CLOUD RELATIONAL DATABASES IN ISLAMIC EDUCATION INSTITUTIONS

Alisher Muhammadiev*

* International Islamic Academy of Uzbekistan,
UZBEKISTAN
Email id: alisherziyo@gmail.com

DOI: 10.5958/2249-7137.2021.02519.2

ABSTRACT

Database Management Systems are observed to operate in distinct environments within organisations. These environments are classified as 'Corporate' and 'Devolved', and different types of database management systems are considered as more appropriate to each. When designing any information system, there are problems of ensuring information security and data protection from unauthorized access to confidential data. The dominant components of information security for cloud information systems (the core of which is a cloud database) are confidentiality, structural stability, and secure access control. The database administrator regulates confidentiality issues and secures access control. However, the problem of ensuring the data structural stability is solved in advance, at the stage of database design. Structural stability is ensured by maintaining the database integrity. Data integrity in cloud information systems guarantees the correct operation of the entire system. The application of an abstract mathematical tool through categories creates a theoretical basis to study data integrity. Data integrity of cloud systems is achieved by solving a variety of problems, among which the dominant ones are to ensure the integrity of domains, the integrity of tables, and referential integrity. The applicability of the methods of category theory, namely, the argumentation of the categorical description of the tasks of maintaining the integrity of domains, the integrity of tables, and referential integrity is presented in the article. The categorical description allows us to represent the cloud information system as a dynamic system. The corporate database is closely associated with an evolutionary model of the entire organisation, whereas each devolved database is a tool used by an individual or group to analyse information necessary to that person's function within the organisation. The requirements for a Restructuring Mechanism to allow a corporate database to be altered in structure to reflect alterations in the organisation and enchancements to the model are identified. Conclusions are drawn that a Restructuring Mechanism is an essential ingredient for a Database Management System, to allow the database to reflect the ever-changing structure of an organisational data model. Areas where future research is likely to be fruitful are identified, and it is suggested that the classification of corporate and devolved will be useful in this respect.

KEYWORDS: *Table Integrity, Morphism, Cloud Information System, Categorical Description, Structural Stability, Data Integrity, Mechanism.*

1. INTRODUCTION

The necessary resources of electronic libraries, machine learning, and artificial intelligence are stored in databases (DB). DB is universal storage of information in all automated information systems. At present, the database is a cloud service of a cloud information system (CIS). The CIS, as one of the last paradigms of distributed data processing, is widely included in our life **[1].**

Cloud information systems provide easy and convenient access to shared computing resources via the Internet. Computing resources are the data transmission networks, servers, storage devices, application programs, service applications, etc. In the field of cloud computing, most service providers not only provide access to cloud computing platforms but also create specialized cloud computing systems that meet technological and the customer's regulatory requirements. The CIS have a number of advantages **[2],** namely, increased flexibility and responsiveness of the system.

Less time is spent on initiating, configuring, tracking, and building cloud environments. It is possible to instantly launch the user's application on any computing hardware infrastructure, and if there is no such virtual server, then you can download a ready-made virtual machine with an installed and configured server from libraries prepared in advance according to user requirements. The accessibility of applications was increased and the continuity of operation of virtual CIS environments was ensured. The use of software setting increases the manageability of the CIS infrastructure. This system reduces the time spent on server administration, provides load balancing and on-line migration of virtual machines. The list of its shortcomings is reflected in **[3].**

The core of the CIS is the database. Currently, there are two most common ways of developing the CIS. The first way: at the initial stage, the basic form of the CIS is developed, and then it is transformed into a cloud service. The second way: first, the type of cloud is formed, and then all developments are implemented in the cloud **[4].** In both cases: in the CIS development and deployment, it is necessary to ensure the integrity of the information in databases. Data integrity means that all data accumulated in the CIS DB are collected in accordance with some predetermined rule and do not contradict each other. In modern databases, a relational database management system is widely used **[5].** The relational model in a relational database has a set of established rules to ensure data integrity. Rules are used to ensure the integrity of information in relational database. Domain constraints, table constraints, and referential constraints can be considered as these rules.

Due to the lack of theoretical argumentation for the data integrity maintenance in the basic structure of the CIS, we propose a categorical description as a universal method of presenting the data integrity maintenance in the CIS.

The relevance of the presented method lies in the fact that fraudsters and hackers attack information resources. Therefore, a colossal number of risks arise in the field of information security. It should be noted that ignoring emerging problems could lead to a loss of competitiveness at the state and corporate levels.

At the same time, not only organizations and enterprises but also ordinary citizens who use gadgets and other means of communication suffer from crimes in the information sphere. The urgency of the threat to the integrity of information and data requires to solve the task of its protection. It is worth noting, that 20 years ago, the task of ensuring information security was solved by using encryption algorithms, installing firewalls, differentiating access, and so on.

Today, these technologies are not enough, and, so, almost any information of a financial, competitive, military or political nature is threatened. Especially if this data is in the cloud. Therefore, there appear new methods of ensuring information security of information systems, among which we can mention the methods given in **[6].**

In particular, the article discusses improved query performance at the stage of generating and analyzing NO-relational data. For this, no clustered indexing and map-reduced data cube numerosity reduction method are used. In **[7],** a hybrid model for assessing information security risks is proposed. The hybrid model is based on threat modeling methodology. Systematic threat analysis allows IT-risk management in the integrated process of information system functioning. In **[8],** to protect accounting information the cryptography methods with block chain technology were used. In articles **[9]** and **[10],** the issues of security of using data in end devices that are clients of cloud services are considered.

The categorical description given in this article presents an innovative method of description of the data integrity maintenance of a cloud information system. The novelty lies in the fact that the basic concepts of parallel algorithms and logical structure can be specified through formalization. In addition, this method has the ability to add new mathematical structures capable of investigating the spatial, structural, and temporal properties of the logical structure of the system.

Theoretical and historical preconditions

Category theory, in general, is a branch of mathematics that studies the properties of the relations between mathematical objects that are not dependent on the internal structure of objects. The theory of categories occupies a key place in modern mathematics, and this section of mathematics is actively used and implemented in computer science, logic, theoretical physics, and other fields of science.

American mathematicians Saunders MacLaine and Samuel Eilenbergare rightfully consideredthe founders of the category theory, which is one of the brightest and most controversial achievements of the 20th century. The theory of categories together with the theory of sets serves as the universal language of modern mathematics. Categories, functors, and their natural transformations are widely used in all mathematical sections as a convenient means of considering various structures in a uniform way and formulating general properties of various structures. The significance of the theory of categories cannot be reduced to the narrow boundaries of the convenience of its expressive possibilities. This theory substantially changed the views on the foundations of mathematics, expanded the boundaries of freethinking. Within the framework of category theory in the 1960s, one of the most ambitious mathematical projects of the 20th century was implemented - the socialization of set-theoretic mathematics. The theory of topos was developed, providing a broad class of categories within which ordinary set theory can be perceived as an ordinary individual theory.

Today, there is a tendency to increase research work on the application of category theory in various fields of natural science, including the study of information technology problems.

The categorical approach was proposed in **[11].** That article investigates the consistency of the theory of homotopy types with the concept of a category. Identity was established between the models of an abstract bundle in the axiomatic homotopy theory with models of object mapping that have a weak ortho-factorization. The study in **[12]** describes the possibility of representing an algebraic model using a polymorphic abstract theory,in particular, the admissibility of representing objects in the form of a polymorphic structure using a categorical description.

In [13] the category theory is applied to solving problems in the field of computer science. A category of algorithms is created with the values of input and output variables. In **[14],** the application of category theory in all areas of modern mathematics is argued. Numerous examples from various branches of mathematics are given where category theory can be applied.

The basic grounds of category theory are described in **[14,15]** and other publications.

2. PROBLEM STATEMENT

We introduce the concept of a category as follows **[16].** Let a set of objects $K_o=\{K_{o_1},K_{o_2},...,K_{o_m},...\}$ be given and K_M be a set of morphisms between objects K_o , i.e. $K_M=\left\{H_{K_o}\left(K_{o_i},K_{o_j}\right)\right\}$,where $H_{K_o}\left(K_{o_i},K_{o_j}\right)$ are the morphisms between elements $K_{o_i},K_{o_j}\in K_o$. Then the class of sets K_o and K_M is called categories of K if the following conditions, presented in table 1,are met:

TABLE 1 - CONDITIONS FOR A CLASS OF SETS

Conditions	$\forall A,B\in K_o\to H_{K_o}(A,B)\in K_M$
	Each morphism from K_M belongs to one and only one of the sets $H_{K_o}(A,B)$
	A private internallaw of composition is introduced in class K_M , i.e. the product $\alpha*\beta$ of morphism $\alpha\in H_{K_o}(A,B)$ by morphism $\beta\in H_{K_o}(C,D)$ is defined if and only if the object B coincides with the object C . In this case $\alpha,\beta\in H_{K_o}(A,D)$. The composition of morphisms is associative: $(\alpha*\beta)*\gamma=\alpha*(\beta*\gamma)$
	Each set $H_{K_o}(A,A)$ contains a morphism 1_A called an identity or unit morphism of object A , such that $\alpha*1_A=\alpha$ and $1_A*\beta=\beta$ for all $\alpha\in H_{K_o}(X,A)$ and $\beta\in H_{K_o}(B,Y)$, where $A,X,Y\in K_o$

The concept of morphism is a more general concept that means the interaction between objects. The morphism, depending on the considered situation, can be a correspondence, mapping, isomorphism, homomorphism, etc.

The purpose of this article is to show the applicability of the category description for solving the problem of data integrity maintenance in the basic structure of the CIS.

3. SOLUTION METHOD

Argumentation of a categorical description of a domain constraint in a relational database. The relational model is based on the concept of "relation" [17]. A relation is defined as a subset of the Cartesian product of domains. A domain is a set of specific relations. The element of relations is characterized by some attribute or property. Mapping is considered as a morphism.

Let us assume that the following domains are specified $D_1, D_2, ..., D_n$. Their Cartesian product D is defined as follows $D = D_1 \times D_2 \times ... \times D_n$; here $D_i = \{d_{i_1}, d_{i_2}, ..., d_{i_n}\}$ and $1 \le i \le n$. $\forall d \in D$ are called a tuple of elements n and are defined as $D = \{d_1, d_2, .., d_n\}$, where $d_i \in D_i, 1 \le i \le n$. Classically, relation R in a Cartesian product D is defined as $R \subset D = D_1 \times D_2 \times ... \times D_n$. That is, a relation is a set of tuples that consist of n elements. In other words, the elements of relations are tuples. The number of elements in a tuple determines the length of relation. If a tuple consists of n elements, then relation R also consists of n elements. Since a relation is a set, it should not have two identical tuples.

Besides, it doesnot matter what order the tuples are in relation. In a relational database, a relation is conveniently represented in the form of a table. The column of this table corresponds to the domain, and the domain is the component of the Cartesian product. Each row is a tuple, the length of which corresponds to the number of domains in the Cartesian product. The table displaying the relation has the properties shown in Figure 1:

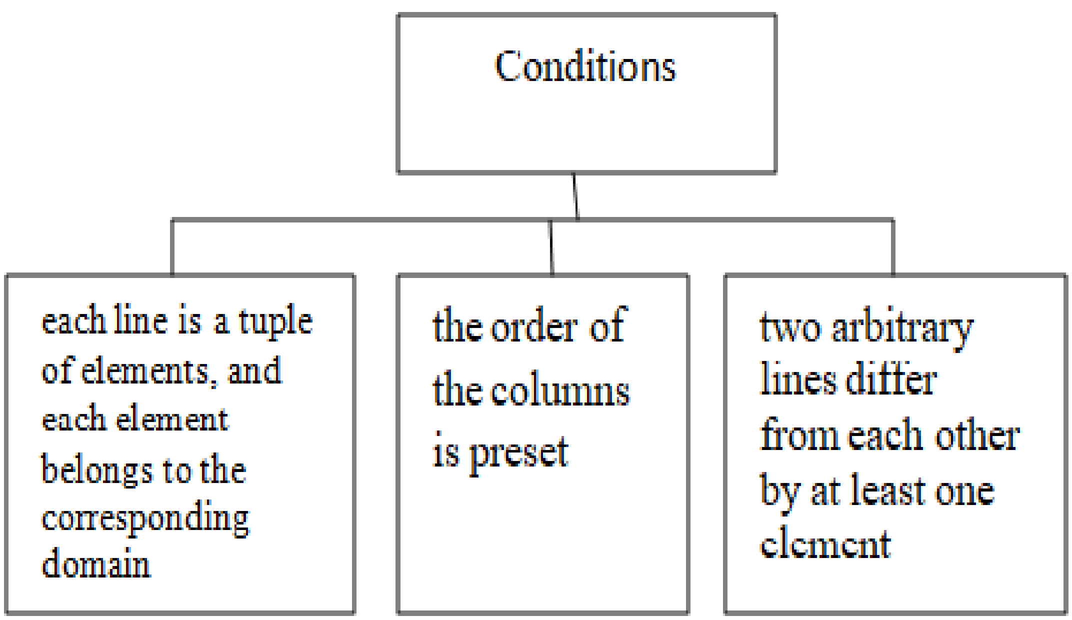

Figure 1. Basic properties of a table displaying relations

Within the meaning of data processing operations, tuples can be processed in any order. Suppose that mapping $f: D \to N$ is given, i.e. mapping f shows the Cartesian product D to the set of natural numbers N. In [10], it was proved that such a mapping is a bijective mapping. Bijectivity of mapping f ensures domain integrity. In addition, for a bijective mapping, there is an inverse mapping f^{-1}, which is also bijective. $f * f^{-1} = 1_A$ is true for f and f^{-1}.

Hence, there is a single mapping for mapping f. Now it remains to show the existence of a composition of mappings. Let $\alpha \in H_{K_o}(A,B)$, $\beta \in H_{K_o}(B,C)$, $\gamma \in H_{K_o}(C,D)$. This means that $A,B,C,D \in K_o$. Then $(\alpha * \beta) * \gamma = \alpha * (\beta * \gamma) : \mathrm{A} \to \mathrm{D}$, or $(\alpha * \beta) * \gamma, \alpha * (\beta * \gamma) \in \mathrm{H}_{K_o}(A,D)$, where $A,B,C,D \in K_o$. Thus, all the conditions of the category are met.

The objectivity of mapping $f: D \to N$ means that there is a primary key that one-to-one identifies the rows of the table. This key allows us to set such an order in the table that its rows are exactly different from each other since the two values of natural numbers do not match.

Ensuring domain integrity means that inter-table links in a relational database are reliable in a well-defined order. Ensuring data integrity in a database is quite a serious and difficult task. The argumentation of the category description means that the methods of category theory can be applied to this direction of research **[16].**

Argumentation of the categorical description of the table constraint. One of the internalconstraintsof data integrity is the integrity of the entire table, i.e. each row of the table must be unique. If this constraint is applied to a table, then each row of the table is uniquely identified.

In order to establish the integrity of the entire table, at the time of its creation, it is necessary to define one column or a group of several columns as the primary key. The unique value of the key field must enter every row of the table.

This means that each row has a unique value of the primary key. If there is a complex key, then the row must have a group of column values. The key field value cannot be NULL. A table can have only one primary key. In many cases, programmers must consider using the primary keys of other tables. To do this, the programmer, when creating the table, enters another alternative or unique key. Alternative and unique keys can be the primary keys in their own tables.

To argue the table integrity in relational databases, we will proceed as follows. Let $S_1, S_2, ..., S_n$ be the columns of a relational table, and $S = S_1 \times S_2 \times ... \times S_n$ be the Cartesian product of these columns. For $g: S \to N$ a mapping presents a Cartesian product on a set of natural numbers. Under these assumptions, the following assertion is true: $(\eta * \mu) * \tau, \eta * (\eta * \tau) \in \mathrm{H}_{K_o}(S^1, S^4)$.

ASSERTION. To ensure the integrity of table values S, mapping g must be bijective.

Proof. Firstly, $\forall s \in S$, $s = \{s_1, s_2, ..., s_n\}, s_i \in S_i, 1 \le i \le n$, $\exists m \in N \Rightarrow m = g(s)$. That is, mapping g is an injective mapping, because the table S consists of a set of tuples, and the corresponding number from the set N can be found for any tuple.

Secondly, $\forall s^1, s^2 \in S$ and $s^1 \ne s^2 \Rightarrow m_1 = g(s^1) \ne m_2 = g(s^2)$.

That is, the mapping is a g -surjective mapping because the table consists of a set of tuples, and two arbitrary tuples differ from each other by at least one element. By the property of relational databases, a table cannot contain rows with matching values.

Since g mapping is injective and surjective, it is bijective. The assertion is proven. An important consequence of this assertion is that a primary key is defined in a table, which is used to establish the order in the table. This established order is responsible for maintaining thetable values integrity.

Further considerations are similar to maintaining the domain integrity. For a bijective mapping, there is an inverse mapping g^{-1}, which is also bijective. $g * g^{-1} = 1_A r$ is true for g and g^{-1}. Hence, there is a single mapping for g mapping. Now it remains to show the existence of a composition of mappings. Let $\eta \in H_{K_o}(S^1, S^2)$, $\mu \in H_{K_o}(S^2, S^3)$, $\tau \in H_{K_o}(S^3, S^4)$, where S^i is the i -th object from the set of objects K_o . It means,that $C = N_1 \times N_2 \times, ..., \times N_n$ $S^1, S^2, S^3, S^4 \in K_o$ .Then $(\eta * \mu) * \tau = \eta * (\eta * \tau): S^1 \to S^4$, or $(\eta * \mu) * \tau, \eta * (\eta * \tau) \in H_{K_o}(S^1, S^4)$, where $S^1, S^2, S^3, S^4 \in K_o$ .These calculations allow us to assert that the methods of category theory are applicable here.

Argumentation for referential constraints maintenance or referentialdata integrity in relational databases. Data integrity is another elementary rule of the standard trust model. The integrity of an association determines the relationship between various columns of a table and a table in a relational database.

This name comes from a reference or match to a column of a column's value or multiple columns. A number of new terms come up when describing referential integrity. The column (or columns) that is linked to another table is called a foreign key. In this case, the column must be a parent key (or primary or unique key) that directs the table to another table.

The foreign key or extension is in the table, and the inherited key is in the inherited table. If the origin and appearance are within the same table, this is called a closed relation (self-reliance).

The argumentation for ensuring referential integrity in a relational database is based on the concept of "relation". Relations are subsets of Cartesian product of domains.

A domain is a set of elements, and elements are added to this set in accordance with predefined properties.

Assume that the following domains $N_1, N_2, ..., N_n$ are given, and their Cartesian product C is defined as $C = N_1 \times N_2 \times, ..., \times N_n$. Here $C_i = \{c_{i_1}, c_{i_2}, ..., c_{i_n}\}$, where $1 \le i \le n$. $\forall c \in C$ is called a tuple of n elements and $c = \{c_1, c_2, ..., c_n\}, c_i \in C_i, 1 \le i \le n$ is true for it.

The concept of relation R is introduced as $R \subset C = C_1 \times C_2 \times ... \times C_n$. That is, a relation is a subsets of Cartesian product of domains and is a set of *n*-dimensional tuples. The number of elements in a tuple determines the number of elements in the relation. In a relational database, a relation is represented as a table. The column in this table corresponds to the domain that is the component of the Cartesian product. Each row is a tuple, the length of which corresponds to the number of domains.

Since a relation is a set, therefore no two identical elements are allowed. We introduce mapping $\delta : C \rightarrow N_1 \times N_2 \times ... \times N_n$. It maps the Cartesian product of domains bythe Cartesian product of natural numbers.

For each N_i , the mapping $\varphi : C_1^i \times C_2^i \times ... \times C_n^i \rightarrow N_i$ must be true, where C_k^i are the domains of the primary table, and N_i are the set of valuesof foreign key. In turn, N_i is a primary key for tables where domains C_k^i participate; here $1 \le k \le n$. From these considerations, it follows that the fidelity of mapping δ is the guarantor of referential integrity in relational database tables.

One of the most important functions of a relational database is the ability to interconnect data from different tables and ensure the application of referential integrity of data in the basic structure of the CIS. A relational database management server efficiently stores data due to the ability to interconnect data from different tables. This option reduces data overflow in database tables. Excessive information and lack of data in tables cause data integrity problems.

Referential integrity ensures that each valueof foreign key matches the inherited key value. Thus, referential integrity not only identifies potential foreign key values but also ensures manipulation integrity when performing operations on the inherited key. For example, suppose a table has two referential integrity constraints.The first referential constraint is a reference to another table that in cascadedeletes that referential integrity through the application. This operation cannot be performed because to delete the row corresponding to the inherited key, all rows associated with the descendant key must be deleted.

The domain mapping of itself meets the requirements of a single morphism. A single mapping does not change the domain structure. The composition of the mappings is defined as follows. If we consider a domain as a set, then for two sets A and B by $H_{K_o}(A,B)$ we mean the set of all mappings A in B , and by composition $\delta * \varphi$ we mean the usual composition of mappings [15]. In this case, K_O and K_M together form a category. So, the categorical approach is also applicable to the study of referential integrity.

4. RESULTS

Using the methods of universal algebra, the applicability of categorical descriptions for solving problems of data integrity maintenance in relational databases was proved. The problem was solved by staged argumentation of the category description of domain constraints, table constraints, referential integrity constraints of data in relational databases.

The argumentation of the categorical description of the domain constraints is based on the assertion about bijectivity of mapping $f : D \rightarrow N$, where $D = D_1 \times D_2 \times ... \times D_n$ and N – is the set of natural numbers. The bijectivity ensures the domain integrity maintenance by one-to-one identifying the rows of the table. In the case of the table restraints, the assertion about the bijectivity of the mapping $g : S \rightarrow N$,is proved, where $S = S_1 \times S_2 \times ... \times S_n$ and N – is the set of natural numbers.

The bijectivity of inverse mapping g^{-1} is also proved. An important result of this assertion is that a one-to-one assertion allows us to compare table rows with a set of natural numbers and to define the primary key forth rows. The primary key forms the order in the rows of the table, and this order is responsible for maintaining the integrity of the table values. At the next stage, the possibility of categorical description of the referential integrity of data is argued. In the process of arguing the categorical description of the referential integrity of data, the relation between the values of the foreign key and the value of the inherited key is shown. This relation acts as a guarantor of the referential integrity maintenance of inter-table links.

Advantages of relational databases. There are plenty of good reasons why relational databases have become the standard in electronic data processing. The following aspects highlight the benefits:

Simple data model: relational databases are based on a data model that is comparatively easy to implement and manage. Plenty of information – like customer data, order lists, or account movements – that companies may want to store long-term can be represented easily using the table structure that the relational database model is based on.

Low data redundancy: the relational database model specifies precisely defined rules for redundancy avoidance with the various normal forms. If normalisation requirements are consistently implemented, relational database systems more or less enable redundancy-free data storage. This simplifies the maintenance and servicing of data, since changes only have to be made in one place.

High data consistency: normalised relational databases enable consistent data storage and so contribute to data consistency. Relational database systems also offer functions that allow integrity conditions to be defined and checked. Transactions that endanger data consistency are excluded.

Quantity-oriented data processing: the relational database system is based on quantity-oriented data processing whereby each entity is broken down into atomic values. This makes it possible to link different entities through their content, as well as complex database queries like JOINs.

Uniform query language: for queries concerning relational databases, the data base language SQL, standardised by a committee from the ISO and IEC, was developed. The purpose of this standardisation is that applications can be developed and executed mostly independently from the underlying database management system. However, support for SQL still varies greatly depending on the DBMS.

Disadvantages of relational databases. Depending on what situation you are using a relational database for, advantages like the simple table-based data model and the distribution of data to several linked tables can also be interpreted as a disadvantage. Furthermore, central features of the relational data model are difficult to reconcile with modern requirements for application programming (like object orientation, multimedia, and big data).

Tabular data display: not all data types can be compressed into the kind of rigid schema required by interconnected two-dimensional tables (impedance mismatch). Abstract data types and unstructured data that occur in connection with multimedia applications and big data solutions cannot be mapped in the relational database model.

No hierarchical database schema: unlike object databases, relational databases offer no option to implement database schemata with hierarchically structured classes. Concepts like subordinate entities that inherit properties from higher-level entities cannot be implementedwith them. For example, you cannot create sub-tuples with them. All tuples in a relational database are on the same hierarchy level.

Data segmentation: the basic principle of a relational database systems of dividing information into separate tables (normalisation) inevitably leads to the data being segmented. Related data is not necessarily stored together. This database design results in complex queries across multiple tables at application level. The resulting high number of queried segments usually also has a negative impact on performance.

Poorer performance compared to NoSQL databases: the relational database model places high demands on data consistency, at the expense of write speed for transactions.

5. CONCLUSION

The categorical approach used in this article to describe the data integrity makes it possible to involve the highest achievement of the theory of functional analysis and universal algebra in solving information security problems. This enriches the tools for studying the structural stability of cloud databases. By strengthening the mathematization of the research apparatus, the rigor of reasoning and logical conclusions is increased. The advantages of this approach are manifested in the fact that the details of the analysis of the possibility of categorical descriptions of data integrity are transferred to the space of universal algebra and research tools become laconic and compact. Furthermore, the categorical approach is a kind of bridge that attracts the latest research achievements of category theory to solving practical problems of ensuring the structural stability of relational databases.

This is especially important when designing the service delivery in the form of a database. Because, unlike other types of distributed information processing, cloud systems are characterized by a high degree of dynamic virtualization and scalability. The categorical approach, unlike other methods, is precisely aimed at studying the dynamic processes occurring within the cloud information system.

In the future, it is planned to develop a working tool for the practical implementation of the solutions presented in the article, using the capabilities of functional programming, and to assess the impact of this approach on the overall performance of the cloud information system.In business areas where **transaction data processing** is at the foreground, relational databases in particular offer numerous advantages. Data on customer campaigns or marketing measures can be ideally mapped in tabular systems. Users also benefit from syntax that enables complex queries despite being relatively simple.

ACKNOWLEDGEMENT

The study was funded by the Ministry of Innovative Development of the Republic of Uzbekistan, a research grant No. И-OT-2019-12 "Development and popularization of an electronic "smart library" platform dedicated to modern scientific and practical interpretation and analysis of the works of our ancestors who made a significant contribution to Islamic studies".

REFERENCES

1. Fingar P. Dot. Cloud: cloud computing - business platform of the XXI century; Moscow: Aquamarine Book, 2011
2. Rivkin M. How Clouds Are Created. Journal, System Discovery: DBMS, 2012;4:22-27.
3. Ullman J. Basics of the database system, Trans. from English by M.R. Kogalovsky and V.V. Kogutovsky, Ed. By M.R. Kogolovsky, Moscow: Finance and Statistics, 1983.
4. Burawis M, Joy R. Query Optimization: Fund Data Generation Applying Non Clustered Indexing and MapReduced Data Cube Numerosity Reduction Method. International Journal of Advanced Trends in Computer Science and Engineering, 2020;9:102-109..
5. Haji S, Tan Q, Soler R. A Hybrid Model for Information Security Risk Assessment. International Journal of Advanced Trends in Computer Science and Engineering, 2019;8:100-106.
6. Krishna ML, Prasanthi, Sambasivarao N. Cryptographic based Message Transfer using Block chain Technology. International Journal of Advanced Trends in Computer Science and Engineering, 2019;8:45-50.
7. Kuyoro S, Ibikunle F, Awodele O. Cloud computing security issues and challenges, International Journal of Computer Networks (IJCN), 2011;3(5): 247– 255.
8. Agrawal R, Bu L, Michel A Kinsy. A post-quantum secure discrete gaussian noise sampler. IEEE International Symposium on Hardware Oriented Security and Trust (HOST). 2020. 23 p. 2020.
9. Fiore M, Hamana M. Multiversal polymorphic algebraic theories: Syntax, semantics, translations, and equational logic. In Proceedings of the 28th Annual ACM/IEEE Symposium on Logic in Computer Science. IEEE Computer Society, 2013. pp. 520–529.
10. Awodey S. Natural models of homotopy type theory, Mathematical Structures in Computer Science, 2018;28(2): 241–286.
11. Barr M, Wells Ch. Category theory for computer science, Reprints in Theory and, Applications of Categories, 2012;(22).
12. Bukur A. Delyanu, Introduction to the theory of categories and functors; Moscow: Mir, 1972.
13. McLain S. Category for the Working Mathematician; Moscow: Fizmatlit, 2004.
14. Awodey S. Category Theory, Oxford university press, 2010.
15. Tursunov N.Kh. Theoretical substantiation of ensuring the domain integrity in the basic structure of cloud information systems, Journal, Science, Technology and Education, Moscow: Problems of Science, 2019;2(55):25-29.
16. Dadamuxamedov A, Mavlyuda X, Turdali J. Cloud technologies in islamic education institutions. ACADEMICIA: An International Multidisciplinary Research Journal, 2020;10(8):542-557.