

COMPARISON OF DATABASES. RELATIONAL AND NON- RELATIONAL DATABASES

Knysh Ludmyla

English teacher

Romanova Yuliana

Student

Ivano-Frankivsk National Technical University of Oil and Gas

Ukraine, Ivano-Frankivsk

yuliana.romanova-po231@nung.edu.ua

Introductions. Databases are used in many areas of life where large amounts of information need to be stored, organized, and processed. Simply, a database is a place to keep data. Databases are used everywhere, and with careful observation, a they can be found in the most unexpected places. For example, they can be a bus schedule, a shopping list, a school diary, banking systems, even Instagram and Facebook. Managing data well is necessary because it helps businesses work better, follow rules, and keep information safe.

Modern database management systems (DBMS) are valuable for working with large and complicated data. A DBMS is software that helps store, organize, and manage data in an easy and effective way. It has two main parts: the database, which is a collection of data, and the management system, which makes it possible to access and work with the data.

There are different types of databases. Hierarchical databases are designed to organize data in a tree-like structure, where each record has a "parent-child" relationship. This is a bit like a binary tree, but in this structure, a parent record can have many child records, while each child record is connected to only one parent. This makes hierarchical databases suitable for data that follows a clear and fixed

hierarchy. It works well for organizational charts or file directories. However, this model has difficulties when adding new relationships or accessing data outside the hierarchy.

Network databases are similar to hierarchical databases, but they allow child records to connect to multiple parent records. This mechanism creates a flexible network of relationships. For instance, in a university database, a student record can be linked to both a department and a club. This model is more complex but provides better support for "many-to-many" relationships, a concept often studied in courses like "Requirements Analysis" or "Introduction to Software Engineering."

Object-oriented databases store data in the form of objects, similar to how data is represented in object-oriented programming. Each object contains both data (attributes) and the operations (methods) that can be performed on it. This approach is ideal for applications that involve multimedia, complex data types, or real-world modeling, such as CAD systems or simulations. It allows developers to reuse code and create more realistic data models.

Cloud databases store data in virtual environments that are accessible over the internet. They allow organizations to store and access data on platforms like Amazon Web Services (AWS), Google Cloud, or Microsoft Azure. Cloud databases provide scalability, flexibility, and cost efficiency, making them suitable for businesses that need to handle large amounts of data or support remote access. They also offer features like automatic backups and disaster recovery.

Relational databases organize data into tables, where rows represent individual records and columns represent attributes of those records. These tables are connected using keys: primary keys uniquely identify rows in a table, while foreign keys establish relationships between tables. This model is widely used because of its simplicity and ability to handle large datasets. It is especially popular in applications that require structured tabular data, such as customer information

systems or e-commerce platforms. Relational databases can be worked with using SQL, for example, through tools like PgAdmin 4.

Non-relational or NoSQL databases are designed to handle unstructured or semi-structured data, such as JSON files, XML documents, or multimedia content. Unlike relational databases, they do not rely on fixed schemas or tables, which makes them more flexible for applications with diverse or evolving data needs. NoSQL databases, such as MongoDB or Cassandra, are optimized for big data, real-time analytics, or social media platforms. They allow horizontal scaling by distributing data across multiple servers.

According to the text, databases and DBMS make it easier to store, access, and use data. They are important for current life because they help keep information safe, organized, and ready to use.

Aim. Consider relational and non-relational databases, compare them, and choose the best one.

Materials and methods. To research this topic, English-language sources were involved, and methods of information analysis and systematization were used.

Results and discussion. The main difference between relational and non-relational databases is how they organize and store data. Relational databases follow strict rules and focus on strong links between pieces of data. It makes them the best choice for systems that need highly accurate and consistent data, where even small mistakes are not allowed.

Non-relational databases are more flexible. They are good at handling large amounts of unstructured or semi-structured data. Social media platforms often use non-relational databases to manage their data. These databases can adjust to changing data needs, making them a good choice for fast and flexible systems where speed and storage are more important than following strict rules.

Conclusions. In some cases, the hybrid approach that combines different types of databases may be needed to achieve the best performance. This approach allows leveraging the strengths of each database type. To choose the right database for a specific project, it is important to analyze the needs of the system carefully, such as data volume, type, and expected growth. Expert's advice guarantees the right solution of project goals.

REFERENCES

1. GeeksforGeeks. Types of Databases. URL: <https://www.geeksforgeeks.org/types-of-databases/> (date of access: 09.12.2024).
2. Built In. 13 Types of Databases to Know. URL: <https://builtin.com/articles/types-of-databases> (date of access: 09.12.2024).
3. MongoDB. Types of Databases. URL: <https://www.mongodb.com/resources/basics/databases/types> (date of access: 09.12.2024).
4. Software Testing Help. What Is Hybrid Database? List of The Best Hybrid Databases. URL: <https://www.softwaretestinghelp.com/hybrid-database/> (date of acces: 12.12.2024)
5. Equant. Hybrid Database. URL: <https://equant.org/articles/hybrid-database/> (date of acces: 12.12.2024)
6. Coursera. Relational vs. Non-relational Database: The Difference Explained. URL: <https://www.coursera.org/articles/relational-vs-non-relational-database> (date of acces: 20.12.2024)
7. Apriorit. Relational vs Non-relational Databases: Which to Choose for Your Product. URL: <https://www.apriorit.com/dev-blog/relational-vs-non-relational-database> (date of acces: 29.12.2024)