Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

UDC 004.75
DOI: https://doi.org/10.30837/ITSSI.2020.14.005

O. AVRUNIN, O. VLASOV, V. FILATOV

MODEL OF SEMANTIC INTEGRATION OF INFORMATION SYSTEMS PROPERTIES IN RELAY DATABASE REENGINEERING PROBLEMS

The **subject** of research is methods of semantic integration of subject areas of heterogeneous information systems and distributed databases. This class of systems are created on the basis of database technologies, are widespread and used in all areas of economic activity. Such systems are characterized by high complexity of design, maintenance and modification. The **purpose** of the research is the development of a subject area model based on the semantic properties and relationships of data elements; development of effective technology of integration of information resources of heterogeneous computer systems on the basis of technology of management of database systems; research and formalization of classes of inhomogeneity of data structures aimed at solving the problem of determining the types of information objects. Development of tools for the design and maintenance of application problems for the integration of heterogeneous information systems and distributed databases. **Results**: the analysis of existing methods and models of integration of subject areas on the basis of semantic properties and connections of data elements is carried out; developed an effective mathematical model, technology and algorithm for semantic integration of information resources of heterogeneous computer systems for relational databases; investigated and formalized classes of inhomogeneity of data structures aimed at solving the problem of determining the types of information objects; the model is developed and means of the logical description of properties of information objects for definition of border of the considered subject area are investigated. **Conclusion**: the article considers a formalized infographic model of the subject area, which focuses on the semantic relationships between information objects of databases. An axiomatic approach to the description of the subject area is formulated, which allows to consider the problem of modeling the relations of the elements of the subject area in the form of a set of rules that determine the existence of data elements. On the basis of the analysis of structures and models of databases of information systems the general approach to construction of the universal technology focused on the decision of problems of management of heterogeneous information resources of computer systems is defined.

Keywords: data model; data semantics; database; data integration; heterogeneous information system.

Introduction

Database-based information systems have gone from cumbersome systems organized as shared systems to flexible distributed intelligent information systems. Among the many factors that contribute to such progress are the improvement of the basic tools of database systems - programming and data management systems. This, in turn, is based on the achievement of theoretical research in the field of data modeling, methods of designing logical and physical structure, non-procedural data processing languages.

Research conducted by the authors of the article is aimed at creating systems for integrating and managing information resources of distributed computing systems. Integration means the management of heterogeneous information, which will allow organizing access to heterogeneous data contained in the generated structures data files and databases.

The solution to the problem of integrating heterogeneous information resources begins with attempts to integrate heterogeneous databases (DB). The direction of integrated or federated heterogeneous information systems appeared in connection with the need to share data based on different models and managed by different database management systems (DBMS). One of the options for solving the problem of integrating heterogeneous databases is to provide users with the ability to see the global schema of the domain. A global schema view is usually implemented in some data model, and supports automatic conversion of global data manipulation statements to statements understood by the corresponding local DBMS. With the strict integration of heterogeneous data, local systems lose their autonomy. Since users of information systems often do not agree to lose local autonomy, nevertheless wanting to be able to work with all local DBMS in one language and formulate queries with simultaneous indication of different local databases, recently much attention has been paid to research in the field of multi-databases. Systems of this class do not support the integrated database global schema, and special methods are used to access objects of local systems. As a rule, in this case, only data sampling is allowed at the global level, which allows you to maintain their autonomy.

As a rule, it is necessary to integrate heterogeneous data distributed in a computer network. This makes implementation much more difficult. In addition to its own integration problems, it is necessary to solve all the problems inherent in distributed DBMS: global transaction management, network query optimization, etc. For the external presentation of integrated and multi-databases, the relational data model is most often used. Therefore, the inclusion of a local relational DBMS into an integrated system is much easier and more efficient than the inclusion of a DBMS based on another data model.

Features of solving the problem of semantic integration

Among the reasons leading to the disagreement of information resources are the following:

1. *Heterogeneity, distribution and autonomy of information resources of the system*. The heterogeneity of resources can be syntactic or semantic (either different types of semantic rules are used, or different aspects of the domain are detailed and / or aggregated). A purely

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)

realizable heterogeneity of information resources is also possible, due to the use of different computer platforms, operating systems, database management systems, programming systems, etc.

2. *Needs for the integration of information system components*. Obviously, the most natural way to organize a complex information system is its hierarchically nested construction. More complex function-oriented components are built from simpler components that could be designed and developed independently, which creates heterogeneity.

3. *System reengineering*. After the creation of the initial version of the information system, the process of its continuous alterations inevitably follows, due to the development and change of the corresponding business processes.

4. *Solving the problem of legacy systems*. Over time, any computer system becomes an object of attention for the organization that operates it, since it constantly has to solve the problem of embedding outdated information components into a system based on new technology and solving new problems.

5. *Extension of the life cycle of the information system*. The longer the information system functions, the more needs arise to change and / or add components designed and developed to meet new challenges.

There is no problem if from the very beginning the information system is designed and developed as an open system, when all components are interoperable. Unfortunately, in practice, such an implementation is difficult to achieve. For various reasons, there are needs for the integration of independently and differently organized information and computing resources.

The development of views on information resources is their representation in the form of a set of typed objects that combine the ability to preserve information content (their state) and information processing due to the presence of certain methods applicable to the object.

The main conclusion from the above analysis is that the problems of integrating heterogeneous information resources are relevant when considering the functioning of information resources, and at the same time it is required to use reasonable combinations of architectural, information and organizational solutions.

General approach to semantic modeling in data integration

By a data model we mean a formalized representation that allows you to implement data interpretation in accordance with the specified requirements. The concept of a model is closely related to the concept of abstraction. The abstraction of a system is a model of this system, in which some details are deliberately omitted [1].

The most general and rigorous concept of a model is defined in mathematics. By the model we mean the basic set of objects O and the set of relations D on O. Thus, the model M can be represented by the pair

$$M =< O, D > . \quad (1)$$

In contrast to a mathematical model, models in information technology must include not only structural but also operational specification. For modeling a subject area, the most acceptable abstraction is algebraic systems that combine, in addition to a set of objects O and relations between objects D, also a set of operations (functions) Ω defined on the main set O. In this case, the model views can be specified by three elements

$$M =< O, D, \Omega > . \quad (2)$$

To formalize the presentation of databases, the term "model" is usually used, implying a triplet $< O, D, \Omega >$ given the definitions introduced. Although, for a more detailed presentation of data semantics, this model will include rules describing possible states of the domain. It is not possible to present a database in a strict mathematical form, and even more so, some fixed subject area is not possible due to the severity of abstractions in the description of the mathematical model. That is, in a mathematical model it is impossible to express the meaning of objects from R, D and Ω. In reality, when designing, it is possible to use several types of models. Each model is defined by different types of relationships between objects of the subject area [2].

One way to establish links between objects is to categorize them. Objects of the same category are considered similar, and the similarity characteristics are usually specified by the category properties. In accordance with the level of requirements for data categorization, models are divided into two types: strongly typed - in which it is assumed that all objects should be assigned to a category; weakly typed – not bound by any assumptions about categories [3–4]. For example, in the *TEACHER* category, with a strongly typed model, all objects must have the same type of structure, which in this case cannot be true, since full-time, part-time workers, payroll, etc.

Unlike strongly typed models, weakly typed models provide data and category integration. It is convenient to provide the realization of such possibilities using the predicate calculus. Many models use predicate calculus to represent knowledge that is not implemented by the underlying means of the model. Modeling using predicate calculus assumes working with linear texts, and can be written both in the usual mathematical notation and in programming languages such as PROLOG or DATALOG. Thus, predicate calculus is not overly complex and poorly structured. With this approach, the emphasis is placed on ensuring the universality of the description tools without regard to artificial restrictions on the typing and categorization of data. The model defines the rules according to which the data is structured. However, structural specifications do not provide a way to fully interpret the semantics of the data and how it is used. Operations on objects and data must also be defined. For example, the objects of the model, depending on the allowed operations, can be added, removed or changed, and also, using operations on data, the values of objects that are not explicitly specified in the model can be obtained [5].

Considering the domain model, we define the properties that it displays. Let's distinguish two classes of

Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14) *ISSN 2522-9818 (print)* *ISSN 2524-2296 (online)*

properties: static and dynamic. Static properties include properties that are invariant in time, they are always valid and unchanged. Dynamic properties are characterized by possible changes in the subject area. Any model must represent these two classes in some way. We will assume that the set of objects is determined by the requirements of the subject area. The choice of acceptable implementations of objects or links between them is set by specifying restrictions in the form of a set of rules that determine the dependencies between objects and possible extensions of the domain, that is, a set of output (calculated) objects.

Using the previously introduced designations, we define a model M as a set of objects O, a set of rules L, and a set of operations Ω. The rules will be set in the form of implications (the symbol "$\leftarrow$" is read as "if-then")

$$O \leftarrow O_1, O_2, ..., O_n, \quad (3)$$

where O_i are domain objects of $i = 1 \div n$.

Rules (3) determine that if all objects O_i are included in the set O, then O must also be included in O. The rules L, generating additional objects O_i, with the help of operations Ω (in this case, the operation of adding) specify an extended set of objects S, including both specified and derived objects. Thus, the model can be defined as

$$M =< O, L, S, \Omega > . \quad (4)$$

Definition. The database $DB(O, L)$ is a set of objects O reflecting the properties of the subject area and a set of integrity L constraints that determine the acceptable state of the database. In this case, the operating specification is determined by the data model used in the development of the database structure. The extension of the carrier O built according to the given rules L will be called the semantics of the database in the notation S. Two databases are equivalent if the rules L_1 and L_2 determine a one-to-one correspondence between semantics S_1 and S_2.

Example 1. Let a set of objects be given:

$O =$ (*Full name, Subject, Grade), and a set of rules,*

$L =$ (*Average mark* $\leftarrow$ *Full name, Assessment; Number of passed exams* $\leftarrow$ *Full name, Subject*).

Then, taking into account the fact that the presence in the S an object "*Average mark*" depends on the presence in O the objects "*Full name*", "*Grade*", and "*Number of passed exams*" depends on the presence in O the objects "*Full name*" and "*Subject*", you can build an extension:

$S =$ (*Full name, Subject, Grade, Average mark, Number of passed exams*), reflecting all possible objects, both static and obtained as a result of performing operations on a set O. Obviously, to assess the information state of the subject area, it is necessary to analyze the set S.

In the considered context of the representation of the data model, we will define how a set of objects in the domain O – let's call it a carrier, a set of rules L – integrity constraints. In this case, such restrictions are understood as the semantic properties of the carrier, which adequately reflect the dependencies between the objects of the subject area.

Example 2. Let the database DB_1 be defined by the set $O_1 =$ (*Full name, Subject Grade*) and a set of rules:

$L_1 =$ (*Average mark* $\leftarrow$ *Full name, Grade; Number of passed exams* $\leftarrow$ *Full name, Subject*),

and DB_2 is defined by the set $O_2 =$ (*Full name, Mathematics, Physics, History*) and a set of rules:

$L_2 =$ (*Average mark* $\leftarrow$ *Full name, Mathematics; Average mark* $\leftarrow$ *Full name, Physics; Average mark* $\leftarrow$ *Full name, History; Number of passed exams* $\leftarrow$ *Full name, Mathematics, Physics, History*).

Accordingly, the extensions for DB_1 and DB_2 will look like

$S_1 =$ (*Full name, Subject, Grade, Average mark, Number of passed exams*),

And accordingly

$S_2 =$ (*Full name, Mathematics, Physics, History, Average mark, Number of passed exams*).

It is intuitively clear that for DB_1 and DB_2 can be reflected the same information if you detail some objects, in particular, clarify what is included in the "*Subject*", what is a set of objects "*Mathematics*," "*Physics*", "*History*" and how to get the "*Grade*" object.

If in the rules for DB_1 and DB_2 add the rules from $L_1^{'}$ и $L_2^{'}$.

$L_1^{'} =$ (*Mathematics, Mathematics, Physics, History* $\leftarrow$ *Subject*),

$L_2^{'} =$ (*Subject* $\leftarrow$ *Mathematics, Physics, History; Grade* $\leftarrow$ *Full name, Mathematics; Grade* $\leftarrow$ *Full name, Physics; Grade* $\leftarrow$ *Full name, History*), then the extensions S_1 и S_2 would be the same, that is,

$S_1 = S_2 =$ (*Full name, Mathematics, Mathematics, Physics, History, Subject, Grade, Average mark, Number of passed exams*).

Thus, having carried out the decomposition in the first case, the "*Subject*" object, and in the second, generalized the "*Mathematics, Physics, History*" object, performing the aggregation of the "*Grade*" object, using the appropriate rules $L_1^{'}$ and $L_2^{'}$ for this, we come to the conclusion about the equivalence of DB_1 and DB_2. In what follows, we will require that, when defining a set L, all objects of the subject area be detailed by the rules of generalization, decomposition or aggregation [6–7].

Research and development of a model for integrating subject areas of information systems

In the general case, generalization can be represented as an abstraction, in which a set of objects with common semantic properties is considered as one generalized object. In turn, decomposition is an abstraction in which one object can be replaced by a set of independent objects,

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)

the totality of which expresses the semantics of the original object. In addition, an abstraction of the aggregation type is possible, in which an object is constructed from other objects and represents a semantic refinement or extension of the original object. On the other hand, an aggregate can act as an object that connects other objects, the semantic individuality of which in the considered subject area is not obvious [8–10].

Considering such an abstraction, many individual differences between objects can be ignored. So in the example, a lot of names of objects can be abstracted as a generalized object "*Subject*". This abstraction neglects individual differences between subjects, such as the fact that subjects have different names, are read by different teachers and listened to by different students. In turn, "*Grade*" is an aggregated object that includes semantically identical values for pairs of objects of the type "*Full Name* ↔ *Mathematics*", "*Full Name* ↔ *Physics*", "*Full Name* ↔ *History*", which is expressed by additional rules (extension rules).

The correctness of operations in the analysis and construction of the semantics of the database is determined by the assumption that each representation of a domain element and a set of requirements are complete and consistent [11-13]. This means that all objects are defined and that no additional detail is required within the views. Thus, the following conditions must be met:

1. The set of objects is complete in terms of the requirements of the subject area.

2. All objects have unique names (exclusion of homonymy).

3. Generalization of identical objects is not required (exception of synonymy).

Data research has mainly dealt only with aggregation (for example, Codd normal forms), and generalization has been largely ignored. The reason was that in simple models, generalization could be dispensed with by choosing a specific approach each time that was appropriate for a given case. Artificial intelligence research on knowledge bases, by contrast, has mainly dealt with generalization (e.g. Quillian's semantic networks), while aggregation has not been used. The opposite abstraction of decomposition was not considered at all when describing data structures [14].

The combination of the principles of generalization and aggregation decomposition can extend the data representation model using the methods used in artificial intelligence [15, 16].

When analyzing the structure of a database schema, it is essential to be able to explicitly represent the types of abstractions. This allows you to ensure that the naming conventions for objects are consistent with the database media. In particular, explicit naming of objects provides the following capabilities:

- applying operations to modified objects;
- replace a set of objects with a generalized or decomposed representation;
- specify the specification of links between objects

Let's go back to the extension rules from $L_1^{'}$ and $L_2^{'}$. As a result of applying these rules, objects that were not explicitly specified, but actually present in the subject area ("*Subject*", "*Grade*", "*Mathematics*", "*Physics*", "*History*") were included in the semantics. Objects included in the data carrier, as well as those obtained as a result of object detailing (generalization, decomposition, aggregation) and constituent elements of semantics are called extensional objects.

In addition to the rules expressing generalized, decomposed, and aggregated objects, when describing a subject area, rules can be specified that define a set of calculated objects. Such objects are initially absent in the medium and are formed as a result of performing final functions, using simple or complex arithmetic operations, logical values, etc. [18–21]. An example of obtaining calculated objects is discussed above.

So the set L contains the rules "*Average score* ← *Full name, Grade*", "*Number of passed exams* ← *Full name, Subject*", while the objects "*Average score*" and "*Number of passed exams*" obviously cannot be present in the DB extension, but they can be obtained as a result of performing arithmetic operations on the values of the objects "*Full Name*", "*Grade*", "*Subject*". Objects that are not explicitly specified in the database media, but obtained (calculated) based on the rules for expanding the database and constituting the elements of the database semantics, are called intensional objects. Thus, when obtaining the semantics of the database, it is necessary to take into account two types of rules: generating extensional and intensional objects. For greater detail of the subject area, the rules that generate various types of objects will be divided into two sets L^{ext} and L^{int} extensional and intensional, respectively [22].

It should be noted that if the division of rules into extensional and intensional is of practical importance, namely, during their formation, different construction logic is used, then the general inference logic is used in the construction of semantics, after which the operation of combining elements S^{ext} and S^{int} is performed. The division of semantics into two types enhances the clarity of the process and in some special cases may be of practical importance.

Analyzing the set S, one can draw attention to the obvious redundancy of objects, that is, the simultaneous presence in syntactically different S objects expressing the same semantics, thus, the third condition for the adequacy of the representation of the semantics of the subject area is violated, namely the emergence of synonymy.

Synonymy can arise mainly in the construction of extensional semantics, since it is during generalization, decomposition or aggregation that semantically unambiguous objects can arise, and synonymy can be between one object and the union of many objects. For example, the "*Supplier*" object expresses the same as the "*Address*" ∪ "*Name*" objects.

To exclude such a situation, when forming the rules for generalization, decomposition and aggregation, we will add compensating (excluding) rules of the form {¬ *Supplier* ←*Address, Name*}. (Here the symbol "¬" denotes the absence of an element in the set.) Moreover, it is obvious that the simultaneous presence of mutually

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14) ISSN 2524-2296 (online)

exclusive objects violates the logic of representing the subject area.

In this case, it is possible to construct an intensional semantics S^{int} for any variant of extensional semantics S^{ext} according to the given intensional rules L^{int}. This fact indicates the need to consider separately two types of rules when building a domain model – extensional and intensional rules. In turn, the general semantics S in the model should reflect one state of the subject area at a particular moment in time. Although during the operation of the system the ability to dynamically change S depending on changes in the requirements of the subject area or users must be taken into account, this issue is resolved at the stage of managing the database.

Thus, based on the introduced notation, the representation of model (4) will be written as

$$M = (O, L^{ext}, L^{\text{int}}, \Omega, S). \quad (5)$$

To confirm the correctness of the considered examples, describe the subject area, we describe the formal modeling apparatus based on the logic of first-order predicates.

Formal model for representing data semantics

Declarative specifications, formulas of propositional calculus, or first-order predicate calculus can be used as a means of defining the structural component. Data objects that meet the specified conditions constitute the valid state of the database [23–25].

We will consider a database as a set of predicates. In this case, the predicate will be considered as a functional statement. In contrast to arithmetic and logical functions, where the range of values and the range of changes in type arguments is the same, that is, homogeneous, the range of values of a function for predicates is logical, and the range of changes of arguments is subject. Thus, the predicate is a non-homogeneous function and can be used to simulate [26].

In predicate logic, an atomic formula is an elementary object with a truth value. An atomic formula consists of a symbolic notation for a predicate and a term. In general, the predicate can be represented as a formula $p(t)$, where p is the designation of the predicate, and t is the term. The number of terms determines the dimension of the predicate, that is, in this case, the predicate p is unary. Essentially, a predicate is a function that returns a Boolean value, true or false, depending on the value of a term.

In the context of database theory, the predicate will be considered as an information component that reflects the value of the corresponding object. In other words, if A is some data object, then

$$p(t) = \begin{cases} True, t \in A \\ False, t \notin A \end{cases}. \quad (6)$$

This representation can be used to describe the semantics of data. For example, for a relational model, when the structure of an information component (table) is defined by a relation of the form

$$\rho = DomA_1 \times DomA_2 \times ... \times DomA_n, \quad (7)$$

where $DomA_i$ – set of valid attribute A_i values or attribute domain $A_i (i = 1 \div n)$, ρ represents a set n of tuples in O (A_1, A_2, ..., A_n) media expressing the semantics of the database.

In this case, the predicate model represents the conjunction of a finite set of predicates corresponding to the relation scheme of a relational database presented in the form

$$P = p_1(t_1) \& p_2(t_2) \& ... \& p_n(t_n), \quad (8)$$

where $p_i(t_i)$ is a predicate corresponding to property (6) and $1 \le i \le n$, P – a set of objects expressing data semantics. Then the support can be represented as a set of unary predicates:

$$O(p_1(t_1) \& p_2(t_2) \& ... \& p_n(t_n)), \quad (9)$$

where predicate $p_i(t_i)$ corresponds to property (6), $1 \le i \le n$ and displays the values of the corresponding objects of the subject area. Let's fix some alphabet ϑ containing constants, variables and predicates For a unary predicate p, a formula $p(t)$ will be called a positive literal l, and a formula $\neg p(t)$ a negative literal $\neg l$. A base literal is a positive or negative literal that does not contain variables. Thus, the set (9) will represent the extension, and will be written as

$$O(l_i, l_2, ..., l_n,). \quad (10)$$

The semantics of the database will be determined by a set of rules of the form

$$L = \{l \leftarrow l_i, l_2, ..., l_m,), \quad (11)$$

where $l \leftarrow l_i, l_2, ..., l_m$ are literals and $m \ge 1$.

The rule can be read as the expression " if $l_i, l_2, ..., l_m$ is executed, then l' is executed and expresses the intensional properties of the data. The condition for allowing objects in semantics S is that if all literals $l_i, l_2, ..., l_m$ are included in O, then l can be included in S. If this condition is not met and the literal being defined as l is included in S without defining literals $l_i, l_2, ..., l_m$, then data consistency may be violated. The main condition for correctness is the compatibility of objects in S. Compatibility consists in the absence of the same positive and negative literal.

Thus, for the predicate model, we will also consider two types of rules defining extensional L^{ext} and intensional L^{int}, and the general set of rules, respectively, as $L = L^{\text{int}} \cup L^{ext}$. It is assumed that the elements O may change depending on the data requirements, and it is necessary to adjust the rules describing the subject area, no matter what the relationships between objects, their details, and possible final operations are. However, the data semantics should automatically change in accordance with changes in O and L. Modification of a set O is

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)

defined by the operation of adding or deleting a literal l, at execution of which S remains the same.

In other words, adding a literal means that l must be present in the semantics of the modified database, and deleting means that l should not be included in the semantics of the modified database, and the simultaneous presence of positive and negative literals is not allowed [27–30].

Algorithm for calculating database semantics

Based on the introduced concepts and assumptions, we can conclude that the information content O can be expanded in accordance with its semantics S by means of the given rules L. Thus, the problem of calculating semantics S arises, and S must be calculated with each change of O elements. Let's consider the calculation algorithm S.

Algorithm:

Input data: Database $DB(O, L)$.

Output data: database semantics S.

Method: calculate the literal sequence S according to the following rules.

1. S^0 is O.

2. S^{i+1} is S^i plus a set of literals l_i such as in L there is a certain rule $l \leftarrow l_i$ и $l \in S^i$. As $S = S^0 \subseteq ... \subseteq S^i \subseteq ... \subseteq A$ and A of course, finally there will be achieved such i, that $S^i = S^{i+1}$. That is $S^i = S^{i+1} = S^{i+2} = ...$.

3. Thus, there is no need to perform calculations after S^i, if $S^i = S^{i+1}$.

Algorithm is finished.

Let's consider an example illustrating the above algorithm.

Let $O = \{A, B\}$ and $L = \{A, B \leftarrow C;\ C \leftarrow A;\ B, C \leftarrow D:\ A, C, D \leftarrow B;$ $D \leftarrow E, G;\ B, E \leftarrow C;\ C, G \leftarrow B, D;\ C, E \leftarrow A, G\}$.

Consider $S^0 = A, B$. For calculating S^1 find rules that have on the right side either separately literals A or B, or a pair A, B together. There are two such rules $C \leftarrow A$ and $A, C, D \leftarrow B$. Attaching literals C and D to S^0 and we believe that $S^1 = A, B, C, D$.

To calculate S^2, look for the right-hand sides of the rules contained in S^1. We find $C, G \leftarrow B, D$, thus $S^2 = A, B, C, D, G$. To calculate S^3, we are looking for rules in which the right-hand sides contain literals A, B, C, D, G either separately or together. These requirements are met by the rules $C, E \leftarrow A, G$.

Thus, we have $S^3 = A, B, C, D, E, G$ – the set of all literals. Therefore, further computation will not change the semantics, since $S^3 = S^4 = ... = S$. As a result, the semantics of the original database $DB(O, L)$ corresponds $S = \{A, B, C, E, G\}$.

Let two databases $DB_1(O_1, L_1)$ and $DB_2(O_2, L_2)$ are given. We will say that DB_1 and DB_2 are equivalent (in designation $DB_1 \equiv DB_2$) if their semantics $S_1 = S_2 S$ are equal. To check the equivalence, it is necessary for each rule $X \leftarrow Y$ from the set of rules L_1 to check whether the left parts of these rules are contained in S_2; in this case, algorithm 1 can be used to calculate the semantics. If it turns out that some literals in L_1 do not belong S_2, then obviously $S_1 \neq S_2$. If every literal from the left side L_1 belongs to S_2, then every literal from S_1 will also belong S_2, and if the converse statement is also true, then the statement $S_1 = S_2$ is also true.

Thus, when comparing two databases, it is necessary to compare their semantics. Moreover, if its semantics did not change during modification, then we can assume that the information content also remained unchanged.

Conclusions

The article deals with a formalized infological model of the subject area, which is focused on semantic relations between information objects of databases. The main components of the domain model are highlighted and formal definitions are given to the basic entities: a set of information objects, the relationship between information objects (rules of logical existence), a support system for structural and information integrity. An axiomatic approach to the description of the subject area is formulated, which allows us to consider the problem of modeling the relations of the elements of the subject area in the form of a set of rules that determine the existence of data elements. Based on the analysis of structures and models of information systems databases, a general approach to the construction of a universal technology focused on solving problems of managing heterogeneous information resources of computing systems is determined.

References

1. Cycritis, D., Lokhovsky, F. (1985), *Data Models* : Trans. from English, Moscow, Finance and Statistics, 344 p.
2. Maltsev, A. I. (1970), *Algebraic systems*, Moscow, Nauka, 392 p.
3. Martin, J. (1980), *Database Organization in Computing Systems* :Tr. from English, Moscow, Mir, 662 p.
4. Buslik, M. M. (1993), *Optimal image of a real database* : Monograph, Kyiv, ISDO, 84 p.
5. Date, K. (2001), *Introduction to database systems* : trans. from English, Moscow, Publishing House "Williams", 1072 p.
6. Tanyanskiy, S. (2005), "Semanticheskaya model' predmetnoyoblasti v zadachakh integratsii neodnorodnykh informatsionnykh sistem", *Vestnik Khersonskogo Natsional'nogo tekhnicheskogo universiteta*, Kherson, No 1 (21), P. 52–59.
7. Yesin, V. I. (2012), "Reinzhynirynh isnuyuchykh baz danykh", *Systemy obrobkyynformatsyy*, KHNU im. V. N. Karazina, Kharkiv, Vol. 2, No. 3 (101), P. 188–191.

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2020. № 4 (14) *ISSN 2524-2296 (online)*

8. Avrunin, O. G., Bodianskyi, Ye. V., Kalashnyk, M. V., Semenets, V. V., Filatov, V. O. (2018), *Suchasni intelektualni tekhnolohii funktsionalnoi medychnoi diahnostyky*, KhNURE, Kharkiv, 236 p. DOI: 10.30837/978-966-659-236-4.
9. Bermúdez Ruiz, F. J., García Molina, J., Díaz García, O. (2017), "On the application of model-driven engineering in data reengineering", *Information Systems*, No. 72, P. 136–160. DOI: 10.1016/j.is.2017.10.004.
10. Blackstock, S., Salami, B., Cummings, G. G. (2018), "Organisational antecedents, policy and horizontal violence among nurses: An integrative review", *Journal of Nursing Management*, No. 26 (8), P. 972–991. DOI: 10.1111/jonm.12623.
11. Filatov, V., Rudenko, D. Grinyova, E. (2014), "Means of integration of heterogeneous data corporate information and telecommunication systems", *Proceedings of the 24th International Crimean Conference Microwave and Telecommunication Technology (CriMiCo-2014), 7-13 sept. 2014*, Sevastopol, Ukraine, P. 399–400.
12. Fioravanti, S., Mattolini, S., Patara, F., Vicario, E. (2016), "Experimental performance evaluation of different data models for a reflection software architecture over NoSQL persistence layers", *Paper presented at the ICPE 2016 - Proceedings of the 7th ACM/SPEC International Conference on Performance Engineering*, P. 297–308. DOI: 10.1145/2851553.2851561.
13. Glava, M., Malakhov, V. (2018), "Information Systems Reengineering Approach Based on the Model of Information Systems Domains", *International Journal of Software Engineering and Computer Systems (IJSECS)*, University Malaysia Pahang, Vol. 4, P. 95–105. DOI: 10.15282/ijsecs.4.1.2018.8.0041.
14. Tsalenko, M. Sh. (1989), *Modeling semantics in databases*, Moscow, Nauka, 288 p.
15. Filatov, V., Radchenko, V. (2015), "Reengineering relational database on analysis functional dependent attribute", *Proceedings of the X Intern. Scient. and Techn. Conf. "Computer Science & Information Technologies" (CSIT'2015), 14-17 sept. 2015*, Lviv, Ukraine, P. 85–88.
16. Kosenko, V. (2017), "Principles and structure of the methodology of risk-adaptive management of parameters of information and telecommunication networks of critical application systems", *Innovative Technologies and Scientific Solutions for Industries*, No. 1 (1), P. 46–52. DOI: https://doi.org/10.30837/2522-9818.2017.1.046.
17. Korneev, V. V., Gareev, A. F., Vasyutin, S. V., Reich, V. V. (2001), *Database, Intellectual information processing*, 2nd ed., Moscow, Nolidge, 496 p.
18. Dubois, D., Prades, A. (1990), *Theory of opportunities. Applications to the representation of knowledge in computer science*, Moscow, Radio and communication, 288 p.
19. Filatov, V., Semenets, V. (2018), "Methods for Synthesis of Relational Data Model in Information Systems Reengineering Problems", *Proceedings of the International Scientific-Practical Conference "Problems of Infocommunications. Science and Technology" (PIC S&T-2018), 9-12 oct. 2018*, Kharkiv, Ukraine, P. 247–251.
20. Filatov, V., Kovalenko, A. (2020), *Fuzzy systems in data mining tasks*. DOI: 10.1007/978-3-030-35480-0_6.
21. Sichkarenko, V. A. (2002), *SQL 99 Database Developer Guide*, Moscow, DiaSoftUP, 816 p.
22. Filatov, V. (2014), "Fuzzy models presentation and realization by mean sof relational systems", *Econtechmod:aninternational quarterly journal on economics in technology, new technologies and modelling processes*, Lublin, Rzeszow, Vol. 3, No. 3, P. 99–102.
23. Filatov, V. A. (2004), *Mul'tiagentnyye tekhnologii integratsii geterogennykh in formatsionnykh system I raspredelennykh baz dannykh*, Dis. d-ratekhn. nauk: 05.13.06, KHNURE, Kharkiv, Ukraine, 341 p.
24. Rumbaugh, J., Blaha, M. (1991), *Object-Oriented Modeling and Design*, N. J., Prentice Hall, 348 p.
25. Maatuk, A. M., Ali, M. A., Aljawarneh, S. (2015), "An algorithm for constructing XML schema documents from relational databases", *Paper presented at the ACM International Conference Proceeding Series, 24-26 September-2015*. DOI: 10.1145/2832987.2833007.
26. Maran, V., Augustin, I., Machado, G. M., Lima, J. C. D., Machado, A., De Oliveira, J. P. M. (2017), "Database ontology-supported query for ubiquitous environments", *Paper presented at the WebMedia 2017 - Proceedings of the 23rd Brazillian Symposium on Multimedia and the Web*, P. 185–188. DOI: 10.1145/3126858.3131575.
27. Maran, V., Machado, A., Machado, G. M., Augustin, I., de Oliveira, J. P. M. (2018), "Domain content querying using ontology-based context-awareness in information systems", *Data and Knowledge Engineering*, No. 115, P. 152–173. DOI: 10.1016/j.datak.2018.03.003.
28. Rob, P., Coronel, K. (2004), *Database Systems: Design, Implementation, and Management* : Trans. from English, SPb., BHV-Petersburg, 1023 p.
29. Langefors, B. (1980), "Infological models and information user views", *Inform. Systems*, Vol. 5, P. 17–32.
30. Pérez-Castillo, R., De Guzmán, I. G. R., Caivano, D., Piattini, M. (2012), "Database schema elicitation to modernize relational databases", *Paper presented at the ICEIS 2012 - Proceedings of the 14th International Conference on Enterprise Information Systems*, P. 126–132.

Received 03.11.2020

Відомості про авторів / Сведения об авторах / About the Authors

Аврунін Олег Григорович – доктор технічних наук, професор, Харківський національний університет радіоелектроніки, завідувач кафедри біомедичної інженерії, Харків, Україна; email: oleh.avrunin@nure.ua; ORCID: https://orcid.org/0000-0002-6312-687X.

Аврунин Олег Григорьевич – доктор технических наук, профессор, Харьковский национальный университет радиоэлектроники, заведующий кафедрой биомедицинской инженерии, Харьков, Украина.

Avrunin Oleg – Doctor of Sciences (Engineering), Professor, Kharkiv National University of Radio Electronics, Head of the Department of Biomedical Engineering, Kharkiv, Ukraine.

Власов Олексій В'ячеславович – Харківський національний університет радіоелектроніки, аспірант кафедри штучного інтелекту, Харків, Україна; email: oleksii.vlasov@nure.ua; ORCID: https://orcid.org/0000-0003-1619-0032.

Власов Алексей Вячеславович – Харьковский национальный университет радиоэлектроники, аспирант кафедры искусственного интеллекта, Харьков, Украина.

Vlasov Oleksiy – Kharkiv National University of Radio Electronics, Postgraduate Student of the Department of Artificial Intelligence, Kharkiv, Ukraine.

Філатов Валентин Олександрович – доктор технічних наук, професор, Харківський національний університет радіоелектроніки, завідувач кафедри штучного інтелекту, Харків, Україна; email: valentin.filatov@nure.ua; ORCID: https://orcid.org/0000-0002-3718-2077.

Филатов Валентин Александрович – доктор технических наук, профессор, Харьковский национальный университет радиоэлектроники, заведующий кафедрой искусственного интеллекта, Харьков, Украина.

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2020. No. 4 (14)

Filatov Valentin – Doctor of Sciences (Engineering), Professor, Kharkiv National University of Radio Electronics, Head of the Department of Artificial Intelligence, Kharkiv, Ukraine.

МОДЕЛЬ СЕМАНТИЧЕСКОЙ ИНТЕГРАЦИИ СВОЙСТВ ИНФОРМАЦИОННЫХ СИСТЕМ В ЗАДАЧАХ РЕИНЖЕНИРИНГА РЕЛЯЦИОННЫХ БАЗ ДАННЫХ

Предметом исследования являются методы семантической интеграции предметных областей гетерогенных информационных систем и распределенных баз данных. Такого класса системы, создаваемые на основе технологий баз данных, нашли широкое распространение и применение во всех сферах хозяйственной деятельности. Такие системы характеризуются большой трудоемкостью проектирования, сопровождения и модификации. **Целью** проводимых исследований является разработка модели предметной области на основании семантических свойств и связей элементов данных; разработка эффективной технологии интеграции информационных ресурсов гетерогенных вычислительных систем на основе технологии управления системами баз данных; исследование и формализация классов неоднородности структур данных, направленных на решение задачи определения типов информационных объектов. Разработка инструментальных средств проектирования и сопровождения прикладных задач интеграции гетерогенных информационных систем и распределенных баз данных. **Результаты**: проведен анализ существующих методов и моделей интеграции предметных области на основании семантических свойств и связей элементов данных; разработана эффективная математическая модель, технология и алгоритм семантической интеграции информационных ресурсов гетерогенных вычислительных систем для реляционных баз данных; исследованы и формализованы классы неоднородности структур данных, направленных на решение задачи определения типов информационных объектов; разработана модель и исследованы средств логического описания свойств информационных объектов для определения границы рассматриваемой предметной области. **Вывод**: в статье рассмотрена формализованная инфологическая модель предметной области, которая ориентирована на семантические отношения между информационными объектами баз данных. Сформулирован аксиоматический подход к описанию предметной области, который позволяет рассмотреть проблему моделирования отношений элементов предметной области в виде набора правил, определяющих существование элементов данных. На основании анализа структур и моделей баз данных информационных систем определен общий подход к построению универсальной технологии, ориентированной на решение задач управления гетерогенными информационными ресурсами вычислительных систем.

Ключевые слова: модель данных; семантика данных; база данных; интеграция данных; гетерогенная информационная система.

МОДЕЛЬ СЕМАНТИЧНОЇ ІНТЕГРАЦІЇ ВЛАСТИВОСТЕЙ ІНФОРМАЦІЙНИХ СИСТЕМ В ЗАДАЧАХ РЕІНЖИНІРИНГУ РЕЛЯЦІЙНИХ БАЗ ДАНИХ

Предметом дослідження є методи семантичної інтеграції предметних областей гетерогенних інформаційних систем і розподілених баз даних. Такого класу системи створюються на основі технологій баз даних, знайшли широке поширення і застосування у всіх сферах господарської діяльності. Такі системи характеризуються великою трудомісткістю проектування, супроводу і модифікації. **Метою** проведених досліджень є розробка моделі предметної області на підставі семантичних властивостей і зв'язків елементів даних; розробка ефективної технології інтеграції інформаційних ресурсів гетерогенних обчислювальних систем на основі технології управління системами баз даних; дослідження і формалізація класів неоднорідності структур даних, спрямованих на вирішення завдання визначення типів інформаційних об'єктів; Розробка та дослідження засобів логічного опису властивостей інформаційних об'єктів для визначення кордону розглянутої предметної області. Розробка інструментальних засобів проектування і супроводу прикладних задач інтеграції гетерогенних інформаційних систем і розподілених баз даних. **Результати**: проведено аналіз існуючих методів і моделей інтеграції предметних області на підставі семантичних властивостей і зв'язків елементів даних; розроблена ефективна математична модель технологія і алгоритм семантичної інтеграції інформаційних ресурсів гетерогенних обчислювальних систем для реляційних баз даних; досліджені і формалізовані класи неоднорідності структур даних, спрямованих на вирішення завдання визначення типів інформаційних об'єктів; розроблена модель і досліджено засобів логічного опису властивостей інформаційних об'єктів для визначення кордону розглянутої предметної області. **Висновок**: в статті розглянута формалізована інфологічна модель предметної області, яка орієнтована на семантичні відносини між інформаційними об'єктами баз даних. Сформульовано аксіоматичний підхід до опису предметної області, який дозволяє розглянути проблему моделювання відносин елементів предметної області у вигляді набору правил, що визначають існування елементів даних. На підставі аналізу структур і моделей баз даних інформаційних систем визначено загальний підхід до побудови універсальної технології, орієнтованої на вирішення завдань управління гетерогенними інформаційними ресурсами обчислювальних систем.

Ключові слова: модель даних; семантика даних; база даних; інтеграція даних; гетерогенна інформаційна система.

Бібліографічні описи / Bibliographic descriptions

Аврунін О. Г., Власов О. В., Філатов В. О. Модель семантичної інтеграції властивостей інформаційних систем в задачах реінженірінгу реляційних баз даних. *Сучасний стан наукових досліджень та технологій в промисловості*. 2020. № 4 (14). С. 5–12. DOI: https://doi.org/10.30837/ITSSI.2020.14.005

Avrunin, O., Vlasov, O., Filatov, V. (2020), "Model of semantic integration of information systems properties in relay database reengineering problems", *Innovative Technologies and Scientific Solutions for Industries*, No. 4 (14), P. 5–12. DOI: https://doi.org/10.30837/ITSSI.2020.14.005