Глобалізація інформаційних систем (ІС) в процесі постійної експлуатації і модернізації породжує нові проблеми, виявляє особливості представлення завдань розвитку ІС. Такою складною галузевою ІС, що вимагає відповідної онтологічної підтримки, є автоматизована система управління вантажними перевезеннями (АСК ВП УЗ-Є). Розвиток інформаційних підсистем інфраструктури, рухомого і тягового складу, управління рухом, фінансів, кадрів і ряду інших вимагають взаємопов'язаного розвитку онтологічного забезпечення.

З урахуванням цього були розроблені методи і засоби онтологічного забезпечення конструктивно-продукційного моделювання (ОКПМ), призначені для підтримки процесів багатоетапного створення, тривалого періоду функціонування і постійного розвитку АСК ВП УЗ-Є. Отримані результати характеризуються універсалізмом, забезпечують можливості подання еволюції об'єкта (ІС) і змісту онтологій в ОКПМ.

Побудова моделей і методів ОКПМ отримано шляхом розширення відносин і відображень, а також за рахунок створення нових структур породження, доповнення класів сигнатур новими відносинами із конструювання. При цьому отримала розвиток модель єдиної, універсальної та такої що налаштовується на різні предметні області онтологічної конструктивної структури (ОКС). ОКС враховує вимоги щодо поданням процесів розширення предметної області, а також уніфікації знань. Для завдань КПМ створені методи і засоби моделювання процесів концептуалізації об'єктів що розвиваються, побудовано конструктивну обчислення класу відображень погоджування, а також методи конструювання онтологічних об'єктів вищого порядку. Для онтологічної підтримки процесів КПМ в умовах розширення предметних областей вдосконалені методи виведення в моделі конструктивної структурі ОКПМ, запропоновані процедури змістовної, структурної та пов'язаної виводимості, а також багаторівневі методи виведення.

Наведені приклади реалізації діючих в даний час у автоматизованій системі АСК ВП УЗ-Е процедур створення нових додатків, з використанням запропонованих засобів КПМ. Приклади демонструють достатність розроблених моделей і засобів ОКПМ щодо реалізації діючих процедур розвитку і супроводу складних залізничних АСУ

Ключові слова: онтологічне забезпечення, конструктивне моделювання, концептуалізація, методи виведення, автоматизовані системи

UDC 004.82/.942
DOI: 10.15587/1729-4061.2018.143968

ONTOLOGICAL SUPPORT FORMATION FOR CONSTRUCTIVE-SYNTHESIZING MODELING OF INFORMATION SYSTEMS DEVELOPMENT PROCESSES

V. Skalozub
Doctor of Technical Sciences, Professor, Dean*
E-mail: skalozub.vl.v@gmail.com
V. Ilman
PhD, Associate Professor*
E-mail: valery_ilman@ukr.net
V. Shynkarenko
Doctor of Technical Sciences, Professor, Head of Department*
E-mail: shinkarenko_vi@ua.fm
*Department of Computer Information Technologies
Dnipropetrovsk National University of Railway Transport named after academician V. Lazaryan
Lazaryana str., 2, Dnipro, Ukraine, 49010

1. Introduction

At present, applied subject ontologies (SOs) and ontological systems (OSs) become widely developed and used in various activities [1, 2]. Moreover, the range of tasks and software within OSs is constantly expanding. Ontological systems provide effective functioning of diverse information technology (IT), mainly in complex, poorly structured and sophisticated high-tech areas of knowledge and activity. The paradigm of developing a knowledge-oriented society [3] implies a substantial reworking of modern information resources, systems, and IT.

There are several main approaches to establishing subject ontologies [4]. The founding principles of designing ontological support for constructive-synthesizing modeling (CSM) processes in heterogeneous, structurally complex IT are expressed in [5–8]. They suggest a structural approach to creating tools for ontological support for CSM. Moreover, the constructive-synthesizing modeling ontology itself was also built using the CSM methodology. In ontologies, CSM extends the possibilities of modeling and representing entities in the constructed process of creating and developing SO knowledge models. The constructivism paradigm in the present study includes the features of modeling, presenting and implementing mechanisms of IT development factors. Besides, the specific properties of complex information systems are not pre-planned processes of their expansion and reconstruction.

As examples of complex sectoral information technologies that require ontological support, we consider automated control systems (ACSs) for railway transport (RT) in Ukraine. The industry is supported by a network of ACSs. The characteristic properties of many RTACSs are the multistage design and implementation, durable operation, continuous development and significant maintenance modifications. An example of an ACS is the unified automated system for managing freight traffic of Ukrzaliznytsia (State Administration of Railway Transport of Ukraine) – ACS FT UZ-U) [9]. The implementation of these IS properties is a significant problem in the ontological support for CSM.

Subject-independent methodologies for developing ontologies are based on several design and implementation principles proposed by Grubber [10]; they include clarity, consistency, extensibility, and minimum ontological obligations. Applied ontologies (AOs) including CSM for RTACSs should also be extensible and modifiable. When designing such AOs, it is essential that new facilities should contain several diverse quality properties. On the one hand, new facilities should be connected and coordinated with the existing ones. On the other hand, the introduction of previously unknown new elements (terminology, concepts, and nodes) to ontologies of RTACSs in practice often somewhat transforms the properties and content of already existing elements of knowledge about the ACS. For example, due to new connections and relations arising between the «nodes» of the subject ontology, additional properties of objects appear. In addition, on the basis of this knowledge, other interpretations of concepts or query performance for OSs can be obtained. The processes of developing an RTACS and implementing the principle of extensibility of the OS require the creation of special models and procedures within the framework of the OCSM methodology.

To clarify some of the problems of expanding and developing the IT for the RTACS, let us consider an example of a part of the ACS FT UZ-U structure. Its components are the following [9]:

1. 1. Operational management of the transportation process (TPOM).

1. 2. Freight transport management.

1. 3. Passenger traffic management.

1. 5. Management of the border area.

1. 6. The system of statistics of volume indicators of income from freight transportation.

1. 7. Analysis of productive and financial activities, etc. Detailing the TPOM site produces such tasks.

1. 1. 1. Control of the location and condition of trains.

1. 1. 2. Control of the location and condition of freight wagons.

1. 1. 3. Station management.

1. 1. 4. The border system.

1. 1. 5. Control of dislocation and condition of cargo shipments.

1. 1. 6. The schedule estimation of the cargo arrival, etc.

Initially, the basic components of the ACS FT UZ-U apparently formed a hierarchical structure. As the ACS was developed, these connections turned out to be broken. Thus, the TPOM (1.1) functions began to include the functions of the border system (1.1.4) related to the management of the border area (1.5). The functions of the statistical system (1.6) are related to the analysis of productive and financial activities (1.7). There are also obvious connections between the blocks of functions (1.1.5) and (1.1.6). Therefore, the correction of information in any of the blocks will require the adjustment of models and nodes of some objects. That is, in practice, the system of concepts that characterize the individual components of the ACS FT UZ-U forms a network structure. Moreover, when introducing new objects into the applied ontology and reconstructing properties or connections between objects, it is necessary to apply special system-wide methods and procedures for maintaining the IT of the ACS.

The problem of presenting a transforming and evolving system of knowledge is relevant for the creation and maintenance of existing information technologies for the RTACS. The tasks of developing and improving the automated control system of the RT, including the control system of the FT UZ-U, entail the creation of a complex of analytical servers (ASs). Decision-making processes (DMPs) have a number of common tasks and functions that, within the framework of the ACS FT UZ-U, should be based on a common information ground and common implementation methods. The AS system, intended for the unified deployment of intelligent services of the ACS FT UZ-U, is also based on the CSM methodology and the applied CSM ontology.

2. Literature review and problem statement

One of the key directions in developing information technology of the latest and current decades is the increase in efficiency by means of intellectualization. To overcome this global problem, many tools and methods have been developed and continue to be devised.

In many cases, the tasks of increasing the efficiency of complex industrial, technological, scientific and other types of IT is solved through the use of ontological support tools [1, 2]. In [1], a concept is considered for building multipurpose ontological systems that can be effectively used to create relatively simple ontological applications to support the systematic integration of subject knowledge. Relations between various ontologies are proposed to be performed in the form of operations ontographs [2]. The essential role of connected ontologies is shown in [11]. There, the key point is the relationship between the ontology of services and universal ontology. Along with two-level ontologies, multilevel ones are also considered in diverse variants of level selection [1, 12].

The formation of new levels can be considered as a simulation of ontology development.

In [1, 2, 11, 12], conceptual directions for the creation and development of applied ontologies and families of ontologies are proposed for various subsequent implementations. To automatize these processes, further formalization is needed: the development and study of appropriate formal models.

In a number of works considered further, the solution of various tasks of ontological support of information systems, regardless of their perspective, directly or indirectly entails the need to develop ontologies, devise ontological support for developing information systems, and formalize these processes.

Complex tasks of OS development are implemented to function both independently and as part of ontologically managed IT, providing information and analytical support for operating procedures [3]. It is important «to apply the methods and means of ontological analysis, including effective procedures for ontological search for primary information and cognitive data processing, up to the automation of developing an ontological description of the subject area». Methods and means of automation suggest the presence of a formal apparatus of inference proposed in this work.

The practice of creating and developing information systems has shown the importance of the results of activities at the early stages. Among them, conceptual data modeling was considered critical in [13]. In [14], it was shown that «the advantage of using ontology for modeling conceptual data is the reuse of domain knowledge. As a result, the concept data model will be produced faster, simpler, and with fewer errors than creating the conceptual data model in the usual way.» When solving specific problems of informatization, the need for conceptualization was identified. This can be viewed as a suggested direction for further research. In all the OS developments noted above, the task of providing IT development factors was clearly not presented. However, for each application, specific models and procedures for expanding knowledge were used.

In [15], ontological support is used to develop an intelligent system for automatic text classification, in which the principle of extension is implemented. The development of ontologies is performed on an empirical basis. In [13], it is proposed to use ontologies for automatic modeling of the subject area by the formation of conceptual data models. At the same time, the data ontology and model are transformed in terms of the calculus of predicates of the first order and are displayed in the form of PROLOG sentences. The formalisms are indicated, on the basis of which the models of developing subject ontologies can be implemented. The formalization of ontology (but not its development) has a rather narrow focus. In this regard, knowledge increment models act as some forms of OS development, on the basis of which means of ontological support can be created.

The importance of «developing the methodology and tools of computer-aided design of a formal (computer) ontology» is also noted in [16]. How automation tasks are solved on the basis of the preference principle is shown in [17]. Another approach to automation is based on the idea of including paradigmatic and syntagmatic relations for data processing [18]. In [16], it is noted that «so far...there remains an open question of the emergence of a constructive theory of the development of formal ontologies,» and in [17, 18], some key elements are suggested as possibly useful in a constructive approach.

It is also essential to note the development of powerful research areas related to the use of ontologies in IT. The importance of ontological support in developing IT is highlighted in [19]. The practical benefits of ontological support in solving diverse tasks in developing information systems are confirmed by a number of studies. Thus, in [20], a methodology for ontology-based classification system is proposed. Using this methodology, classification accuracy can be improved in some cases. In [21], ontological support ensures the survivability of the plans of the Ministry of Defense. In [22], it is noted that «the semantic approach, in which the main solution is annotated with ontological statements, thereby makes the requirements explicit and accessible for further machine processing». The development of theoretical foundations will make it possible to unify and improve these machining processes.

The benefits of ontological support during the development of information systems are shown in [23]. However, the problems of connected, simultaneously developing ISs are not taken into account. Principles for developing ontologies and tools for ontological compatibility [24] do not yet have a sufficient formal mathematical justification, since the processes of devising and developing ontologies themselves are constructive in nature. Therefore, the most adequate approach to their study and improvement is a rather specific constructive modeling.

The use of ontologies in developing IT is not possible without working together on ontologies [25].

Modern ontological support for IT extends both to the formation of models of concepts (individual concepts, links, means of creating entities, etc.) as well as the use of specialized output procedures. Such attributes have different applied ontologies for mathematically and logically complex IT applications (fuzzy control, multicriteria analysis, and data interpretation). The noted features of applied ontologies that represent developing IT provide the basis for the formation of an improved system of models of the CSM ontology.

When summing up the opinions on the ontological support of information systems on the general problems of SOs development while taking into account the requirements for displaying the processes of expansion and development of knowledge, it is necessary to note the lack of mathematical substantiation and tools for creating ontological support for constructive modeling procedures. The above-mentioned articles specify, in particular, the need for automation, conceptualization and extension of the ontological support for developing information systems. In most of these works, such needs were satisfied by highly specialized or empirical methods.

In this regard, the development of methods and means of ontological support for CSM, taking into account the development of IT, is an urgent task for the constructive formation of an OS.

3. The aim and objectives of the study

The aim of the present study is to develop methods and improve the means of ontological support for CSM processes that take into account the principle of the expansion and development of an OS.

To achieve this goal, the following objectives were set and solved:

– to develop methods and procedures for constructive modeling of conceptualization processes of developing objects and the formation of higher-level ontology domains;

– to develop methods and procedures for the content and structural deductibility of the highest order in the constructive structure of the OCSM;

– to create means of support for the development of applied ontologies for the existing RTACS.

4. Materials and methods of research on the ontological support for constructive-synthesizing modeling of developing information systems

4. 1. Fundamentals of the constructive-synthesizing modeling

According to the constructivism paradigm, all objects and properties of a SO must be explicitly constructed using the models of the constructive structure of the CSM [7]. In the framework of the OCSM, all instances of models of the ontology $\mathfrak{K}$ concepts have special and generic forms of description, constructing properties, indicators, and attributes among other components. Conceptualization of the ontology $\mathfrak{K}$ is given by forming notions, IT concepts, classes of objects $Ob\mathfrak{K}$, relationships and mappings $Rf\mathfrak{K}$. Conceptualization of SO elements is performed by a constructive model of the OCSM, defined by the quartet:

$$C = \langle \mathrm{M}, \Sigma, \Lambda, \mathrm{Z} \rangle, \tag{1}$$

where $\mathrm{Z} = \mathrm{Z}_V \cup \mathrm{Z}_v$ is the class of performers, consisting of subclasses of external Z_V and internal Z_v performers and a hybridization of the performers Z_q; classes of calculations $\Lambda = \Lambda_0 \cup \Lambda^*$, action and relation signatures $\Sigma = \Sigma_0 \cup \Sigma^+ \cup \Sigma^*$ as well as the carrier $\mathrm{M} = \mathrm{M}_0 \cup \mathrm{M}^*$, intended for the formation of the class of objects $Ob\mathfrak{K}$. Within the OCSM, the basic classes Λ_0, M_0, Σ_0, and Σ^+ are system-generating, providing the debelopment of generating classes Λ^*, M^*, and Σ^*. The base carrier M_0 is the initial one for the step-by-step formation of the generic constructive classes: objects $\mathfrak{O}^k$, features $\mathfrak{L}^k$, and derived ontological objects $\mathfrak{O}^k_* \subset \mathrm{M}^*$ of the k levels (stages) of development. The generating signature Σ_0 consists of the class of simple operators $\mathfrak{D}^0 \subset \mathfrak{D}$, the class of simple relations and mappings $\mathfrak{Y}^0 \subset \mathfrak{Y}$, including binding subclasses, and also the mapping class of generating selection $\mathfrak{V}^0$ of models of ontological objects. A simple action class is intended for the model C to form complex relationships and mappings of the class Σ^*.

Technologically, the design of entities in the ontological model of the OCSM occurs according to the scheme:

– an external performer supplies the constructive structure model with structure elements (samples) of the subject domain (SD), action (reflection) and relations between the samples, as well as systems of rules, instructions, and axioms (calculus) over entered objects;

– according to the rules of calculating the comparison relations between elements of the CSM, the design of the elementary characteristics of the samples is carried out;

– relations of subordination of features and samples generate the attributive entities of ontology;

– the entities are formed by mapping the connection across attribute constructs;

– complex objects of knowledge are constructed recursively by compositions of the indicated relations and mappings, and they are classified according to the levels of entity formation;

– the formed entities represent free classes of knowledge on which new entities can be constructed, or external executors organize an exit from the CSM. Development of the basic ontology into an OCSM is possible due to the expansion of relations and mappings as well as by creating new generating structures, extending the classes of signatures with new constructor relations, etc.

4. 2. Methods of designing families of level ontologies of higher-order domains for applied CSM ontologies

In model (1), the formation of models of zero-order SO objects is performed on the base classes M_0, Σ_0, and Λ_0 as well as performers Z. The build-up of the constructive capabilities of the OCS is accomplished by expanding structure (1) by the following methods.

4. 2. 1. Extension of the basic classes and first-order constructs

To extend the classes of ontology, we use the fact that in model (1) the signature Σ^+ is given on the classes of simple $\mathfrak{Q}$ and complex $\mathfrak{M}$ mappings.

The definition. A pair of classes of concepts or mappings (A,B), $A, B \subset Q$, $Q \subset \mathfrak{O}^k | \Sigma^*$, and $\mathfrak{O}^k \subset \mathrm{M}$, where $B \subset A$, defines in the OCS structure the class B extension by class A in the mapping class $\mathfrak{Q} \subset \Sigma^+$, if the calculations of $(\forall a \in B, a \in \mathfrak{O}^k)$ or $(\forall \mu \in B, \mu \in \Sigma^*)$ are connected by the dependence: inclusion $a\Lambda \subset A\Lambda$ $(\mu\Lambda \subset A\Lambda)$ involves inclusion $a\Lambda \subset B\Lambda$ $(\mu\Lambda \subset B\Lambda)$, where $a\Lambda$ and $A\Lambda$ are the results of the calculations on the class A and its element $a \in A$. In this case, the extended map $\bar{\mu}$ has the property of an inducing morphism.

The structure signature (1) contains the mapping class $\mathfrak{M} \subset \Sigma^+$, which helps create new constructs of the generating class $\Sigma^* \subset \Sigma$, for which $\mathfrak{Y}, \mathfrak{V}, \mathfrak{D} \subset \Sigma^*$ on classes $\mathfrak{Y}^{k-1}_n \subset \mathfrak{Y}$, $\mathfrak{V}^{k-1}_n \subset \mathfrak{V}$ and operators $\mathfrak{D}^{k-1}_n \subset \mathfrak{D}$. Let us consider the formation and application of more complex constructive mappings.

4. 2. 2. Constructive calculus of the generating class of maps

Suppose $\omega \in \mathfrak{M}$ is any generating map of the n-th order (or rank), and the family of concepts of arbitrary orders is as follows:

$$\mathfrak{H} = \mathfrak{F}_{\mathfrak{O}^k} \,\&\, \mathfrak{F}_{\mathfrak{D}^r} \,|\, \mathfrak{F}_{\mathfrak{R}^i} \,|\, \mathfrak{F}_{\mathfrak{A}^j} \,|\, \mathfrak{F}_{\mathfrak{Y}^q}$$

and

$$\mathfrak{G} = \mathfrak{F}_{\mathfrak{D}^{r+r_1}} \,|\, \mathfrak{F}_{\mathfrak{R}^{i+i_1}} \,|\, \mathfrak{F}_{\mathfrak{A}^{j+j_1}} \,|\, \mathfrak{F}_{\mathfrak{Y}^{q+q_1}},$$

then the following properties of the formation of constructive-synthesizing models take place:

– the mapping ω acts on the families $\mathfrak{H}$ and $\mathfrak{G}$ so that:

$$\omega : \prod_n (\mathfrak{H}, \mathfrak{G}) \to \mathfrak{G}, \tag{2}$$

– for specific generators of the maps ω and the families $\mathfrak{H}$ and $\mathfrak{G}$, it is possible to have a hybrid representation of the ontological constructions:

$$\prod_{n}{}_{\mathfrak{H}} f_j = f \text{ and } f_{j_1} f_{j_2} \cdots f_{j_k} = f,$$

where

$$f_j, f_{j_i} \in \mathfrak{H} | \mathfrak{G},\ i = 1..k \text{ and } f \in \mathfrak{G}; \tag{3}$$

– the construction $f_0 \in \mathfrak{G}$ is empty if it is composed of empty components;

– the mapping $\omega_0 \in \mathfrak{M}$ is neutral if it forms the entity:

$$\left(\prod_{n}{}_{\mathfrak{H}} f_0\right)\omega_0 = \prod_n \mathfrak{H} \text{ or } \left({}_{f_0} f_j\right)\omega_0 = f_j \text{ or}$$

$$\left(f_0 f_j\right)\omega_0 = f_j, \text{ or } \left(f_j f_0\right)\omega_0 = f_j; \tag{4}$$

– the mapping $\omega \in \mathfrak{M}$ is neither commutative nor associative with respect to the composition operator $\phi = \bullet$;

– the class $\mathfrak{M}$ is a class of weak morphisms (there are partial properties of morphisms) with respect to the composition operator;

– $\forall \phi \in \mathfrak{D}^{k-1}_n$ can be formed on the entities:

$$\prod_{n}{}_{\mathfrak{H}} f_r = f \text{ and } (f_i f_j \cdots f_k)\phi = f$$

as well as hybrid representations of entities so that:

$$(f, \prod_{n\mathfrak{H}} f_j)\phi,\ (\prod_{n\mathfrak{H}} f_j, f)\phi,\ \left(\prod_{n\mathfrak{H}} f_j, \prod_{n\mathfrak{H}} f\right)\phi, \text{ etc.,} \qquad (5)$$

– the mapping structures:

$$\prod_{n\mathfrak{H}} f_j = f,\ (f_i f_j \cdots f_k)\phi = f$$

and the corresponding hybridization allow for iterative formations of ontological entities in the classes $\mathfrak{H}$ and $\mathfrak{G}$.

Examples of hybrid constructive ontological operators and mappings are the following constructions:

$$((_{v_j} v_i)_s, (_{\rho_j} v_i), \prod_{n\mathfrak{H}} f)\phi,$$

$$\begin{pmatrix} (_{\rho_j}\rho_i)_s, (f_{j_1} f_{j_2} \cdots f_{j_k})_k, \\ (_{v_j}\rho_i)_s, (\prod_{n\mathfrak{H}} f_r \prod_{n\mathfrak{H}} f)_k \end{pmatrix}\phi \text{ and others.} \qquad (6)$$

A fragment of the process of level and ordinal construction of the classes of models of ontological objects, mappings and domain ontologies $\mathfrak{K}^k$, $k = 0,1,2,\cdots$ is presented in Fig. 1.

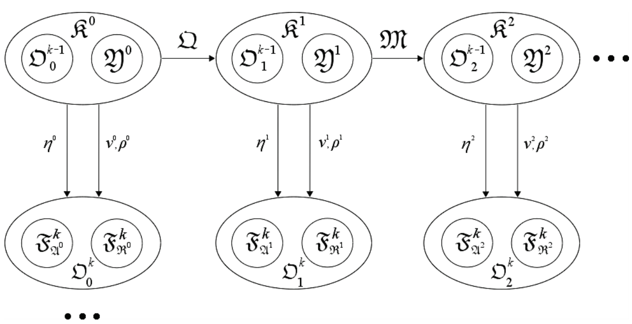

Fig. 1. The outline of the process of level and sequential formation of models of entities of ontologies

In Fig. 1, the vertical direction shows the level, while the horizontal direction shows the order formation of the structural models of ontologies.

5. Development of methods for modeling objects of higher order and output in the constructive structure of the OCSM as a result of the study

Families of ontological objects of higher order help introduce an ordered system of domain models of an IT area. Models and methods of output play an important role in the OCSM. On their basis, numerous tasks are performed to ensure the functioning of the CSM ontologies: the implementation of solution search, verification of identities or equivalence of models of ontological objects, the formation of ways to construct objects with the required properties, the construction of models of complex concepts, etc. Due to the possibility of expanding and developing IT domains, the structure of the OCSM (1) must have certain means that ensure the transformation of the output methods. Let us consider the issues of the development of withdrawal procedures in the structure of the OCSM (1).

In the OCSM, the models of the generated objects have both content and structural components, which should be taken into account in the output procedures. For this purpose, the content σ_c^0 and structural σ_s^0 modifications of the mappings are determined in the class of mappings of the derivative of the signature generator $\mathfrak{V}^0$. The content deducibility takes into account the generic and special features of the structure of the design model, whereas the structural deducibility accounts for properties of binding relations.

5. 1. Methods of modeling of ontological objects of the *m*-th order

Families of ontological objects of higher order help introduce an ordered system of domain models of an IT area. The construction of families of subclass objects $\mathfrak{O}_m^{k+1} \subset \mathfrak{O}_*^{k+1}$, $\mathfrak{O}_*^{k+1} \subset \mathrm{M}$, is performed using the iterative procedure:

$$\mathfrak{F}_{\mathfrak{O}_{m+1}^k} \leftarrow (\mathfrak{F}_{\mathfrak{O}_m^k}, \mathfrak{F}_{\mathfrak{R}^m}, \mathfrak{F}_{\mathfrak{D}^m}, \mathfrak{F}_{\mathfrak{A}^m}),\quad m = 0,1,2,\cdots, \qquad (7)$$

where $\mathfrak{R}^m, \mathfrak{D}^m, \mathfrak{A}^m \subset \Sigma^*$, $(\mathfrak{R}^m, \mathfrak{A}^m \subset Rf\mathfrak{K}^m)$, and $\mathfrak{O}_m^k \subset \mathfrak{O}_*^k$ are already formed families of class constructions. For values of $m > 1$, they are formed by calculating the generating signature Σ^+ and the class mappings Σ^*.

Let the class of families $\mathfrak{H}_{m-1}^k$ of the *k*-th level and $m-1$-st order be built on the four classes $\mathfrak{O}_{m-1}^k$, $\mathfrak{R}^{m-1}, \mathfrak{D}^{m-1}, \mathfrak{A}^{m-1}$. The scheme of the algorithm for constructing the ontology $\mathfrak{K}_m^{k+1}$ of the classes $\mathfrak{R}^m, \mathfrak{D}^m, \mathfrak{A}^m, \mathfrak{O}_m^{k+1}, \mathfrak{L}_m^{k+1}$ is as follows:

(1) the mapping:

$$\omega_{\mathfrak{R}} : \prod_n (\mathfrak{H}_{m-1}^k, \mathfrak{R}^{m-1}) \to \mathfrak{R}^m,$$

$\omega_{\mathfrak{R}} \in \mathfrak{M}$ forms the class of relations $\mathfrak{R}^m$, where

$$\mathfrak{H} = \mathfrak{F}_{\mathfrak{O}} \,\&\, \mathfrak{F}_{\mathfrak{D}} \mid \mathfrak{F}_{\mathfrak{R}} \mid \mathfrak{F}_{\mathfrak{A}};$$

(2) the mapping:

$$\omega_{\mathfrak{D}} : \prod_n (\mathfrak{H}_{m-1}^k, \mathfrak{D}^{m-1}) \to \mathfrak{D}^m,$$

$\omega_{\mathfrak{D}} \in \mathfrak{M}$ constructs a family of operators and algorithms of the class $\mathfrak{D}^m$;

(3) a complex class of mappings of the compound $\mathfrak{A}^m$ is constructed by the mappings:

$$\omega_{\mathfrak{A}} \in \mathfrak{M} \text{ and } \omega_{\mathfrak{A}} : \prod_n (\mathfrak{H}_{m-1}^k, \mathfrak{A}^{m-1}) \to \mathfrak{A}^m;$$

(4) the operators $\phi \in \mathfrak{D}^m$ on the mappings of the class $\mathfrak{R}^m$ and operator algorithms of the class $\mathfrak{D}^m$ on the objects of the class $\mathfrak{O}_{m-1}^k$ construct the family of structures $\mathfrak{F}_{\mathfrak{R}}^{k+1}$;

(5) the hybrid of mappings and operators of the classes $\mathfrak{A}^m$ and $\mathfrak{D}^m$ on the class of objects $\mathfrak{O}_{m-1}^k$ forms the family of the concepts $\mathfrak{F}_{\mathfrak{A}}^{k+1}$;

(6) the class mappings $\mathfrak{R}^m$ and $\mathfrak{A}^m$ as well as a hybridization on the class $\mathfrak{O}_{m-1}^k$ form the hybrid family of concepts $\mathfrak{F}_g^{k+1}$;

(7) the constructed families are included in the ontological class used to generate the concepts $\mathfrak{F}_{\mathfrak{R}}^{k+1}, \mathfrak{F}_{\mathfrak{A}}^{k+1}, \mathfrak{F}_g^{k+1} \subset \mathfrak{O}_m^{k+1}$;

(8) the subfamilies of the family $\mathfrak{F}_g^k$ and relationships of the class $\mathfrak{R}^m$ are used to construct the class of characters $\mathfrak{L}_m^{k+1}$.

The existence of a certain neutral mapping $\omega_0 \in \mathfrak{M}$ and an empty construction $f_0 \in \mathfrak{G}$, produce the theorem.

The theorem.

The consequence. The generation classes of ontological objects make the following taxonomy valid:

$$\cdots \subset \mathfrak{O}_{m-1}^{k+2} \subset \mathfrak{O}_{m}^{k+2} \subset \mathfrak{O}_{m+1}^{k+2} \subset \cdots \subset \mathfrak{O}_{m-2}^{k+1} \subset \mathfrak{O}_{m-1}^{k+1} \subset \mathfrak{O}_{m}^{k+1} \subset \cdots$$
$$\cdots \mathfrak{O}_{m-3}^{k} \subset \mathfrak{O}_{m-2}^{k} \subset \mathfrak{O}_{m-1}^{k} \subset \mathfrak{O}_{m}^{k} \subset \mathfrak{O}_{m+1}^{k} \subset \cdots$$
$$\cdots \subset \mathfrak{O}_{m-1}^{k-1} \subset \mathfrak{O}_{m}^{k-1} \subset \mathfrak{O}_{m+1}^{k-1} \subset \cdots \subset \mathfrak{O}_{m}^{k-2} \subset \mathfrak{O}_{m+1}^{k-2} \subset \mathfrak{O}_{m+2}^{k-2} \subset \cdots.$$

The above algorithmic scheme helps construct extensions and models of the elements of subject ontologies – in particular, complex procedures for generating the mappings $\mathfrak{Y}^m$ and operator constructions of the class $\mathfrak{D}^m$. In general, the methods of forming mappings and operators of the classes $\mathfrak{Y}^m$ and $\mathfrak{D}^m$, as well as the class of concepts generated by them, ensure the creation of models of objects of the k-th level and m-th order of the subject ontology.

5. 2. Calculus of substantial and structural derivability

Calculus enables the formation of ontological objects by hybrid methods of derivation, as well as the construction of complex ontology objects based on various derivability procedures. The procedure for the derivability of the contents of the concepts σ_c^0 of the generating class $\mathfrak{O}_0^0$ is given by the rule:

$$\sigma_c^0:\ b\sigma_c^0 = (c_k\varphi_r(x_k\varphi_{r1}), c_i\varphi_h(x_i\varphi_{h1}))_j, \tag{8}$$

where

$$b \neq b_0,\ \ a_m\varphi_q = a_n,\ \ a = c\,|\,x,\ \ c \in \mathfrak{C}_0,\ \ x \in \mathfrak{X}_0$$

and

$$q = r, r1\,|\,h, h1,\ \ m \neq n\,|\,m = n;$$

are inducing mappings, and

$$b = (c_k(x_k), c_i(x_i))_j \in \mathfrak{O}_0^0$$

are generic and specific entities constructed by some connecting relation or mapping $\nu_j, \rho_j, \eta_j \in \mathfrak{Y}^0$.

Structural derivability of the SO concepts σ_s^0 for the class of objects $\mathfrak{O}_0^0$ is determined by the relationship according to the rule:

$$b_{mhj}\sigma_{s_i}^0 = b_{mhi},$$

where

$$i: v_i\,|\,\rho_i\,|\,\eta_i \in \mathfrak{Y}^0. \tag{9}$$

There is an associated derivability of the meaningful and structural mappings σ_c^0, σ_s^0, which sets the direction of the associated derivability $\cdots b\sigma_h^0\sigma_h^0\sigma_h^0 \cdots = \psi^0$, where $h = c\,|\,s$. The associated derivability ψ^0 is controlled by internal performers.

The statement. For meaningful and structural relations of deducibility, the methods of level and ordinal expansion are valid.

The proof is based on the rules:

$$(b_1, b_2, \ldots, b_n)\delta_i\sigma_b^1 = \left(b^j\varphi_1, b^j\varphi_2, \ldots, b^j\varphi_m\right)\delta_i,$$

where $b^j = (b_{j_1}, b_{j_2}, \ldots, b_{j_r})$,

$$r \in 1..\,n,\ \ \delta_i \equiv \rho_i\,|\,\eta_i\,|\,\circ_i,\ \ b\varphi_j = b^k\rho_r\,|\,b^k\eta_q\,|\,b^k\sigma_h^i,\ \ h = c\,|\,s;$$

and

$$(b_1, b_2, \ldots, b_n)\delta_i\sigma_\delta^1 = \left(b\delta_i\varphi_1, b\delta_i\varphi_2, \ldots, b\delta_i\varphi_m\right),$$

$$b\delta_i\varphi_j = b\delta_{ij},\ \ \delta_{ij} \equiv \rho_{ij}\,|\,\eta_{ij}.$$

The associated deducibility ψ^0 is controlled by internal executors if $\forall\sigma_h^0 \in \mathfrak{V}^0$, $\exists(\rho_k \in \mathfrak{R}_V\ \&\ p_r \in \mathfrak{L}^0\ \&\ z_j \in Z^v)$ and the rule is fulfilled that:

$$(p_r, \overset{z_j}{\sigma}{}_h^0)\rho_k = {}^{\rho_k}_{p_r}\overset{z_j}{\sigma}{}_h^0, \tag{10}$$

whereas

$$\cdots b\ {}^{\rho_3}_{p_{r,3}}\overset{z_3}{\sigma}{}_h^0\ {}^{\rho_2}_{p_{r,2}}\overset{z_2}{\sigma}{}_h^0\ {}^{\rho_1}_{p_{r,1}}\overset{z_j}{\sigma}{}_h^0 \cdots = \psi_u^0.$$

In the derivability mapping of ${}^{\rho}_{p_r}\sigma_h^0$ (10), the feature p_r, is used on the basis of the linking subordination relation $\rho \in \mathfrak{R}_V$, which shows that a display mapping is applied to the mapping σ_h^0.

5. 3. Advanced and multi-level methods for derivability in the CSM

The formation of a class of ontological objects of the first order $\mathfrak{O}_1^k$ in the CSM is performed by maps and operators of extended classes $\bar{\mathfrak{Y}}^0, \bar{\mathfrak{D}}^0$, as well as compositions on the class of objects $\mathfrak{O}_0^0$. Therefore, derivability on the class $\mathfrak{O}_{1*}^k$ is performed with the help of extended mappings of the derivability $\sigma^1 \in \bar{\mathfrak{V}}^0$, of features of the classes $\bar{\mathfrak{L}}^0$ or $\mathfrak{L}^1$ by performers of the class Z_v.

The method of extending the derivability mapping for the ontological objects $b_j \in \mathfrak{O}_1^k$ is determined by the rules:

$$(b_1, b_2, \ldots, b_n)\delta_i\sigma_b^1 = \left(b^j\varphi_1, b^j\varphi_2, \ldots, b^j\varphi_m\right)\delta_i, \tag{12}$$

where

$$b^j = (b_{j_1}, b_{j_2}, \ldots, b_{j_r}),\ \ r \in 1..\,n,\ \ \delta_i \equiv \rho_i\,|\,\eta_i\,|\,\circ_i,$$

$$b\varphi_j = b^k\rho_r\,|\,b^k\eta_q\,|\,b^k\sigma_h^i,\ \ h = c\,|\,s;$$

$$(b_1, b_2, \ldots, b_n)\delta_i\sigma_\delta^1 = \left(b\delta_i\varphi_1, b\delta_i\varphi_2, \ldots, b\delta_i\varphi_m\right),$$

$$b\delta_i\varphi_j = b\delta_{ij}, \tag{13}$$

where $\delta_{ij} \equiv \rho_{ij}\,|\,\eta_{ij}$, $b = (b_1, b_2, \ldots, b_n)$; both simple and complex derivability is possible: $(\sigma_{b1}^1, \sigma_{b2}^1)\phi$, $(\sigma_b^1, \sigma_\delta^1)\phi$, $(\sigma_\delta^1, \sigma_b^1)\phi$, $(\sigma_\delta^1, \sigma_b^1, \sigma_h^0)\phi$, $(\sigma_\delta^1, \sigma_b^1, \sigma_\delta^1)\phi$ and so on, where $\phi \in \mathfrak{D}^0\,|\,\mathfrak{D}^1$; the units of the mapping extensions are the units of the corresponding classes $\bar{\mathfrak{R}}^0, \bar{\mathfrak{D}}^0, \bar{\mathfrak{A}}^0$.

5. 4. Implementation of procedures for the expansion of automated systems

Let us give an example of the task of creating a new application for the ACS FT UZ-U using CSM tools, which provides the numbering and control of the use of wheel pairs (NCWP). The NUWP application performs the functions of generating, monitoring, and displaying such data: filed data for each wheel pair (WP) at all stages of operation, WPs of wagons and depots. To implement the NCWP, it is necessary to create new systems for reporting data as well as informational messages in the ACS FT UZ-U. For example, let us attribute this information to a new task of «Managing repairs and maintenance of freight wagons». The new application is created according to the rules in [9]:

1. Determine the place (category) of the NCWP application in the ACS FT UZ and its functional tasks.

2. Form a new object (objects) of the Logical Database.

3. Define operations with the object of the logical database that reflect the implementation of functional tasks.

4. Create a new object (objects) of the Physical Database, intended for building information models related to the NCWP.

For the NCWP, it is also necessary to develop (or modify) documents for some supporting systems. The conventional tasks for creating a new application for the ACS FT UZ are presented in Fig. 2.

The specified procedures to expand the automated control system of freight transportation by Ukrzaliznytsia are implemented in the OCSM as follows. Using the actions of the comparison relationship class $\mathfrak{R}_v^0 \subset \mathfrak{R}^0 \subset \Sigma_0$, models of the characteristics of wheel pairs are created (WP – wagon, WP – operation, WP – repair, and other relations). As a result, we obtain the acquired knowledge of the specific concept of the WP. If C_i is a specific instance of the WP class $\mathfrak{C}_0 \subset \mathrm{M}_0$, and its generic copies (internal properties: type, manufacturer, and WP parameters) $x_j \in \mathfrak{X}_0 \subset \mathrm{M}_0$, then the wheel concept (WC) is represented as $c_i(x_1, x_2, \cdots, x_k) = a_{ik}$. The WC has certain indicators and values of the corresponding classes $\mathfrak{I}_0$ and $\mathfrak{U}$, $\mathfrak{I}_0, \mathfrak{U} \subset \mathrm{M}_0$. The set of these values forms a set of WC attributes. For the formation of the external characteristics of a WP, the comparison actions $v_r \in \mathfrak{R}_v^0$ of the class $\mathfrak{R}_v^0$ are used along with the actions of the class of subordination relations $\mathfrak{R}_V^0 \subset \mathfrak{R}^0$ [7].

In the OCSM, the construction of the model of a simple object of the ACS FT UZ-U is performed using mappings of the compound class $\mathfrak{A}^0$. Complex units are created by actions over simple or complex entities. In the class of WP entities, the mapping is defined for the connection $\eta_k \in \mathfrak{A}^0$, for which $\forall b_i, b_j \in \mathrm{M}^*$ is determined by the connecting rule:

$$(b_i, b_j)\eta_k = (b_i, b_j)_k = b_i \eta_k b_j.$$

Here, the mapping:

$$\eta_k = (\varphi_{k1}, \varphi_{k2}, \cdots, \varphi_{k,n-1})$$

has the property of inverse morphism, and the binding of conceptual entities can be either linier:

$$(b_1, b_2,, b_n)\eta_k = \\ = b_1\varphi_{k1}b_2, b_2\varphi_{k2}b_3, \cdots, b_{n-1}\varphi_{k,n-1}b_n$$

or not. To obtain complexly connected constructions of the entities e_{kj}, compositions of maps are used and, with the help of external performers, the relationships φ_{kj} acquire an attribute meaning. Many objects $\{e_{kj}\}_j$ form a new class E_k of the WP entities.

Creating a new application and a logical database object (LDB) in the OCSM is conditionally shown in the example of Fig. 2. For a new object, it is also necessary to define an interaction interface that provides valid methods of action. Let us use b_{1j} to designate those WP entities that are connected by the relations of subordination $\rho_j \in \mathfrak{R}_V^0$. In them, the index $j=1$ correlates with the LDB – wagons, $j=2$ – WD, $j=3$ – dispatch, etc. Let K_i be filed information containing the WP passport $\{e_{ij}^{Ki}\} = E^{Ki}$ and the dislocation information D_i, characterizing states and relations with other LBD objects $\{e_{km}^{Di}\} = E^{Di}$. A number of operation entities $\{o_{lq}\} = O^Q$ are introduced further: train reception $l=1$, write-off $l=2$ from the balance $q=1$; reception $l=3$, submission $l=4$ to the site of the node $q=2$; placement $l=5$, removal $l=6$ from the wagon $q=3$. Operations without alternatives $l=0$ have the values of «transfer» at $q=4$ and receipt $q=5$. The Is (14) interface indicates possible interaction options for the objects entered.

To set the description of the interface of conditions Is in the created fragment of the ACS FT UZ, it is necessary to use the OCSM taxonomy $\left\langle E^{Ki} \cup E^{Di} \cup O^Q \cup D, \mathfrak{A}^0 \right\rangle$, with the $D \subset \mathrm{M}$ class of various characters of separators, operations, etc.:

$$Is = \left(\overset{\rho_l}{s^{1k_m}_{r_m}} a_{ij}^1 \eta_k \vee \left(\begin{matrix} (E^{Ki}\varphi_{k_1})\varphi_{k_2}(E^{Di}\varphi_{k_3}) \vee \\ \vee(o_{11}, o_{21})\varphi_{k_4} \vee (o_{32}, o_{42})\varphi_{k_5} \cdots \end{matrix} \right) \right). \tag{14}$$

Expression (14) represents paths P of constructing a knowledge model consisting of P_i as separate allowable fragments of the paths P.

The actions of deleting/adding objects of the ACS FT UZ-U will be performed by breaking certain connections and then entering/deleting the required object. Another method of deleting/adding objects is based on the inclusion of a submodel of the formation of new objects in the OCSM carrier. For this purpose, the required actions on the submodels are entered into the main OCSM model.

The mentioned procedures of the OCSM fully implement all the needs of the ontological modeling to expand the ACS FT UZ-U system, and they also supplement the corresponding knowledge models, since the nodes of the upper levels receive new attributes. New objects of the physical database are created by standard means of the ACS FT UZ-U.

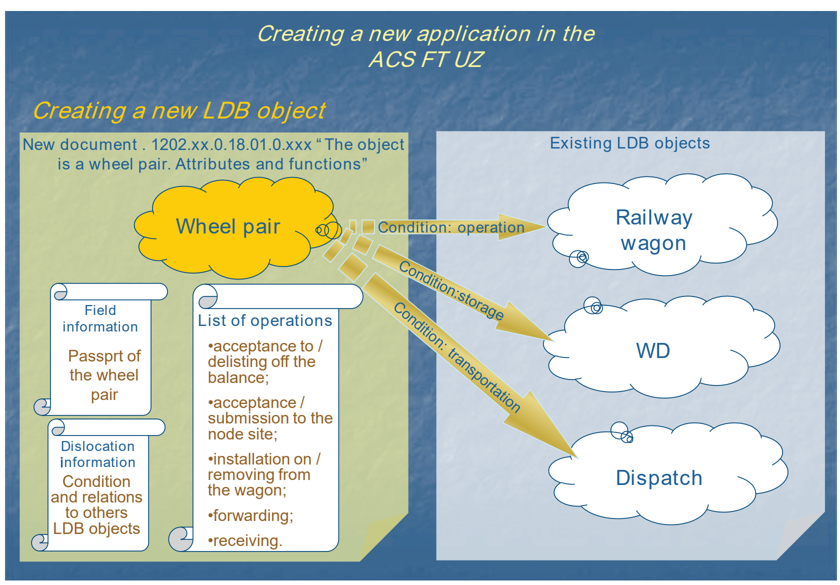

Fig. 2. The procedures for creating a new application and a logical database object of the automated system

6. Discussion of the results of developing and researching the ontological support of the constructive-synthesizing modeling

The practice of creating, implementing and maintaining modern integrated production information technologies, as well as the corresponding means of ontological support, shows the priority of ensuring the expansion and development processes of the ontological system (OS). This is due to the numerous stages of development, the long period of operation, numerous and heterogeneous categories of IS users. The implementation of the goals and objectives of the research was possible due to the use of methodological principles and methods for creating the ontological support of the CSM. They are based on a universal and customizable model for subject areas of the ontological constructive structure developed in [7]. The CSM tools, which are uniformly used in the creation of various information systems, were modified in the present study to represent the processes of expansion and development of the areas of applied ontologies, here – the domain of the ACS of the RT. Ontological support is also required by many other RTACS, for example, railway stations [26].

The presented study is specific in developing such OCSM methods that take into account the features and properties of ontological support tools for modeling the processes of expanding and developing AOs. The extensible model of the OCSM was supplemented with methods for constructing ontological entities and derivability procedures corresponding to the properties of the developed system of concepts, characterizing the individual components of the ACS that form a network or recursive structure. Moreover, the proposed OCSM tools provide opportunities for creating special system-wide methods and procedures for maintaining the IT of automated process control systems to ensure reconstruction of properties or relationships when introducing new objects into an AO. The presented example has shown the adequacy of the developed OCSM tools for the implementation of the existing procedures for expanding complex railway automated control systems.

The processes of developing information technologies and systems are diverse; they differ in formalization structures and have specific features at various stages of the life cycle. Above is the implementation of the procedures for the development of subject ontologies within the framework of the methodology of the CSM, based on the modification of a single ontological constructive structure as well as derivability procedures. At the same time, the study did not consider boundary possibilities, the «horizon» of new AO objects, and the form of the procedures to interpret the resulting structures of objects. Besides, it still remains necessary to research the questions of completeness of axiomatics and effectiveness of derivability procedures. The work mentions but does not specify the models and procedures to implement the tasks of ensuring the functioning of the OCSM (finding solutions, checking identities, and verifying equivalence of ontological objects). These restrictions show the future directions of research and development to improve the methodology of the CSM, as well as the formation of ontological tools for the OCSM to display the development processes of complex information systems.

7. Conclusion

1. The methodology of constructive-synthesizing modeling has been developed as methods and formal ontological support procedures, taking into account the properties and requirements of creating complex developing information systems. The properties of expanding and developing subject ontologies are provided by modifying the model of a unified and universal ontological constructive structure as well as derivability procedures. In contrast to the well-known, constructive calculus of the generating class of mappings and methods for constructing higher-order ontological objects were constructed to model the processes of conceptualization of developing objects.

2. Methods of inference have been developed in the constructive structure of the OCSM, which differ in the procedures of meaningful, structural and related deducibility, as well as in the developed advanced and multi-level methods for the formulation of CSM objects. On their basis, ontological support of CSM processes is provided in the context of expansion and development of the subject areas of information systems, as well as the evolution of the OCSM.

3. OCSM tools have been formed to provide support for the development of applied ontologies. By the example of the unified automated system for managing freight traffic by Ukrzaliznytsia (ACS FT UZ-U), sufficient possibilities have been shown for applying the proposed methods for the implementation of the established procedures to expand the existing automated control system of railway transport.

Acknowledgments

The presented study was carried out as part of the research work «Increasing the competitiveness of railway transport based on unified intelligent technologies for the transportation and operation of technical system parks», state registration number 0117U004392.

References

1. Palagin A. V., Petrenko N. G., Mitrofanova A. E. K voprosu postroeniya ontologicheskih sistem raznogo naznacheniya // Kompiuterni zasoby, merezhi ta systemy. 2016. Issue 15. P. 5–9.
2. Palagin A., Kryviy S., Petrenko N. Razrabotka, issledovanie i predstavlenie funkciy i operaciy na ontologiyah // Information Theories and Applications. 2015. Vol. 22, Issue 2. P. 103–114.
3. Palagin A. V. Ontologicheskaya koncepciya informatizacii nauchnyh issledovaniy // Kibernetika i sistemniy analiz. 2016. Vol. 52, Issue 1. P. 3–9.
4. Guarino N. Understanding, building and using ontologies // International Journal of Human-Computer Studies. 1997. Vol. 46, Issue 2-3. P. 293–310. doi: https://doi.org/10.1006/ijhc.1996.0091
5. Shinkarenko V. I., Il'man V. M. Konstruktivno-produkcionnye struktury i ih grammaticheskie interpretacii. I. Obobshchennaya formal'naya konstruktivno-produkcionnaya struktura // Kibernetika i sistemniy analiz. 2014. Vol. 50, Issue 5. P. 8–16.

6. Shinkarenko V. I., Il'man V. M. Konstruktivno-produkcionnye struktury i ih grammaticheskie interpretacii. II. Utochnyayushchie preobrazovaniya // Kibernetika i sistemniy analiz. 2014. Vol. 50, Issue 6. P. 15–28.
7. Skalozub V., Ilman V., Shynkarenko V. Development of ontological support of constructive-synthesizing modeling of information systems // Eastern-European Journal of Enterprise Technologies. 2017. Vol. 6, Issue 4 (90). P. 58–69. doi: https://doi.org/10.15587/1729-4061.2017.119497
8. Modeli ontolohichnoi pidtrymky avtomatyzovanykh system keruvannia zaliznychnymy vantazhnymy perevezenniamy v Ukraini / Skalozub V. V., Shynkarenko V. I., Tseitlin S. Yu., Cherednychenko M. S. // Sistemnye tekhnologii. 2017. Issue 5 (112). P. 153–165.
9. Skalozub V. V., Ceytlin S. Yu., Cherednichenko M. S. Intellektual'nye informacionnye tekhnologii i sistemy zheleznodorozhnogo transporta // Sistemnye tekhnologii modelirovaniya slozhnyh processov. Dnepr, 2016. P. 560–589.
10. Gruber T. R. Toward principles for the design of ontologies used for knowledge sharing? // International Journal of Human-Computer Studies. 1995. Vol. 43, Issue 5-6. P. 907–928. doi: https://doi.org/10.1006/ijhc.1995.1081
11. A commitment-based reference ontology for services / Nardi J. C., Falbo R. de A., Almeida J. P. A., Guizzardi G., Pires L. F., van Sinderen M. J. et. al. // Information Systems. 2015. Vol. 54. P. 263–288. doi: https://doi.org/10.1016/j.is.2015.01.012
12. Chauhan A., Vijayakumar V., Ragala R. Towards a Multi-level Upper Ontology/ foundation Ontology Framework as Background Knowledge for Ontology Matching Problem // Procedia Computer Science. 2015. Vol. 50. P. 631–634. doi: https://doi.org/10.1016/j.procs.2015.04.096
13. Ontology-Based System for Conceptual Data Model Evaluation / Kazi Z., Kazi L., Radulovic B., Bhatt M. // The International Arab Journal of Information Technology. 2016. Vol. 13, Issue 5. P. 542–551.
14. Benslimane S., Malki M., Bouchiha D. Deriving Conceptual Schema from Domain Ontology: A Web Application Reverse Engineering Approach // The International Arab Journal of Information Technology. 2010. Vol. 7, Issue 2. P. 167–176.
15. Manuja M., Garg D. Intelligent text classification system based on self-administered ontology // Turkish Journal of Electrical Engineering &Computer Sciences. 2015. Vol. 23. P. 1393–1404. doi: https://doi.org/10.3906/elk-1305-112
16. Komp'yuternaya ontologiya: zadachi i metodologiya postroeniya / Bova V. V., Leshchanov D. V., Kravchenko D. Yu., Novikov A. A. // Informatika, vychislitel'naya tekhnika i inzhenernoe obrazovanie. 2014. Issue 4. P. 44–55.
17. Pancerz K., Lewicki A., Tadeusiewicz R. Ant-based extraction of rules in simple decision systems over ontological graphs // International Journal of Applied Mathematics and Computer Science. 2015. Vol. 25, Issue 2. P. 377–387. doi: https://doi.org/10.1515/amcs-2015-0029
18. Pancerz K. Paradigmatic and Syntagmatic Relations in Information Systems over Ontological Graphs // Fundamenta Informaticae. 2016. Vol. 148, Issue 1-2. P. 229–242. doi: https://doi.org/10.3233/fi-2016-1432
19. Ontological Support for the Evolution of Future Services Oriented Architectures / Gonen B., Fang X., El-Sheikh E., Bagui S., Wilde N., Zimmermann A. // Transactions on Machine Learning and Artificial Intelligence. 2014. Vol. 2, Issue 6. P. 77–90. doi: https://doi.org/10.14738/tmlai.26.784
20. Grabusts P., Borisov A., Aleksejeva L. Ontology-Based Classification System Development Methodology // Information Technology and Management Science. 2015. Vol. 18, Issue 1. P. 129–134. doi: https://doi.org/10.1515/itms-2015-0020
21. Ontological Support for Living Plan Specification, Execution and Evaluation / Thomsen E., Read F., Duncan W., Malyuta T., Smith B. // Semantic Technology in Intelligence, Defense and Security (STIDS), CEUR. 2014. Vol. 1304. P. 10–17.
22. Towards Ontological Support for Principle Solutions in Mechanical Engineering / Breitsprecher T., Codescu M., Jucovschi C., Kohlhase M., Schröder L., Wartzack S. // Frontiers in Artificial Intelligence and Applications. 2014. P. 427–432. doi: https://doi.org/10.3233/978-1-61499-438-1-427
23. Identification of ontologies to support information systems development / Beydoun G., Low G., García-Sánchez F., Valencia-García R., Martínez-Béjar R. // Information Systems. 2014. Vol. 46. P. 45–60. doi: https://doi.org/10.1016/j.is.2014.05.002
24. The eXtensible ontology development (XOD) principles and tool implementation to support ontology interoperability / He Y., Xiang Z., Zheng J., Lin Y., Overton J. A., Ong E. // Journal of Biomedical Semantics. 2018. Vol. 9, Issue 1. doi: https://doi.org/10.1186/s13326-017-0169-2
25. Simperl E., Luczak-Rösch M. Collaborative ontology engineering: a survey // The Knowledge Engineering Review. 2013. Vol. 29, Issue 01. P. 101–131. doi: https://doi.org/10.1017/s0269888913000192
26. Zhukovyts'kyy I. Use of an automaton model for the designing of real-time information systems in the railway stations // Transport problems. 2017. Vol. 12, Issue 4. P. 101–108.