International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 5 Issue 10, October - 2021, Pages: 73-80

Features of Database Types

Zhanna Deineko[1], Svitlana Sotnik[2], Oleksandr Vovk[3], Vyacheslav Lyashenko[3]

[1]Department of Software Engineering, Kharkiv National University of RadioElectronics, Ukraine

[2]Department of Computer-Integrated Technologies, Automation and Mechatronics, Kharkiv National University of RadioElectronics, Ukraine

[3]Department of Media Systems and Technology, Kharkiv National University of Radio Electronics, Kharkiv, Ukraine
e-mail: lyashenko.vyacheslav@gmail.com

Abstract—*Research in field of developing databases and working with modern applications, their implementation in modern enterprises is constantly expanding, so issue of researching database features still requires further attention. This paper discusses functions that typical database management system (DBMS) should provide. The review is devoted to highlighting advantages and disadvantages of existing databases. The article summarizes material on key aspects that should be present in modern DBMS, which will allow you to better understand procedures for building, working and using them***.

Keywords—database; architecture; management; systems.

1. INTRODUCTION

Modern person every day interacts with Internet or with devices with Internet access to obtain necessary information. The volume of information generated by humanity is growing at an alarming rate, and its search is carried out using an information retrieval system (IRS), which not only finds information you need, but also allows you to filter, sort and store large arrays of structured information. Therefore, the IRS fined their application in various fields of science and industry [1]-[3].

The IRS in abstract form should consist of information retrieval language (IRL), rules for switching to this language, and semantic correspondence criterion that determines issuance volume of documents or information.

IRS includes databases (DB) and database management systems.

DBMS allows you to create and store large arrays of text and digital information, as well as create convenient screen forms for dialogue with user.

User interfaces have evolved – green letters on black screen have been replaced by client / server, and then Internet applications.

DB is structured virtual storage of information related to specific subject area. There are several types of databases, based on different approaches to organizing their structure and on different ways of interacting with data stored in them.

Although relational databases have lost their monopoly on flexibility and even scalability, tasks of choosing and using information, ability to accumulate and make decisions, remains significant, since development of rational database for information systems will improve quality and automate work of modern enterprises, which can affect reduction of production costs and an increase in production capacity.

At present, modern enterprises must quickly adapt to changes in global trends and modernize their production lines in connection with outdated production technologies and outdated methods of recording information about products and controlling its quality, therefore this article is devoted to urgent problem today.

2. RELATED WORK

Currently, there are huge numbers of different software products all over world. Most likely, they include databases storing huge amounts of information. In addition, correct selection and construction of database architecture plays an important role [4]. The authors show diagram of modernized architecture, describe approaches to achieving specified key indicators.

An overview of cloud architecture and database architecture is presented in [5]. The authors highlighted advantages and disadvantages of transferring database structures in cloud. Issues related to development of cloud databases are discussed in detail.

Work [6] is devoted to architecture of data streams. The authors focused on databases that use data structures with an irregular flow of control: trees, hash tables, which do not match well with existing database accelerators.

In [7], methods of designing new databases are considered. The authors identified important criteria in database design, analyzed evolution of design methods, their applicability and adaptability in real-life scenarios.

In [8], special attention is paid to types of NewSQL databases. The authors described key features of NewSQL databases.

Databases and their types, information systems are considered in [9].

Database types related to NoSQL (key-value store, document store, column store, and graph store) are presented in [10].

3. Features of database types

Information retrieval is carried out using information retrieval systems.

IRS – a set of software, hardware and other aids, technological processes and functionally defined groups of workers that provide collection, presentation and accumulation of information resources in certain subject area, search and delivery of information necessary to meetinformation needs of established contingent of system users-subscribers [11].

Solving problems associated with information retrieval is of great importance in practice [12], [13].

The rate of growth in volume of information is significantly outstripping progress in improving information retrieval. Therefore, choosing right database model plays fundamental role in creation of information retrieval system.

There are different types of databases that are used to store different types of data (fig. 1).

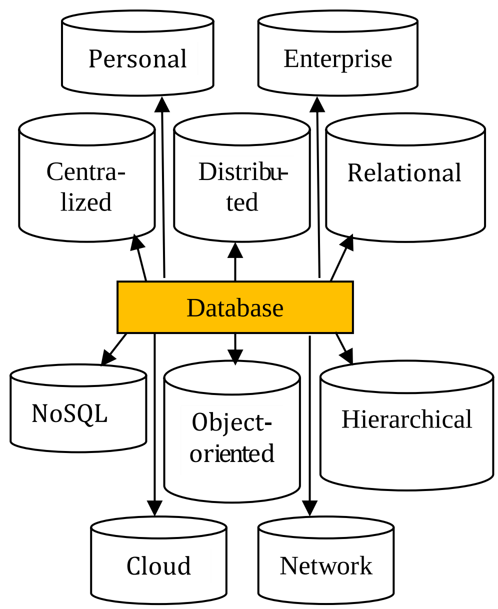


Figure 1. Database types

Let's take quick look at features of each type of database.

Centralized Database (CDB) – Information is stored and maintained in one place, for example, central computer, CPU server, or mainframe. CDB is distinguished by its traditional database architecture (table 1).

Personal (PDB) – database, which is often intended for single user and is focused on solving problems of local user or small group of users (table 2).

Distributed (DDB) – DB, consisting of several, possibly overlapping or overlapping parts of each other, which are located in different nodes of network. Such databases provide more flexible forms of service for many remote users when working with significant amounts of information within corporate networks or structural dissociation.

Distributed databases provide ample opportunities for managing complex multi-level and multi-tier objects and processes. However, there should be single conceptual diagram of entire network – it will provide logical transparency of data for user, as result of which user will be able to form request to entire database, being behind separate terminal. The main criterion for distribution of data in network is as follows – data should be located where there is greatest frequency of access to it. There are three main ideologies: client-server, Web, and distributed objects (DCOM, CORBA). Within each direction, there are also large number of solutions and standards from different manufacturers, for example, Apache Cassandra, HBase, Ignite, etc. (table 3).

Table 1: Features CDB

Advantages CDB	Disadvantages CDB
- minimum costs for adjustments; - provides better data quality, allowing organizations to set data standards; - consistency of data, since data management occurs in central repository.	- need to transfer large data stream; - relatively "long time" of response to data sample, which is due to large volume of database; - low reliability of outdated data transmission channels (communication lines); - difficulty in updating database due to its scale.

Table 2: Features CDB

Advantages PDB	Disadvantages PDB
- relative ease of use, which allows you to create workable user applications on their basis; - ease of updating; - relatively limited requirements for hardware resources, since it is small in size.	- scalability.

Table 3: Features DDB

Advantages DDB	Disadvantages DDB
- failure of one server will not affect entire dataset; - modular development is possible, that is, system can be expanded by including new computers and connecting.	- there is threat of mismatch of data stored in different parts of system; - complex mechanism for managing simultaneous processing is needed, which, in particular, must ensure synchronization when information is updated, this ensures data consistency; - high requirements for reliability and throughput of communication lines.

Enterpris (EDB) – concurrent access by users of large organizations or enterprises is used to manage huge amounts of data (table 4).

Table 4: Features EDB

Advantages EDB	Disadvantages EDB
- ability to manage huge set of data; - supports multiple processes, which allows parallel queries to be executed on system.	- limited compatibility.

Relational (RDB) – is a large number of interconnected tables, each of which contains information about certain objects. Each row in such table contains data on only one object, and columns contain various characteristics of these objects, called attributes [14].

RDB scenarios: accounting, financial and banking systems; inventory management systems; transaction management systems. Examples of relational databases: MySQL, Microsoft SQL Server, Oracle, etc. (table 5).

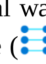
NoSQL – is used to store large number of data sets (data not only in tabular form, but also in several ways), as need to create modern applications has increased. The four most common types of databases NoSQL: key-value storage (); document-oriented database (); graph databases (); wide-column stores (). NoSQL Scripts: Mobile Apps; analytics in real time; content management; personalization; Internet-things applications [9]. Examples of non-relational databases: CouchDB, Couchbase, MongoDB, eXist, Berkeley DB XML (table 6).

Table 5: Features RDB

Advantages RDB	Disadvantages RDB
- simplicity, since there is only one information structure that formalizes tabular presentation of data, which is familiar to users; - data independence, that is, when it is necessary to change structure of relational database, this, as rule, leads to minimal changes in application programs; - security, since authorization is supported in RDB.	- low speed when performing connection operation; - large memory consumption for representation of relational database.

Table 6: Features NoSQL

Advantages NoSQL	Disadvantages NoSQL
- provides fast and flexible application development; - provides high scalability; - provide extremely efficient processing of unstructured data, often at incredibly high speeds; - processing large amounts of data or storing single large database in easily distributed server clusters; - provide easy database migration to cloud for existing NoSQL workloads.	- limited capacity of built-in query language, since almost all query languages and API methods of NoSQL storages were created on basis of certain SQL functions, but they have less functionality; - difficulties in quickly moving from one non-relational database to another; - you have to develop your own tools for working with database; - most companies simply do not have such volumes of data and other specific operating conditions in which NoSQL solutions would be beneficial as main database.

Cloud (CDB) – data is stored in virtual environment and executed on cloud computing platform. It provides users with various cloud computing services (SaaS, PaaS, IaaS, etc.) to access database. Examples of cloud platforms: Amazon Web Services (AWS); Microsoft Azure; Kamatera; PhonixNAP; ScienceSoft; Google Cloud SQL, etc. (table 7).

Table 7: Features CDB

Advantages CDB	Disadvantages CDB
- increased level of flexibility and innovative capabilities; - high security, since cloud DBMS providers operate in protected environments using additional measures (firewalls, antivirus, etc.); - allows scaling of resources directly during operation; - ability to recover data. Many providers provide backup and quick database recovery services. In particular, it is possible to configure storage of up-to-date copy of DBMS in several data centers; - assumes use of multi-node cluster system, which increases speed of data processing.	- complexity of transporting large amounts of data; - dependence on chosen provider (possibility of losing direct access to your information); - when transferring systems unchanged, consumption of cloud resources can be unreasonably expensive.

Object-oriented (OODB) is database in which data is modeled in form of objects, their attributes, methods and classes [15]. OODB is based on number of basic concepts such as object, identity, class, inheritance, overloading, and lazy binding. Any object upon its creation receives unique identifier generated by system, which is associated with object throughout its existence and does not change when state of object changes. Every object has state and behavior. Examples of commercial OODBs: O2, ORION, GemStone, Iris (table 8).

Hierarchical (HDB) – data model in which database is represented in form of tree-like (hierarchical) structure, consisting of objects (data) of various levels [16]. The hierarchical data model is built on principle of hierarchy of object types, that is, one type of object is main one, and rest, located at lower levels of hierarchy, are subordinate.

Table 8: Features OODB

Advantages OODB	Disadvantages OODB
- no need for user-defined keys, since OODB has concept of object identifiers automatically generated by system and guaranteed to be unique for each object; - lightweight design of some connections. OODB maintains an inverse linkage facility for expressing reciprocal links between two objects (binary link). Such system provides referential integrity by establishing appropriate backlink immediately after forward link is created; - allow you to represent complex objects in more direct way than relational systems.	- security concerns, as most OODBs lack authorization; - limited support for integrity constraints, since there are no mechanisms for declaring key properties of attributes (for example, class attribute cannot be declared as primary key of class), or uniqueness constraints, explicit constraints on integrity; - absence of standard query algebra, which also makes it difficult to optimize queries; - lack of support for representations, since development of object-oriented representation mechanism is complicated by such properties of model as identifiable objects.

The data is stored as records linked by links.

Examples of databases with hierarchical model are: Information Management System (IMS); Time-Shared Date Management System (TDMS) etc. (table 9).

Table 9: Features HDB

Advantages HDB	Disadvantages HDB
- simplicity (hierarchical principle of concepts subordination is natural for many tasks); - minimum memory consumption.	- data is accessed only through root relation; - non-universality, since many important variants of data interconnection cannot be realized by means of hierarchical model, or implementation is associated with increase in redundancy in database; - cumbersomeness of such model for processing information with rather complex logical connections; - there may be redundant data; - operations of inclusion and removal become more complicated; - deleting original objects leads to deletion of generated objects.

Network (NDB) – allows you to display various relationships of data items in form of arbitrary graph. A network database consists of records set and related links set. There are no special restrictions on formation of links.

If in hierarchical structures stream record could have only one ancestor record, then in network data model descendant record can have arbitrary number of ancestor records (free parent) [17], [18]. Databases have wide variety of applications in business systems (table 10).

Table 10: Features NDB

Advantages NDB	Disadvantages NDB
- many-to-many relationship, since data HDB does not account for many human organizational events that require entity to have multiple upstream relationships with other entities; - high-speed performance, unlike HDB, flexibility in data storage; - versatility in comparison with other models, as well as ability to access data through values of several relations.	- high complexity and rigidity of database schema (when database structure changes, it leads to restructuring of entire database); - complexity of understanding and performing information processing; - problem of ensuring safety of information in database.

The design of information systems, including databases, is carried out at physical and logical levels.

It has been determined that solution to problems of designing information systems at physical level largely depends on DBMS used and is often automated and also hidden from user.

International Journal of Engineering and Information Systems (IJEAIS)
ISSN: 2643-640X
Vol. 5 Issue 10, October - 2021, Pages: 73-80

In some cases, user is given opportunity to configure individual parameters of system, which is not big problem.

Modern DBMS should provide following types of functions and services (services).

Let's consider one of main functions of DBMS – storing, retrieving and updating data. DBMS should provide users with ability to save, retrieve and update data in database. This is fundamental function of DBMS.

In course of processes analysis of storing, retrieving and updating data in database, it was determined that way these functions are implemented in DBMS should make it possible to hide from end user internal details of physical implementation of system (for example, file organization or storage structures used).

Next, consider directory available to end users.

DBMS must have directory accessible to end users, in which description of data items is stored.

Key feature, for example, of ANSI-SPARC architecture, is presence of integrated system catalog with data about schemas, users, applications, etc. [19].

It is assumed that catalog is accessible to both users and DBMS functions. System catalog, or data dictionary, is repository of information describing data in database (in fact, it is metadata). Depending on type of DBMS used, amount of information and way it is used may vary.

Typically, following information is stored in system catalog [20]:

- names, types and sizes of data items;

- links names;

- integrity support constraints imposed on data;

- names of authorized users who have been granted right to access data;

- external, conceptual and internal schemes and mappings between them;

- statistical data, for example, frequency of transactions and counters of calls to database objects.

The system catalog allows you to achieve certain advantages listed below [20]:

- information about data can be centrally collected and stored, which will allow you to control access to this data, like any other resource;

- you can define meaning of data, which will help other users to understand their purpose;

- message is simplified as precise definitions of meaning of data are preserved. The system catalog can also indicate one or more users who are owners of data or have right to access them;

- due to centralized storage, redundancy and inconsistencies in description of individual data elements can be easily detected;

- changes made to database can be recorded.

- consequences of any changes can be determined even before they are made, since all existing data elements, links established between them, as well as all their users are recorded in system catalog;

- security measures can be further strengthened;

- there are new opportunities for organizing data integrity support;

- an audit of stored information can be performed.

Consider transaction support function.

DBMS must have mechanism that guarantees execution of either all update operations of given transaction, or none of them.

A transaction is collection of actions performed by individual user or application to access or change contents of database. Examples of simple transactions include adding, deleting, or updating information about an object to database. If transaction fails during execution, database is in inconsistent state because some changes have already been made and others have not yet. Therefore, all partial changes must be rolled back to revert database to its previous, consistent state. Features of concurrency management services.

DBMS must have mechanism that guarantees correct database update when many users perform update operations in parallel [20]. At the same time, parallel access is relatively easy to organize if all users only read data, since in this case they cannot interfere with each other. However, when several users simultaneously access database, conflict with undesirable consequences can easily arise, for example, if at least one of them tries to update data.

DBMS must ensure that such conflicts do not occur when many users access database at the same time.

Recovery services. When discussing transaction support, it was mentioned that if transaction fails, database must be returned to consistent state, which must be guaranteed by capabilities of DBMS [14].

Data access control services. DBMS must have mechanism to ensure that only authorized users can access database. The term "security" refers to protecting database from intentional or accidental unauthorized access. It is assumed that DBMS provides mechanisms for such data protection.

Data exchange support. DBMS must be able to integrate with communication software in order to organize access for remote users to centralized database (within framework of distributed processing system).

Data Integrity Support Services. DBMS must have tools to control that data and its changes comply with specified rules. The integrity of database means correctness and consistency of stored data and is expressed in form of restrictions or rules for maintaining consistency of data, which should not be violated in database [20].

Data independence support services. DBMS must have tools to support independence of programs from database structure. Data independence is discussed above and is usually achieved by implementing a view or subcircuit support mechanism. Physical independence from data is fairly easy to achieve, as there are usually several types of allowable changes to physical characteristics of database that do not affect views in any way. However, achieving complete logical independence from data is more difficult. As a rule, system easily adapts to adding new object, attribute, or relationship, but not removing them. In some systems, it is generally forbidden to make any changes to already existing components of logic circuit [19].

4. Conclusion

Research in field of developing databases and working with modern applications, their implementation in modern enterprises is constantly expanding, so issue of researching database features still requires further attention.

Over past decade, number of new solutions and technologies for databases has appeared, corresponding to new types of applications and requirements to which they contribute, main ones are considered and, as result, their features are determined.

Thus, database is organized structure for storing information. The concept of database management system is closely related to concept of database.

This paper discusses functions that typical DBMS should provide (presence of integrated system catalog with data about schemas, users, applications; support for transactions; correct database update and recovery; data access control; support for data exchange and data integrity).

The review is devoted to highlighting advantages and disadvantages of existing databases. The article summarizes material on key aspects that should be present in modern DBMS, which will allow you to better understand procedures for building, working and using them.

5. References

[1] Al-Sherrawi, M. H., & et al.. (2018). Corrosion as a Source of Destruction in Construction. International Journal of Civil Engineering and Technology, 9(5), 306–314.

[2] Khan, A., & et al.. (2015). Some Effect of Chemical Treatment by Ferric Nitrate Salts on the Structure and Morphology of Coir Fibre Composites. Advances in Materials Physics and Chemistry, 5(01), 39-45.

[3] Nevliudov, I., & et al.. (2021). Development of a cyber design modeling declarative Language for cyber physical production systems. J. Math. Comput. Sci., 11(1), 520-542.

[4] Kostyuk, A. I., Bespalov, D.A., & Voloshin, A.V. (2020). Principy sozdaniya arhitektury raspredelennoj bazy dannyh dlya zadachi ohrany perimetra // Vestnik Adygejskogo gosudarstvennogo universiteta. Seriya 4: Estestvenno-matematicheskie i tekhnicheskie nauki, 3 (266), 17-24.

[5] Huang, S. Y., & Teghem, J. (2020). A Review on Cloud and Database Architecture. Fuzzy Systems, 12(2), 29-32.

[6] Vilim, M., Rucker, A., & Olukotun, K. (2021, June). Aurochs: An Architecture for Dataflow Threads. In 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA) (pp. 402-415). IEEE.

[7] Roy-Hubara, N., & Sturm, A. (2020). Design methods for the new database era: a systematic literature review. Software and Systems Modeling, 19(2), 297-312.

[8] Almassabi, A., Bawazeer, O., & Adam, S. (2018). Top NewSQL databases and features classification. International Journal of Database Management Systems, 10(2), 11-31.

[9] Zgambo, I. K. (2020). Databases and information systems. Problemy effektivnogo ispol'zovaniya nauchnogo potenciala obshchestva, 2020(2), 25-27.

[10] Zhao, G., & et al.. (2014). Schema conversion model of SQL database to NoSQL. In 2014 Ninth International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (pp. 355-362). IEEE.

[11] Sotnik, S.V. (2015). Tekhnologiï informacijno-poshukovih system. Kiïv-58, prosp. Kosmonavta Komarova 1, 349.

[12] Baranova, V. & et al.. (2019). Wavelet Coherence as a Tool for Studying of Economic Dynamics in Infocommunication Systems. In 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T) (pp. 336-340). IEEE.

[13] Dadkhah, M., Maliszewski, T., & Lyashenko, V. V. (2016). An approach for preventing the indexing of hijacked journal articles in scientific databases. Behaviour & Information Technology, 35(4), 298-303.

[14] Raut, A. B. (2017). NOSQL database and its comparison with RDBMS. International Journal of Computational Intelligence Research, 13(7), 1645-1651.

[15] Thearle, R. W. (2019). A survey of object oriented database systems. In Object Management (pp. 23-30). Routledge.

[16] Domdouzis, K., Lake, P., & Crowther, P. (2021). Hierarchical Databases. In Concise Guide to Databases (pp. 205-212). Springer, Cham.

[17] Zhou, N. (2020). Database design of regional music characteristic culture resources based on improved neural network in data mining. Personal and Ubiquitous Computing, 24(1), 103-114.

[18] Wang, R. (2021). Optimization of Computer Network Database Security Management Technology in the Era of Big Data. In The International Conference on Cyber Security Intelligence and Analytics (pp. 134-142). Springer, Cham.

[19] Moraru, D. (2018). Arhitectura ANSI-SPARC.

[20] Elmasri, R., & Navathe, S. (2017). Fundamentals of database systems (Vol. 7). Pearson.