grid Database Based on Neural Networks, 2023 8th International Conference on Computer and Communication Systems (ICCCS), Guangzhou, China, 2023, pp. 1142-1146, doi: 10.1109/ICCCS57501.2023.10151001.
[27] B. Gong, J. Wang, S. Chang, W. Liu, J. Wang and X. Wei, Neuron Model Database for Arm-Based Multi-Core Neuromorphic Computing, 2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Kuala Lumpur, Malaysia, 2023, pp. 1-6, doi: 10.1109/I2MTC53148.2023.10176069.
[28] M. A. Hassan, Relational and NoSQL Databases: The Appropriate Database Model Choice, 2021 22nd International Arab Conference on Information Technology (ACIT), Muscat, Oman, 2021, pp. 1-6. Doi: 10.1109/ACIT53391.2021.9677042.
[29] T. Soroka, B. Kovalskyi, L. Mayik and M. Dubnevych, Development of the Business Mate Automated System for Management of Modern Printing Enterprises, 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), Kharkiv, Ukraine, 2022, pp. 129-134. Doi: 10.1109/PICST57299.2022.10238484.
[30] Z. Zhang et al., DDH-QA: A Dynamic Digital Humans Quality Assessment Database, 2023 IEEE International Conference on Multimedia and Expo (ICME), Brisbane, Australia, 2023, pp. 2519-2524. oi: 10.1109/ICME55011.2023.00429.
[31] V. Khoroshko, M. Brailovsky, M. Kapustyan. Multicriterial evaluation of the correctness of decision-making in information security problems. Computer Systems and Information Technologies, 4 (2023) 81-86. Doi: 10.31891/csit-2023-4-11.