General-purpose semantic representation language and semantic space*

Valerian Ivashenko
Department of Intelligent Information Technologies
Belarusion State University of Informatics and Radioelectronics
Minsk, Republic of Belarus
ivashenko@bsuir.by

***Abstract*—In the article, models and tools that provide a unified representation of knowledge and their integration within a "semantic space" are considered. For this, the concept of a "generalized formal language" is introduced, which makes it possible to identify the relation between formal languages and known knowledge representation languages such as semantic networks for the purpose of analysis. Based on this analysis, the semantics of the languages of the unified semantic knowledge representation model is specified. A general-purpose language is also introduced as the basis for the technology of developing intelligent systems. And as a result, the concept of "semantic space" is given. The latter is focused on the usage in order to assess the quality of intelligent computer systems within the OSTIS technology. Based on the proposed models, applied problems and further prospects for the development of technologies and their components are considered.**

***Index Terms*—Semantic Space, Distensible Sets, Generalized Formal Language, Generalized String, Generalized Kleene's Closure, Set Ordination, Individual Set, Ordered Set, Unified Knowledge Representation Model, Knowledge Specification Model, OSTIS, Knowledge Integration, Introscalar product, Introscalar basis, Taxonomy optimization, Homogeneous Semantic Network, denotational semantics, operational semantics, game semantics, Holomovement, Interoperability, Convergence, Space-Time, Topological Space, Semantic Metric Space, Semantic Metric, Manifold, Becoming, Finite Structure**

I. Introduction

In the article, models and tools that provide a unified representation of knowledge and their integration within the "semantic space" in order to develop a standard for the technology of designing intelligent computer systems are considered [5]. This takes into account various aspects of integration, including the integration of data and knowledge representation levels, as well as dynamic aspects of integration that are closely related to the operational semantics of knowledge [9], [11]. As for the review of languages and models focused on a unified semantic representation of knowledge and their integration, and the corresponding approaches to solving these problems, it is proposed to refer to the work [10] for access to review materials. In this paper, the review part is dedicated to the issues and history of refining the concept of "semantic space". The need to consider knowledge representation languages that provide a unified semantic representation of knowledge is conditioned be the need to represent elements of the semantic space.

In the article, the following will be considered: 1) the formulation of the problem of forming concepts that can express the meaning of the term "semantic space", the formulation of the identified problems that need to be overcome in order to solve the problem; generalization of the concept of formal language [20], and mathematical foundations for the model representation of texts of knowledge representation languages in the form of texts of generalized formal languages; 2) identifying requirements for the alphabet of a knowledge representation language focused on the semantic (meaningful) representation of knowledge, which are the rationale for choosing such a language and identifying such a language among generalized formal languages, as well as languages of the unified semantic knowledge representation model; 3) the language with its core proposed on the basis of the identified abstractions, which provide a unified semantic representation of knowledge in intelligent systems within an open technology for the development of intelligent computer systems; 4) historically formed approaches to the genesis of the concept of "semantic space" and those close to it, including some abstractions of "space" in mathematics; 5) proposed approaches, formalisms, and models for the becoming of concepts capable of expressing the meaning of the term "semantic space" in accordance with the model of a unified semantic representation of knowledge; 6) application of some of the proposed problem-solving models of the level of knowledge control in ontologies on the example of taxonomies; 7) a conclusion containing the main results and prospects for the application of the proposed models.

The semantic space implies the inclusion of various meanings, therefore, an important problem on the way to learning the corresponding "semantic space" concept is the integration of knowledge and the represented meanings.

In knowledge-based systems [22], four directions of integration (of knowledge and models of their representation) are distinguished:

- vertical (introspection);
- horizontal profile (knowledge engineering);
- horizontal frontal (unification);

* The author is grateful to Dr. Tech. Sc., Professor V.V. Golenkov for clarifying the formulation of the problem and the topic of this work, as well as for providing additional materials to the OSTIS standard. The author is also grateful to E. Bobyor, A. Shichko and D.A. Ivashenko for technical assistance and support in the work on the article.

- the direction of continuous integration (training and adaptation).

Problems of integrating knowledge into a single semantic space (unification) are:

- the availability and usage of non-finite structures in models and methods for the representation and processing of knowledge and the formalization of meanings, which make it difficult or exclude the algorithmization of working with such representations, including their analysis and unification;
- uncertainty of models reflecting the results of vertical integration of models and information representation languages, including formal languages, and providing consistent (continuous) integration of texts of dynamic knowledge representation models;
- the presence of different types of knowledge semantics: game [23], [24], [9], operational [9], [25], denotational [5], [26], model [8], etc., requiring correlation in order to identify the equivalence of knowledge;
- dynamic, non-monotonic nature of knowledge, the presence of non-factors [1] in knowledge, the presence of reliable knowledge and knowledge about the unreliable and hypothetical.

Let us formulate the main problems:

- search, comparison, and grounding of the chosen means for knowledge representation;
- search for models for representing and analyzing the structure of the text elements in the languages of the selected knowledge representation model;
- application of the results of the analysis to solve the problems of the level of knowledge operation.

To solve the problem and overcome the identified problems, it is proposed to use methods of mathematical modeling, including embedding (isomorphic injective mapping) of some mathematical structures into others, application of models and methods of theories of formal languages and formal systems, combinatorics, and discrete optimization.

II. Generalization of formal languages and semantic network knowledge representation models

As it is known, the language as a "sign system" is designed to perform "communicative", "cognitive" ("epistemological"), "representative", and other "functions" [27]. Within the language, signs are organized into texts, which have a sequential structure performing a "communicative function": sounds, words of oral speech are organized into a sequence, just like letters, written signs are into lines. Mathematically, a string is an "oriented" ("ordered") "set" (of characters) [29]. The "set" itself is a mathematical abstraction of thinking reflecting its ability to generalize and move from parts to the whole. The question of correlating "oriented sets" and "unoriented sets" belongs to the foundation of mathematics [28]. This question is retaining its actuality. The need to consider this issue is related to the combination of classes "oriented" and "unoriented sets" used in the representation of knowledge, and corresponding to them within the "semantic space" ("semantic metric"), in which it is necessary to correlate elements of these classes (Fig. 1 and 2). There are known particular solutions to this issue for the concept of "oriented" ("ordered") "pairs" proposed by N. Wiener, F. Hausdorff, K. Kuratovsky, and others.

Definition according to N. Wiener [30]:

$$\langle \chi, \gamma \rangle_W \overset{def}{=} \{\{\{\chi\}, \emptyset\}, \{\{\gamma\}\}\}$$

Definition according to F. Hausdorff [30]:

$$\langle \chi, \gamma \rangle_{12} \overset{def}{=} \{\{\chi\} \cup \{1\}\} \cup \{\{\gamma\} \cup \{2\}\}$$

The disadvantage is that either $(\{\chi\} \cap \{2\}) \times (\{\gamma\} \cap \{1\}) = \emptyset$ or $\langle 2, 1 \rangle_{12} = \{\{1, 2\}\}$. In addition, there are lacks for the technical implementation.

Definition according to K. Kuratovsky [31]:

$$\langle \chi, \gamma \rangle_K \overset{def}{=} \{\{\chi\}\} \cup \{\{\chi\} \cup \{\gamma\}\}$$

The disadvantage is that $\langle \chi, \chi \rangle_K = \{\{\chi\}\}$. There are also lacks for the technical implementation.

Other definitions:

$$\langle \chi, \gamma \rangle_{reverse} \overset{def}{=} \{\{\chi\} \cup \{\gamma\}\} \cup \{\{\gamma\}\}$$

Disadvantages are similar to $\langle \chi, \gamma \rangle_K$.

$$\langle \chi, \gamma \rangle_{short} \overset{def}{=} \{\chi\} \cup \{\{\chi\} \cup \{\gamma\}\}$$

The disadvantage is that either $(\{\chi\} \cap \{\{\gamma\}\}) \times (\{\gamma\} \cap \{\chi\}) = \emptyset$ (axiom of foundation) or $\langle \{\gamma\}, \chi \rangle_{short} = \{\chi\}$. It is also a disadvantage from the point of view of type theories that the elements in the set will have a different type, whereas χ and γ are of the same type. For ordinal numbers constructed according to von Neumann, [33] we have $\langle \emptyset, \emptyset \rangle_{short} = \langle 0_{Ord}, 0_{Ord} \rangle_{short} = 2_{Ord}$.

$$\langle \chi, \gamma \rangle_{01} \overset{def}{=} \{\{\chi\} \cup \{0\}, \{\gamma\} \cup \{1\}\}$$

Disadvantages are similar to $\langle \chi, \gamma \rangle_{12}$.

Let

$s(\chi) \overset{def}{=} \{\emptyset\} \cup \{\{\tau\} | \tau \in \chi\}$ then, according to M. Morse, an oriented pair, triple, etc. will be [32]:

$\langle \chi, \gamma \rangle_M \overset{def}{=} (\{0\} \times s(\chi)) \cup (\{1\} \times s(\gamma))$,

$\langle \chi, \gamma, \zeta \rangle_M \overset{def}{=} (\{0\} \times s(\chi)) \cup (\{1\} \times s(\gamma)) \cup (\{2\} \times s(\zeta))$.

The disadvantage is that the Cartesian product uses a pair in accordance with the definition of K. Kuratovsky.

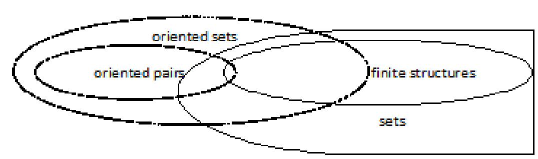

Fig. 1. A correlation diagram for the set and oriented set concepts.

As shown, the above proposals do not solve the issue in general or have their drawbacks. Other definitions rely on the existence of infinite structures (Fig. 1).

Let us define an oriented set σ in the following way :
$\sigma \overset{def}{=} \bigcup_{\iota=1}^{|\sigma|} \left\{ (|\sigma| - \iota + 1)^{\{\sigma_\iota\}} \right\}_\iota$.

An oriented set σ is the combination of individual sets of order ι of all ordinations of its components singletons σ_ι to the base of $|\sigma| - \iota + 1$, where ι takes values from 1 to $|\sigma|$.

Ordination of the set σ to the base 1:
$1^\sigma \overset{def}{=} \sigma$.

Ordination of the set σ to the base $\iota + 1$:
$(\iota + 1)^\sigma \overset{def}{=} \{\iota^\tau \,|\tau \subseteq \sigma\}$.

It should be noted that the logarithm of a set σ to the base 2 is its boolean 2^σ [34] .

An individual set of order 1 of element χ:
$\{\chi\}_1 \overset{def}{=} \{\chi\}$.

An individual set of order $\iota + 1$ of element χ:
$\{\chi\}_{\iota+1} \overset{def}{=} \{\{\chi\}\}_\iota$.

It should be noted that an oriented set of one element is : $\langle\chi\rangle \overset{def}{=} \left\{1^{\{\chi\}}\right\}_1 = \{\{\chi\}\}$, an oriented set of two elements coincides with an oriented pair according to N. Wiener :
$\langle\chi, \gamma\rangle \overset{def}{=} \left\{2^{\{\chi\}}\right\}_1 \cup \left\{1^{\{\gamma\}}\right\}_2 = \{\{\{\chi\}, \emptyset\}\} \cup \{\{\{\gamma\}\}\} = \{\{\{\chi\}, \emptyset\}, \{\{\gamma\}\}\}$, and the oriented triple is :
$\langle\chi, \gamma, \zeta\rangle \overset{def}{=} \left\{3^{\{\chi\}}\right\}_1 \cup \left\{2^{\{\gamma\}}\right\}_2 \cup \left\{1^{\{\zeta\}}\right\}_3 = \{\{\{\{\chi\}, \emptyset\}, \{\emptyset\}\}, \{\{\{\gamma\}, \emptyset\}\}, \{\{\{\zeta\}\}\}\}$.

Fig. 2. A correlation diagram for the set and defined oriented set concepts.

Thus, according to this definition, oriented sets are sets and finite structures (Fig. 2).

If it is necessary for some non-empty oriented sets to be non-oriented, it is possible to define an own subclass as unoriented sets, according to the definitions:
$\bigcup_{\tau\in\cdot\chi} \{[\tau]\} \overset{def}{=} \bigcup_{\tau\in\cdot\chi} \left\{2^{\{\tau\}}\right\}_{\kappa(\chi)}; \quad \chi \overset{def}{=} \bigcup_{\tau\in\cdot\chi} \left\{2^{\{\tau\}}\right\}_{\kappa(\chi)}$,
where $\kappa(\chi) \overset{def}{=} 1$.

For example:
$\{[\chi]\} \cup \{[\gamma]\} \overset{def}{=} \{\{\{\chi\}, \emptyset\}\} \cup \{\{\{\gamma\}, \emptyset\}\}$.

Knowledge representation languages (formal languages) and semantic networks with a graph structure are used to represent knowledge, but there is no known general model in which these knowledge representation means can be compared in order grounding the choice of anyone of them.

Let us extend a class of languages beyond the known class of formal languages in order to ground the choice of the means (language) of knowledge representation to overcome the problem of the lack of known models that reflect the results of vertical integration [11] of models and information representation languages and provide consistent (continuous) integration [11] of texts of dynamic knowledge representation models (Fig. 4).

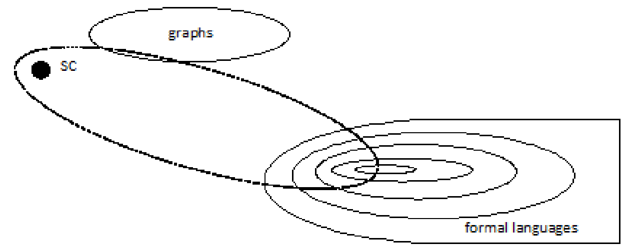

Fig. 3. The correlation of syntax knowledge representation means.

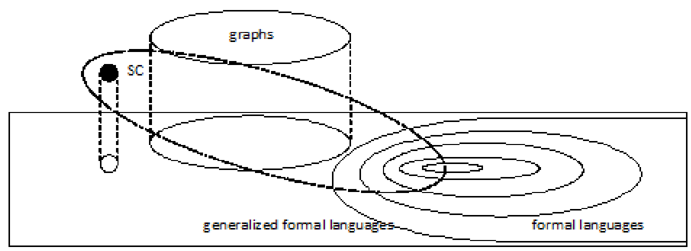

Fig. 4. The correlation of generalized formal languages and knowledge representation means.

We introduce the concept of generalized Kleene closure and generalized formal language [3], [20] in order to extend the class of languages beyond formal languages. The expediency of this extension is conditioned by the need to endow knowledge representation languages with associative properties. The associative properties of a language are reduced to the presence of associations in its texts. The simplest associations are abstract connections (connectives), which are considered mathematically as sets (or directed sets). We will also consider only oriented sets as associations (strings and generalized strings, i.e. strings whose components can only be symbols of the alphabet of the language, strings from them and other generalized strings) in order for the texts of the language to remain finite and the "communicative function" of the language to be preserved.

As it is known, a formal language Λ is a subset of the Kleene closure [20] of its alphabet. A:
$\Lambda \subseteq \mathrm{A}^*/\left(\mathrm{A}/\mathrm{A}^1\right)$,
where the Kleene closure of the alphabet A (the definition is slightly modified to preserve the extensiveness and idempotency of the closure operator):
$\mathrm{A}^{*\Sigma} \overset{def}{=} \mathrm{A} \cup \left(\bigcup_{\iota\in\mathbf{N}/\{0\}} \mathrm{A}^\iota\right)^{\oplus\Sigma}$.

$$\mathrm{A}^* \overset{def}{=} \mathrm{A}^{*\mathrm{A}}$$

The operation $\oplus_\mathrm{A}$ for each element χ of the set to which it is applied, if γ is a string, "insert" in order all the components of the string χ, which is a component of the string χ, into this string χ instead of (γ) if each inserted component belongs to A and the string γ itself does not belong to A. Empty lines are excluded from the string unless they are in A.

Here and below:

- $A \times B$ is the Cartesian product of set A and set B;

- A^n is the Cartesian power n of set A;
- B^A is the exponential (a set of completely defined functions with domain A and domain B);
- B_+^A is the extended exponential (set of functions with origin A and range B ($B^A \subseteq B_+^A$));
- 2^A is the boolean of set A (set of all subsets of set A ($B_+^A \subseteq 2^{A\times B}$)) ;
- R^{-1} is the inverse binary relation to R;
- $R \circ S$ is the composition of binary relations R and S;
- R° is the transitive closure of the binary relation R.

A generalized formal language Λ is a subset of the generalized Kleene closure of the alphabet A:

$$\Lambda \subseteq \mathrm{A}^{(*^*)} / \left(\mathrm{A}/\mathrm{A}^1\right).$$

The generalized Kleene closure of the alphabet A satisfies the following definitions:

$$\mathrm{A}^{(*^*)} \stackrel{def}{=} \bigcup\nolimits_{\iota\in\mathbf{N}/\{0\}} \mathrm{A}^{(*^\iota)};$$

$$\mathrm{A}^{(*^{\iota+1})} \stackrel{def}{=} \left(\mathrm{A}^{(*^\iota)} \cup \mathrm{A}\right)^*;$$

$$\mathrm{A}^{(*^1)} \stackrel{def}{=} \mathrm{A}^*.$$

The generalized Kleene closure is extensive:

$$\mathrm{A}^{(*^*)} \cup \mathrm{A} = \mathrm{A}^{(*^*)}.$$

The generalized Kleene closure is monotonic:

$$\mathrm{A}^{(*^*)} \subseteq (\mathrm{A} \cup \Delta)^{(*^*)}.$$

The generalized Kleene closure is idempotent:

$$\left(\mathrm{A}^{(*^*)}\right)^{(*^*)} = \mathrm{A}^{(*^*)}.$$

The cardinalities of the generalized Kleene closure and the Kleene closure of the non-empty either finite or countable alphabet A are equal (aleph-0).

A generalized formal language Λ is called a symmetric language [3] if and only if for any Δ:

$$((\emptyset \subset \Delta^n \cap \Lambda) \to (\Delta^n \subseteq \Lambda)).$$

Symmetric languages, as a rule, correspond to languages in which the order of transmission of text elements is not essential for the preservation of the transmitted meaning. Among such languages, there are languages focused more on the performance of "cognitive" and "representative" "functions" rather than "communicative", i.e. refer to representation languages.

A generalized formal language Λ is called an associative language [3] if and only if:

$$\exists \mathrm{T} \exists \Delta \exists n \left(\emptyset \subset (\Delta^n \cup \mathrm{T})^* \cap \Lambda\right).$$

A generalized formal language Λ is called a pseudograph (graph) language if and only if A exists and for any Δ, T, n:

$$\Lambda \subseteq \left(\mathrm{A}^2 \cup \mathrm{A}\right)^*;$$

$$\left(\left(\emptyset \subset \left(\Delta^2 \cup \mathrm{T}\right)^n \cap \Lambda\right) \to (\Delta \subseteq \mathrm{T})\right).$$

A syntactically distinguishable fragment of text is a fragment of text (subtext) whose structure differs from others or its position in such language texts as (semantically equivalent) permutations of elements of one of them differs from other fragments (is unique) relative to the structure of all such texts [8].

The consideration of generalized formal languages as a model for knowledge representation languages is conditioned by:

- the finite structure of texts;
- the ability to consider knowledge representation means with a more complex structure of associations than in graphs, in particular, the ability to construct texts that one-to-one correspond to abstract simplicial complexes [3], [35];
- the possibility of associative coding of meaning (representation of knowledge) by associations of a given level and below within the hierarchical structure of the text, which allows for continuous integration (monotonicity) of knowledge represented in this way, changing only the structure of connections in associations of upper levels (in the limit, only the order of elements in the text) but not the state of text elements during processing information in dynamic (procedural, non-monotonic) models of knowledge processing;
- the possibility of syntactic (hierarchical) inclusion of texts of one language into the texts of other languages as a whole (without including their parts and without reflecting their structure) which allows working with texts of several languages from the position of a single model in a single syntactic space and ensure representing the results of processing texts of one language in another, which is typical for vertical integration processes [11];
- the natural way of the syntax reflection of such artificial intelligence languages as LISP [36], [8].

Statement. There are $p*(p+1)/2$ connected distinguishable text fragments for texts with p characters with a linear association structure [8].

Statement. There are $\left(\left(\left(\sqrt{5}+1\right)/2\right)^{2*p+1} - \left(\left(1-\sqrt{5}\right)/2\right)^{2*p+1}\right)/\sqrt{5} - 1$ distinguishable text fragments for texts with p characters with a linear association structure [8].

Statement. There are at least $(\lceil (q-p)/(2*p) \rceil + p - 1)!/(p! * (\lceil (q-p)/(2*p) \rceil - 1)!)$ connected distinguishable text fragments for connected texts with p symbols with a non-linear structure of q ($q \geq p$) associations [8].

The choice of pseudograph (graph) languages is determined by (Fig. 5):

- the ability to build a non-linear association structure, which makes it possible to achieve a number (exponentially dependent on the size of the text) of connected syntactically distinguishable text fragments which are potential answers to questions to the knowledge base, in contrast to languages with a linear association structure (corresponding to formal languages that are not associative), which have

only a quadratic number of connected syntactically distinguishable fragments of texts [8];
- the fact that hypergraph languages do not qualitatively raise the number of connected syntactically distinguishable fragments in their texts but cause difficulties in implementation.

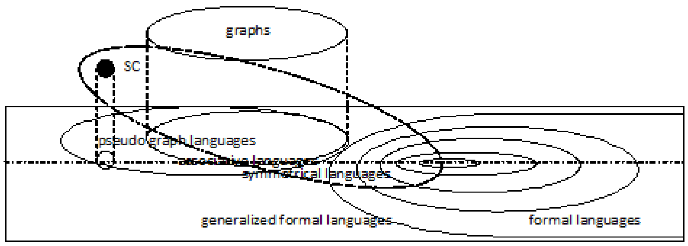

Fig. 5. Symmetrical, associative, and pseudograph languages.

III. Unified representation

Requirements for the knowledge representation language [5]:

- semantic representation, each element of the language text, with the exception, perhaps, of syntactic connectives of their incidence, should be a designation (sign) of the entity;
- universality of representation:
  - representation of phenomena of arbitrary structure, including associations of an unlimited number of elements,
  - representation of knowledge and expression of semantics of any kind;
- simple syntactic structure of texts;
- minimum number of alphabet elements types, unification of representation;
- ensuring the basic properties of knowledge representation (the possibility of eliminating synonymy and bringing it to homogeneous (refined) semantic networks).

A simple syntactic structure of texts means a case when the number of incident connectives is expressed as no more than a linear dependence on the number of symbols in the text, i.e. the relation of the number of incident connectives to the number of symbols in the text does not exceed a certain constant. So, for example, if there are m nodes and n arcs in the text, each of which has three connectives of incidence, the dependence has the following form $3 * n$, and the relation is expressed by the formula $\frac{3*n}{n+m}$, what is less than three for any natural m and n. In the case of a complex syntactic structure, the number of incident connectives in the texts of pseudograph languages can be expressed as a dependence on the number of designations in the text, reaching n^2.

Thus, the alphabet close to the minimum one I should contain:

- designations of entities (including connectives) that can have an unlimited number of incident designations (and corresponding incident connectives);
- designations (of basic connectives), which can have a limited number of incident designations (and corresponding incident connectives), in the simplest case — equal to two;
- designations (of common connectives) that can have a limited number of incident designations (and corresponding incident connectives), in the simplest case — equal to two.

Table I
Comparison of languages with different types of alphabet elements

	one kind			***two kinds***				***more than two kinds***
syntactic restrictions	-	+		-	+			+
simple syntactic structure	-	+		-	+			+
semantic representation	+	+	-	+	+	-	+	+
arbitrary phenomena structure representation	+	-	+	+	-	+	+	+
ability to represent any knowledge and to express semantics of any kinds	-			+			-	+

Basic and common connectives are assumed to be oriented, since unoriented connectives are relatively easy to define with a pair of oriented connectives and the designation of an entity that does not denote a basic or common connective. In order to define an oriented connective through unoriented connectives, at least three unoriented connectives and at least two entity designations are required that do not denote the basic or common unoriented connectives.

Thus, the alphabet of the required knowledge representation language (core) must specify designations of at least three types: vertices ($V_{E_{SC}}$) and two types of arcs ($B_{E_{SC}}$, $C_{E_{SC}}$):

$$E_{SC} = V_{E_{SC}} \cup B_{E_{SC}} \cup C_{E_{SC}}.$$

Each designation may belong to (be a member of) more than one type, however, for each occurrence of the designation in the text of the required knowledge representation language, of which this designation is an element, its type is determined unambiguously.

For arcs, it is possible to define a single view as the union of both types of arcs:

$$E_{E_{SC}} = B_{E_{SC}} \cup C_{E_{SC}}.$$

If we consider the designations of the uncertain type $U_{E_{SC}} \subseteq E_{SC}$, convenient when representing knowledge in the presence of such NON-factors [1] as incompleteness, then the remaining types of designations can be expressed as:

$$V_{E_{SC}} = U_{E_{SC}} / E_{E_{SC}},$$

$$C_{E_{SC}} = E_{E_{SC}}/B_{E_{SC}}.$$

The alphabet of the required language of knowledge representation contains designations of the types:

- elements of the uncertian type $U_{E_{SC}}$;
- arcs of an unified type $E_{E_{SC}}$;
- basic arcs $B_{E_{SC}}$.

The alphabet of the extension of the required language for knowledge representation differs in the contents of a larger number of types of designations (elements):

- elements of an uncertain type (vertexes) $U_{E_{SC}}$;
- nodes $V_{E_{SC}}$;
- arcs of permanent membership $P_{E_{SC}}$;
- arcs of temporal actual membership (basic arcs) $A_{E_{SC}}$ ($B_{E_{SC}}$);
- arcs of temporal phantom membership $T_{E_{SC}}$;
- arcs of fuzzy membership (arcs of an unified type) $F_{E_{SC}}$ ($E_{E_{SC}}$);
- arcs of temporal phantom non-membership $H_{E_{SC}}$;
- arcs of temporal actual non-membership $G_{E_{SC}}$;
- arcs of permanent non-membership $N_{E_{SC}}$.

The listed types of elements are connected by the following relations:
$U_{E_{SC}} \cup V_{E_{SC}} \cup E_{E_{SC}} \subseteq E_{SC}$, where
$P_{E_{SC}} \cup A_{E_{SC}} \cup T_{E_{SC}} \cup F_{E_{SC}} \cup H_{E_{SC}} \cup G_{E_{SC}} \cup N_{E_{SC}} = E_{E_{SC}}$,

- a node cannot be an arc: $V_{E_{SC}} \cap E_{E_{SC}} = \emptyset$;
- the arc of permanent membership cannot be the arc of temporal membership, the arc of permanent membership cannot be the arc of temporal non-membership, the arc of permanent non-membership cannot be the arc of temporal membership, the arc of permanent non-membership cannot be the arc of temporal non-membership: $(P_{E_{SC}} \cup N_{E_{SC}}) \cap (A_{E_{SC}} \cup T_{E_{SC}} \cup H_{E_{SC}} \cup G_{E_{SC}}) = \emptyset$;
- the arc of permanent non-membership cannot be the arc of permanent membership: $P_{E_{SC}} \cap N_{E_{SC}} = \emptyset$.

Such types of elements and their relations are conditioned by the need to solve the problems of horizontal profile integration [11] by unifying the representation.

When there is mapping into generalized formal languages for the purpose of vertical integration of languages (texts) for representation and reduction of texts to the fundamental alphabet A_{SC}, the following is true:

$$E_{SC} \tilde{\subset} A_{SC}{}^{(*^*)}$$
$$E_{SC}{}^2 \tilde{\subset} A_{SC}{}^{(*^*)}.$$

The model of the unified semantic representation of knowledge [8] is set by a triple:

$$\langle S_{SC}, R_{SC}, F_{SC} \rangle$$

The model languages of the unified semantic representation of knowledge are defined based on the elements of the alphabet and syntax, the set of all texts forms a general sc-language (Semantic Code, L_{SC} language), as well as the set of all its subsets ($S_{SC} = 2^{L_{SC}}$) is sc-languages (sc-sublanguages):

$$L_{SC} \tilde{\subset} A_{SC}{}^{(*^*)}.$$

The syntax of the model languages for the unified semantic representation of knowledge (sc-languages) [8] describes the properties of connectives of incidence relations in the alphabet elements of these languages in their texts. Two incidence relations for elements of the alphabet of these languages are distinguished I_{SC} and C_{SC}:

$$I_{SC} \tilde{\subseteq} E_{SC}{}^2;$$
$$C_{SC} \subseteq I_{SC}.$$

Based on these relations I_{SC}, C_{SC} one incidence relation R can be determined:

$$R = (I_{SC} - C_{SC})^{-1} \cup I_{SC};$$
$$R \tilde{\subseteq} E_{SC}{}^2;$$
$$L_{SC} \tilde{\subseteq} (R \cup E_{SC})^*.$$

The syntax of sc-languages can be set as follows. For each text S of sc-languages and the set of all its components and only them, X ($S \tilde{\in} X^n$):

$$(\forall Y\, ((Y \subset X) \to (\neg\, (S \tilde{\in} Y^n)))),$$

there will be such sets of incident connectives I, C and the set of occurrences of designations T (terminals), V (nodes), E (edges), A (arcs), and B (basic arcs), that:

$$|I \cup C| + |T \cup V \cup E| = n; I \tilde{=} I_{SC} \cap X; C \tilde{=} C_{SC} \cap X;$$
$$A \tilde{=} E_{E_{SC}} \cap X; B \tilde{\subseteq} (P_{E_{SC}} \cup A_{E_{SC}}) \cap X; T \cup V \cup E \tilde{\subseteq} X;$$
$$V \tilde{\subseteq} V_{E_{SC}} \cap X; I \cap C = C; V \cap A = \emptyset; E \cap A = A; A \cap B = B,$$

neither terminals nor nodes that are not edges are incident to each other:

$$I \cap ((T \cup V/E) \times (T \cup V/E)) = \emptyset;$$

any edge is incident to at least one designation:

$$\forall e\, ((e \in E) \to (\emptyset \subset I \cap (\{e\} \times (V \cup T \cup E))));$$

no more than two designations are incident to any edge:

$$\forall e\, ((e \in E) \to (|I \cap (\{e\} \times (V \cup T \cup E))| \leq 2));$$

one (second) designation is incident to any arc:

$$\forall e\, ((e \in A) \to (|C \cap (\{e\} \times (V \cup T \cup E))| = 1));$$

at least one node is incident to any edge, if it is a basic arc, the element that is not incident to it (not the second one) is a node:

$$\forall e ((e \in B) \to ((\emptyset \subset I \cap (\{e\} \times V)) \wedge ((I - C) \cap (\{e\} \times V) = (I - C) \cap (\{e\} \times (V \cup T \cup E))))).$$

There are interpretations of the texts of model languages for the unified semantic representation of knowledge as texts of the symmetric (pseudo-) graph (or multipseudograph) language (see Fig. 6, Fig. 7, and Fig. 8). Where $k + 1$ order associations correspond to connectives of incidence relations and sc-elements (designations in the texts of model languages for the unified semantic knowledge representation) correspond to k order associations of the fundamental alphabet A_{SC}.

When representing connectives of both incidence relations in the texts of a generalized formal language, duplication of connectives is allowed (multipseudograph), the second occurrence of the connective corresponds to belonging to the second relation, each designation corresponds to the vertex of the multipseudograph (Fig. 6):

$$\langle\langle a, b\rangle, \langle a, e\rangle, a, b, e, \langle a, e\rangle\rangle .$$

Рис. 6. Multipseudograph representation (right) of SC-text (left)

Fig. 7. Pseudograph representation (right) of SC-text (left)

Transformation of arcs of a unified type (sc-arcs), to which the nodes (sc-nodes) are incident (Fig. 7).

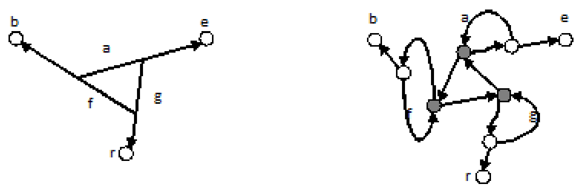

Fig. 8. Pseudograph representation (right) of SC-text with incident arcs (left)

Transformation of arcs of a unified type (sc-arcs), to which arcs of a unified type (sc-arcs) are incident (Fig. 8).

Transition from text to oriented pseudograph (Fig. 7 and Fig. 8):

- mapping the vertices of elements (sc-elements);
- mapping of arcs to connections (connectives) of incidence.

To reduce "changes in state" of "memory elements" to "changes in the connections between them" when representing texts of sublanguages of symmetric associative (pseudo)graph (or multipseudograph) language, it can be agreed that the actual elements (sc-elements, sc-arcs) are separated from the phantom elements (sc-elements, sc-arcs) by connectives of incidence relations (some are before those, and others are after).

Representation for connectives of relations of actual and phantom membership (Fig. 9) in the texts of a generalized formal language:

$$\langle\langle a, b\rangle, \langle a, e\rangle, \langle f, r\rangle, \langle f, a\rangle, a, b, e, r, \langle a, e\rangle, \langle f, a\rangle, f\rangle .$$

Relations of the unified semantic knowledge representation model [8]:

- sublanguage (sc-sublanguage),
- injective language mapping (of sc-languages).

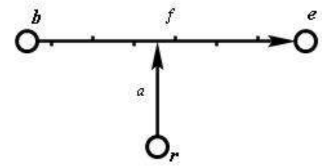

Fig. 9. SC-language text with connectives of actual and phantom membership

Features (functions) of the model languages of the unified semantic representation of knowledge (sc-languages) [8]:

- proper and non-proper sublanguage key elements for (pairs of) languages (language and sublanguage);
- semantic neighborhoods of key elements in sublanguage texts for language pairs;
- semantic interpretations of text elements.

When representing arcs of a unified type (sc-arcs), it can be assumed that they denote a pair $\langle\langle x, \mho\rangle, y\rangle$.

IV. Internal language of ostis-systems

The internal language of ostis-systems – an ***SC-code*** (Semantic Computer code) – is a model language of unified semantic representation, that is, the language of unified semantic representation of knowledge in the memory of intelligent computer systems [5].

An ***SC-code*** is [5]:

- an abstract language, that is, a language for which the way of representing symbols (syntactically elementary fragments) that are part of the texts of this language is not specified, but only the alphabet of these symbols is specified, that is, a family of character classes that are considered syntactically equivalent to each other;
- a pseudograph language;
- a universal language that provides internal representation and structuring of all (!) knowledge used by the ostis-system in the course of its functioning, and is the result of unification (refinement) of syntax and denotation semantics of semantic networks.

Graph language (pseudograph language) is a language, each text of which [5]:

- is defined by a set of elementary fragments (symbols) included in it, which, in turn, consists of a set of nodes (vertices), possibly of syntactically different types, and a set of connective designations, which may also belong to different syntactically distinguished classes;
- is defined in the general case by several relations of the incidence of connective designations with the components of these connectives (in this case, the specified components in the general case can be not only nodes but also connective designations).

Each abstract language can be matched with a family of real languages that provide an isomorphic real representation of the texts of the specified abstract language by clarifying the ways of representation (images, encoding) of the symbols included in these texts, as well as by clarifying the rules for establishing syntactic equivalence of these symbols. In all other respects, the syntax and denotational semantics of these real languages are completely similar and correspond to the syntax and denotational semantics of the corresponding abstract language [5].

The universality of the ***SC-code*** is also ensured by the fact that the elements of sets denoted by the elements of ***SC-code*** texts can be signs of the described entities of any kind, including signs of connections between the described entities and/or their signs [5].

Texts of the ***SC-code*** are graph structures of an extended form, in which the designations of the described connections can connect not only the vertices (nodes) of the graph structure but also the designations of other connections [5].

The ***SC-code*** is the basic universal language of the internal representation of knowledge in the ostis-systems memory (the basic internal language of ostis-systems) [5], this means that it is the maximum internal language of ostis-systems, in relation to which all other (specialized) internal languages are its sublanguages (subsets), that is, it is a set of all possible text of the ***SC-code*** (sc-texts). The signs (designations) of all entities described in sc-texts (texts of the ***SC-code***) are represented as syntactically elementary (atomic) fragments of sc-texts and, therefore, do not have an internal structure in the same sc-text, not consisting of simpler fragments of sc-text, such as names (terms), which represent the signs of the described entities in familiar languages and consist of letters. Names (terms), natural language texts, and other information constructions (generalized strings) that are not sc-texts can be contained in sc-elements included in the sc-text as files described (specified) by sc-texts [5], [3]. Thus, the knowledge base of an intelligent computer system built on the basis of the ***SC-code*** may contain names (terms) denoting some of the described entities in the form of corresponding files. Each sc-element will be called an internal designation of some entity, and the name of this entity, in the form of a file (sc-file), will be called an external identifier (external designation) of this entity. An external identifier can be not only a name (term) but also a hieroglyph, a pictogram, a voiced name, a gesture. It should be particularly noted that the external identifiers of the described entities in an intelligent computer system built on the basis of the ***SC-code*** are used for: (1) analyzing information coming into this system from outside from various sources and entering (understanding and immersing) this information into the knowledge base, (2) synthesis of various messages addressed to various subjects (including users).

Texts of the ***SC-code*** (sc-texts) generally have a pseudograph (graph, nonlinear) structure, since the sign of each described entity can be incident to an unlimited number of other signs, since each described entity can be connected by an unlimited number of connections with other described entities [5].

The knowledge base, represented by the text of the ***SC-code***, is a graph structure of a special kind, the alphabet of elements of which includes many elements of the explained type, many nodes, many basic arcs – arcs of a specially highlighted type that provide structuring of knowledge bases, as well as [8] a set of arcs of permanent non-membership, a set of arcs of temporal actual non-membership, a set of arcs of temporary phantom non-membership, a set of arcs of fuzzy membership, a set of arcs of phantom membership, a set of arcs of permanent membership, a set of special nodes, each of which has content that is a file stored in the memory of an intelligent computer system. The structural feature of this graph structure is that its arcs and edges can connect not only a node with a node but also a node with an arc or an arc with another arc [5].

An arc is the designation of a binary oriented connective between two entities. An arc of a special kind (base arc) is a sign of connection between a node denoting a certain set of elements of the graph structure under consideration and one of the elements of this graph structure that currently belongs to the specified set. At the same time, the connections denoted by the elements of the graph structure under consideration can be permanent (always existing) and temporal (connections that correspond to the period of their existence) [5].

In the considered graph structure, which is a representation of the knowledge base in the ***SC-code***, there may but should not exist different elements of the graph structure denoting the same entity. If a pair of such elements is detected, then these elements can be pasted together (equated). Thus, the synonymy of internal designations in the knowledge base of an intelligent computer system built on the basis of the ***SC-code*** is undesirable. At the same time, the synonymy of external designations is considered as a normal phenomenon [5].

In addition to files (***sc-files***) representing various external designations (names, hieroglyphs, pictograms), files of various texts (books, articles, documents) can be stored in the memory of an intelligent computer system built on the basis of the ***SC-code***, notes, comments, explanations, drawings, pictures, schemes, photographs, video and audio materials [5].

Any entity, that is capable of having a designation, in the text of the ***SC-code*** can be associated with an sc-element denoting a set to which only the designation of this entity belongs. This is one of the factors that ensure the universality of the ***SC-code***. We emphasize that sc-elements are not just designations but designations that are elementary (atomic) fragments of a sign construction, i.e. fragments whose detailed structure is generally not required for "reading" and understanding this sign construction [5].

The text of the ***SC-code***, like any other graph structure, is an abstract mathematical object that does not require detailing (refinement) of its encoding in the memory of a computer system (for example, in the form of an adjacency matrix, an incidence matrix, a list structure). But such detailing will be required for the technical implementation of the memory in which sc-texts are stored and processed [5].

The most important additional property of the ***SC-code*** is that it is convenient not only for the internal representation of knowledge in the memory of an intelligent computer system but also for the internal representation of information in the memory of computers specifically designed to interpret semantic models of intelligent computer systems. That is, the ***SC-code*** defines the syntactic, semantic, and functional principles of organizing the memory of next-generation computers focused on the implementation of intelligent computer systems – the principles of organizing graphodynamic associative semantic memory [5].

The ***SC-code*** includes the ***SC-code Core*** and is considered as an ***Extension of the SC-Code Core*** [5].

It should be emphasized that unification and the maximum possible simplification of syntax and denotational semantics in the internal language of intelligent computer systems are primarily necessary because the overwhelming amount of knowledge stored in the knowledge base of an intelligent computer system are meta-knowledge describing the properties of other knowledge. Meta-knowledge, in particular, should include various kinds of logical propositions and various kinds of programs, descriptions of methods (skills) that provide solutions to various classes of problems. It is necessary to exclude the dependence of the form of the represented knowledge on the type of this knowledge. The form (structure) for the internal representation of knowledge of any kind should depend only on (!) from the meaning of this knowledge [5].

Moreover, constructive (formal) development of the theory of intelligent computer systems is impossible without clarification (unification, standardization) and ensuring semantic compatibility of various knowledge types stored in the knowledge base of an intelligent computer system. It is obvious that the variety of forms for representing semantically equivalent knowledge makes the development of a general theory of intelligent computer systems practically impossible [5].

The ***Alphabet of the SC-code Core*** [5], as well as the alphabet of the required knowledge representation language, contains:

- ***sc-elements of an uncertain type*** (vertexes, ***sc-elements***) $(= U_{E_{SC}})$ [1];
- ***sc-arcs*** (arcs of a unified type) $(= E_{E_{SC}})$;
- ***basic sc-arcs*** (basic arcs) $(= B_{E_{SC}})$.

During processing the text of the ***SC-code Core***, from text to text, the syntactic type of ***sc-elements*** can be specified – an ***sc-element of an uncertain type*** can be an ***sc-arc***, an ***sc-arc – basic sc-arc***.

The ***Alphabet of the SC-code Core*** corresponds to the features of the classification of sc-elements and sets the syntactic classification of sc-elements [5].

Syntactic classification of ***sc-elements*** of the ***SC-code Core*** [5]:

- ***sc-elements of an uncertain type*** $(= U_{E_{SC}})$;
  - ***sc-nodes*** $(= V_{E_{SC}})$;
  - ***sc-arcs*** $(= E_{E_{SC}})$;
    - ***sc-arcs of a common type*** $(= C_{E_{SC}})$;
    - ***basic sc-arc*** $(= B_{E_{SC}})$.

All classes of sc-elements included in the syntactic classification of sc-elements are syntactically highlighted classes of sc-elements [5]. The ***SC-code*** is referred to as the syntactic extension of the ***SC-code Core***, since the ***Alphabet of the SC-code*** is an extension of the ***Alphabet of the SC-Code Core***.

The syntactic extension of the SC-code Core consists in the introduction of an additional class of syntactically equivalent elementary fragments of constructions of the***SC-code Core*** – sc-elements designating internal files stored in the ostis-system memory [5].

All files representing electronic images of information constructions external to the ***SC-code*** can be represented in the ***SC-code*** using graph structures in which sc-elements designate letters of texts or pixels of images [5].

The most important type of internal files of ostis-systems are files of external identifiers of sc-elements (in particular, names of sc elements) representing sc-elements in texts of external languages (including in texts of SCs- and SCn-codes) [8], [10], [5]. The ***Set of all the elements of the SC-code Core constructions*** and the ***Set of all the elements of the SC-code constructions*** completely coincide, since for each element of the ***SC-code Core*** construction there is an element synonymous with the ***SC-code*** construction and vice versa. It follows from this that the semantic classification of the elements of ***SC-code*** and ***SC-code Core*** information constructions are also completely identical. Everything that can be designated and described by texts of the ***SC-code*** can be designated and described by texts of the ***SC-code Core***. The difference between the ***SC-code*** and the ***SC-code Core*** is that a new syntactically highlighted class of sc-elements is added to the ***SC-code*** – a class of sc-elements that are signs of specific (constant) files stored in the ostis-system memory. Such "internal" files are necessary so that information constructions that are not texts of the ***SC-code*** can be stored and processed in the ostis-system memory, which is necessary when entering (perceiving) information coming from outside, as well as when generating information structures transmitted to other subjects. The inclusion in the ***SC-code*** of special syntactically highlighted sc-nodes denoting electronic images (files) of various types of information constructions that are not SC-code constructions makes it possible to process not only in the ostis-system memory, that is, in the same storage environment, not only ***SC-code*** constructions but also constructions "external" for it. Without the implementation of the ostis-system interface, it is impossible to implement syntactic analysis, semantic analysis, and understanding, as well as it is also impossible to realize the synthesis (generation) of external information constructions belonging to a given external language and semantically equivalent to a given meaning. Since all the syntactic and semantic properties of the ***SC-code*** and the ***SC-code Core*** are very close, when describing the ***SC-code***, attention is focused on its differences from the ***SC-code Core***, as well as for a more detailed consideration of the semantic classification of elements [5].

The ***Syntax of the SC-code*** differs from the ***Syntax of the SC-Code Core*** by the fact that in the ***Alphabet of the SC-code***, the class of ***sc-elements*** is additionally introduced, which are signs of files stored in the ostis-system memory [5].

The ***Alphabet of the ostis-systems language***, the Alphabet of sc-elements within the SC-code, the alphabet of the extension

[1]Similarly (by the equality of the name in parentheses) we will denote synonyms of key elements of knowledge representation languages, in order to reduce the length of formulas in which they are used as synonymous names in a local context

of the required knowledge representation language contains:

- ***sc-element of an uncertain type*** (vertexes, ***sc-elements of the common type***) ($= U_{E_{SC}}$);
- ***sc-elements with contents*** (***sc-files***) ($= D_{E_{SC}}$);
- ***sc-nodes*** (nodes) ($= V_{E_{SC}}$);
- ***sc-arcs of permanent membership*** (arcs of the permanent membership) ($= P_{E_{SC}}$);
- ***sc-arcs of temporal actual membership*** (basic arcs, ***basic sc-arcs***) ($= A_{E_{SC}}$ ($B_{E_{SC}}$));
- ***sc-arcs of temporal phantom membership*** (arcs of temporal phantom membership) ($= T_{E_{SC}}$);
- ***sc-arcs of fuzzy membership*** (arcs of fuzzy membershipи, arcs of a unified type ($= F_{E_{SC}}$ ($= E_{E_{SC}}$));
- ***sc-arcs of temporal phantom non-membership*** (arcs of temporal phantom non-membership) ($= H_{E_{SC}}$);
- ***sc-arcs of temporal actual non-membership*** (arcs of temporal actual non-membership) ($= G_{E_{SC}}$);
- ***sc-arcs of permanent non-membership*** (arcs of permanent non-membership) ($= N_{E_{SC}}$).

The ***Alphabet of the ostis-systems language*** is [5]:

- a family of maximum sets of syntactically equivalent (within the ***SC-code***) sc-elements;
- a family of classes with syntactically equivalent sc-elements of the ***SC-code***;
- a family of sets, each of which includes all syntactically equivalent to each other (within the ***SC-code***) sc-elements and only them.

The ***Alphabet of the ostis-systems language*** sets the signs (parameters) of syntactic equivalence of sc-elements [5].

The set of all elements of the ***SC-code*** constructions coincides with the set of all elements of the ***SC-code Core*** constructions. Just in the ***SC-code*** constuctions some sc-elements having a "syntactic label" (syntactic type) of an ***sc-element of a common type***, will have the "label" of the sc-element, which is the sign of an internal file stored in the ostis-system memory [5].

Syntactic classification of sc-elements of the SC-code [5]:

- ***sc-element of an uncertain type*** (vertexes, ***sc-elements***) ($= U_{E_{SC}}$);
  - ***sc-nodes*** (nodes) ($= V_{E_{SC}}$);
  - ***sc-arcs*** (***sc-arcs of fuzzy membership***) ($= F_{E_{SC}}$ ($E_{E_{SC}}$));
    - ***sc-arcs of a common type*** ($= C_{E_{SC}}$);
      - ***sc-arcs of permanent membership*** ($= P_{E_{SC}}$);
      - ***sc-arcs of temporal phantom membership*** ($= T_{E_{SC}}$);
      - ***sc-arcs of temporal phantom non-membership*** ($= H_{E_{SC}}$);
      - ***sc-arcs of temporal actual non-membership*** ($= G_{E_{SC}}$);
      - ***sc-arcs of permanent non-membership*** ($= N_{E_{SC}}$).
    - ***sc-arcs of temporal actual membership*** ($= B_{E_{SC}}$).

The sets of ***sc-elements of an uncertain type***, ***sc-nodes***, ***sc-arcs of a unified type***, and ***sc-files*** are subsets of the set of ***sc-elements***.

The set of ***sc-arcs*** is equal to the union of the sets of ***sc-arcs of permanent membership***, ***sc-arcs of temporal actual membership***, ***sc-arcs of temporal phantom membership***, ***sc-arcs of fuzzy membership***, ***sc-arcs of temporal phantom non-membership***, ***sc-arcs of temporal actual non-membership***, ***sc-arcs of permanent non-membership***.

The sets of ***sc-arcs*** and ***sc-nodes*** do not intersect, that is, they do not have common elements.

The set of ***sc-arcs of permanent membership*** does not intersect neither with the set of ***sc-arcs of temporal actual membership***, nor with the set of ***sc-arcs of temporal phantom membership***, nor with the set of ***sc-arcs of temporal actual non-membership***, not with the set of ***sc-arcs of temporal phantom non-membership***.

The set of ***sc-arcs of permanent non-membership*** does not intersect neither with the set of ***sc-arcs of temporal actual membership***, nor with the set of ***sc-arcs of temporal phantom membership***, nor with the set of ***sc-arcs of temporal actual non-membership***, not with the set of ***sc-arcs of temporal phantom non-membership***.

The set of ***sc-arcs of permanent membership*** does not intersect with the set of ***sc-arcs of permanent non-membership***.

Such types of elements and their correlations are caused by the need to solve the problems of horizontal profile integration by unifying the representation [11], [8], [12], [3].

In order to ensure vertical integration, some elements of the alphabet can be represented by non-atomic information constructions, which can be interpreted as "contents" of these elements [10], [3], [8], [5], for example, based on generalized formal languages. Such elements can be distinguished into a separate type of alphabet elements.

When mapping to generalized formal languages for the purpose of vertical integration (texts), a set of sc-elements (E_{SC}) corresponds to a subset of a generalized formal language with a given alphabet (A_{SC}), just as the set of all pairs of sc-elements corresponds to a subset of a generalized formal language with a given alphabet (A_{SC}).

This Syntactic classification of sc-elements from the ***Syntactic classification of sc-elements of the SC-code Core*** is distinguished by an additional clarification of the syntactic typology of sc-elements.

A. Syntax of the ostis-system internal language

The ***Syntax of the SC-code Core*** corresponds to the syntax of the languages of the unified semantic knowledge representation model (sc-languages) and is set by the ***Alphabet of the SC-code Core*** and two mentioned incident relations of the alphabet elements of these languages by the ***Incidence relation of sc-connectors**** ($= I_{SC}$) and the ***Incidence relation of incoming sc-arcs**** ($= C_{SC}$).

The ***Incidence relation of sc-connectors**** is a binary oriented relation, the first component of each oriented pair of which is some ***sc-connector*** and the second component is one of the ***sc-elements*** connected by the specified ***sc-connector*** with some other ***sc-element***, which is specified in another incidence pair for the same ***sc-connector*** [5], [8].

The set of ***sc-connectors*** is a subset of ***sc-elements*** [5], [8].

The set of ***sc-arcs*** is a subset of ***sc-connectors*** [5].

The ***Incidence relation of incoming sc-arcs**** is a binary oriented relation, the first component of each oriented pair of which is some ***sc-arc*** and the second component is an ***sc-element***, in which the specified ***sc-arc*** is included, i.e. the ***sc-element***, which is the second component connected (linked) by the specified ***sc-arc*** [5].

The ***Incidence relation of incoming sc-arcs**** is a subset of the ***Incidence relation of sc-connectors**** [5].

On the basis of these relations, one incidence relation can be distinguished ***Incidence relation**** $(= R)$. The ***Incidence relation**** is a union of the ***Incidence relation of sc-connectors**** and the backward relation to its symmetric difference with the ***Incidence relation of incoming sc-arcs****.

For each ***sc-connector*** (E), there are two and no more than two pairs of the ***Incident relation of sc-connectors****, the specified ***sc-connector*** is the first binding component. At the same time, for each ***sc-arc*** (A), one of the specified incident pairs must belong to the ***Incidence relation of the incoming sc-arc****.

The ***sc-connectors*** connecting the ***sc-element*** to itself will be called ***loop sc-connectors*** (***loop sc-edges*** and ***loop sc-arcs***). The incidence pairs of ***loop sc-connectors*** are as if they were multiples.

To the ***Incidence relations of sc-connectors**** and ***Incidence relations of incoming sc-arcs**** definition domain, not only ***sc-nodes*** are included but also***sc-connectors***. This means that an ***sc-connector*** can connect (link) not only an ***sc-node*** with an ***sc-node*** but also an ***sc-node*** with an ***sc-connector*** and even an ***sc-connector*** with an ***sc-connector***.

In the sc-text, for each occurrence of an ***sc-element***, its syntactic class can be set (***syntactic class of the occurrence of the sc-element in the sc-text****), according to which syntactic subclasses of occurrences of sc-elements can be distinguished for this sc-text: ***syntactic class of occurrences of terminal sc-elements in the sc-text****(T), ***syntactic class of occurrences of node sc-elements in the sc-text**** (V), ***syntactic class of occurrences of edge sc-elements in the sc-text**** (E), ***syntactic class of occurrences of arc sc-elements in the sc-text**** (A), ***syntactic class of occurrences of basic sc-elements in the sc-text**** (B). For each sc-text, it is possible to determine the set of all its components (***components of sc-text****) and the relation of occurrences of the components of the ***Incidence relation of sc-connectors**** and ***Incidence relations of sc-arcs**** connectives: ***Relation of occurrences of incident sc-connectors of the sc-text**** (I), ***Relation of occurrences of incident sc-arcs of the sc-text**** (C).

In this case, the following features will be performed:

- the sum of powers for unions of sets of ***Relation of occurrences of incident sc-connectors of sc-text**** and ***Relation of occurrences of incident sc-arcs of sc-text**** and sets of ***syntactic class of occurrences of terminal sc-elements in sc-text****, ***syntactic class of occurrences of node sc-elements in sc-text**** and ***syntactic class of occurrences of edge sc-elements in sc-text**** to the number of components of this sc-text (***components of sc-text****);
- the ***syntactic class of occurrences of arc sc-elements in sc-text**** is a subset of ***syntactic class of occurrences of edge sc-elements in sc-text****;
- The ***Relation of occurrences of incident sc-connectors of sc-text**** is a set of pairs connecting the occurrence of a ***component of sc-text**** with the occurrence of a component of the same sc-text, a pair of these components is an element of the ***Incidence relation of sc-connectors****;
- The ***Relation of occurrences of incident sc-arcs of sc-text**** is a set of pairs connecting the occurrence of a ***component of sc-text**** with the occurrence of a component of the same sc-text, a pair of these components is an element of the ***Incidence relation of sc-arcs****;
- The ***Relation of occurrences of incident sc-arcs of sc-text**** is a subset of the ***Relation of occurrences of incident sc-connectors of sc-text****;
- the ***syntactic class of occurrences of basic sc-elements in sc-text**** is a subset of the ***syntactic class of occurrences of arc sc-elements in sc-text****;
- a ***syntactic class of occurrences of arc sc-elements in sc-text**** is a subset of the ***syntactic class of occurrences of edge sc-elements in sc-text****;
- the ***syntactic class of occurrences of arc sc-elements in sc-text**** does not intersect with the ***syntactic class of occurrences of node sc-elements in sc-text****;
- the ***syntactic class of occurrences of arc sc-elements in sc-text**** is a set of pairs connecting sc-text with its component (***components of sc-text****), which is an ***sc-arc***;
- the ***syntactic class of occurrences of node sc-elements in sc-text**** is a set of pairs connecting sc-text with its component (***components of sc-text****), which is an ***sc-node***;
- the ***syntactic class of occurrences of basic sc-elements in sc-text**** is a set of pairs connecting sc-text with its component (***components of sc-text****), which is an ***sc-arc of actual membership***;
- the direct product of the set (***Cartesian product of the set****) of the union of the ***syntactic class of occurrences of terminal sc-elements in the sc-text**** with the difference of the ***syntactic class of occurrences of node sc-elements in the sc-text**** with the ***syntactic class of occurrences of edge sc-elements in the sc-text**** of itself does not intersect with the ***Relation of occurrences of incident sc-connectors of sc-text****;
- for each pair of the ***syntactic class of occurrences of edge sc-elements in the sc-text****, there is at least one pair of the ***Relation of occurrences of incident sc-connectors of sc-text****, the first component of which it is;
- for each pair of the ***syntactic class of occurrences of edge sc-elements in the sc-text****, there are no more than two pairs of the ***Relation of occurrences of incident sc-connectors of sc-text****, the first component of which it is;
- for each pair of the ***syntactic class of occurrences of arc sc-elements in the sc-text****, there is a single pair of the ***Relation of occurrences of incident sc-connectors of sc-***

text*, the first component of which it is;
- for each pair of the ***syntactic class of a of occurrences of basic sc-elements in the sc-text****, there is at least one pair of the ***Relation of occurrences of incident sc-connectors of sc-text****, not belonging to the ***Relation of occurrences of incident sc-arcs of sc-text****, but only if there is another pair of the ***Relation of occurrences of incident sc-connectors of sc-text****, the first component of which it is and the second component of which belongs to the ***syntactic class of occurrence of node sc-elements in the sc-text****.

The ***Syntax of the SC-code Core*** [5] is set:

- by the ***Alphabet of the SC-code Core***;
- by the ***Incidence relation of sc-connectors**** and the ***Incidence relation of incoming sc-arcs****;
- by the rules of connection (incidence) of ***sc-elements*** (for example, which types of sc-elements cannot be incident to each other) when they occur in sc-texts;
- by structural (syntactic) constraints in the semantic neighborhood of the key elements of the ***SC-code Core***.

The syntax of the internal language of ostis-systems (the ***Syntax of the SC-code***) is set by the syntax of the languages for the unified semantic representation of knowledge model (sc-languages) and with the exception of the ***Alphabet of the SC-Code Core*** and the ***Alphabet of the SC-code*** is exactly the same as the ***Syntax of the SC-Code Core***.

The ***Syntax of the SC-code*** [5] is set:

- by the ***Alphabet of the SC-code***, that is, the typology (alphabet) of sc-elements (atomic fragments of SC-code texts);
- by the ***Incidence relation of sc-connectors**** and the ***Incidence relation of incoming sc-arcs****;
- by the rules of connection (incidence) of ***sc-elements*** (for example, sc-elements of which types cannot be incident to each other) when they occur in sc-texts;
- by structural (syntactic) constraints in the semantic neighborhood of the key elements of the ***SC-code***.

B. Basic denotation semantic of the ostis-system internal language

Denotational semantics is a description of the correspondence of information constructions belonging to the language (the ***SC-code Core***) and entities described by these constructions [5].

Denotational semantics of the SC-code Core [5]. According to the unified semantic knowledge representation model, the semantics of the ***SC-code Core*** (including the basic one) is expressed by:

- proper and non-proper sublanguage key elements for (pairs of) languages (language and sublanguage);
- semantic neighborhoods of key elements in sublanguage texts for language pairs;
- semantic interpretations of text elements.

Due to the fact that the semantics and meaning of designations in the texts of languages of the unified semantic representation of knowledge model are expressed through the connections of elements, the family of all texts of a language defines a set of key elements in relation to any of its superlanguages and vice versa. Accordingly, the semantics (interpretations) of text elements and semantic neighborhoods of key elements of the language are designated [8].

Thus, semantics can be defined by enumeration of the key elements of a language (semantic types of elements) or enumeration, formation of its texts. It is important to note that the becoming of language texts in the process of their integration is a natural mechanism that sets not only the denotational semantics of the language, but also the operational and game semantics [9], [23], [24], allowing to consider their joint and unified formalizations within one semantic (meaningful) spaces.

Due to the above and the fact that the integration and processing of knowledge is unthinkable without movement and change, which is a special case of becoming [12], [3], the concept of becoming and the key element of ***becoming**** is a necessary key element of the ***SC-code Core*** and the internal language of ostis-systems (***SC-code***).

The basic mathematical concept that allows expressing the basic denotational semantics of the language in question is the concept of a distensible set (***sc-set***) [8], [3].

Projectively, the concept of an distensible set ($Distensible$) can be expressed in accordance with the scheme [8]:

$$\langle Universe, Events, Becoming, Designator, Distensible, \square, \mathbf{Z}_{+}, [0;1]\rangle$$

through the mathematical concept of a set as follows:

$$Distensible \tilde{\subseteq} \Xi,$$

where

$$\Xi = 2^{Events} \times \Phi^{Universe},$$

$Universe$ – a set of elements (including designations of distensible sets), $Events$ – a set of (elementary) events, becoming relation $Becoming$:

$$Becoming \subseteq Events \times Events,$$

designation event function $Designator$:

$$Designator \in \left(2^{Events}\right)^{Universe},$$

and

$$\Phi = \bigcup_{s}^{s\in\square} (\Psi)^{\{s\}}; \Psi = \nabla \times \left(\Delta^{Events}\right);$$

$$\nabla = \bigcup_{p}^{p\in\mathbf{Z}_{+}} D^{\left(D^{p-1}\right)}; \Delta = \bigcup_{q}^{q\in_{+}} D^{\left(D^{q-1}\right)}; D = [0;1],$$

where $\square$ – a linearly ordered set ($\mathbf{Z}_{+} \subseteq \square$).

An algebraic structure over distensible sets with operations is permissible:

$$\left\{\bullet_{\Xi}^{k}\right\} \subseteq \Xi^{\Xi\times\Xi},$$

when $k \in \{\cup, \cap, \otimes, \oplus, ...\}$, which can be expressed:

$$\langle \alpha, \delta\rangle \bullet_{\Xi}^{k} \langle \beta, \gamma\rangle = \left\langle \alpha \cup \beta, \tau\left(\left\langle \delta, \gamma, \bullet_{\Phi}^{k}, Universe\right\rangle\right)\right\rangle,$$

where

$$\tau\left(\langle \alpha, \beta, \varphi, \sigma\rangle\right) = \left\{\langle \chi, \varphi\left(\langle \alpha\left(\chi\right), \beta\left(\chi\right)\rangle\right)\rangle \,|\chi \in \sigma\right\},$$

and operations

$$\left\{\bullet_{\Phi}^{k}\right\}\subseteq\Phi^{\Phi\times\Phi}$$

in turn

$$\alpha\bullet_{\Phi}^{k}\beta=\tau\left(\left\langle\alpha,\beta,\bullet_{\Psi}^{k},\square\right\rangle\right).$$

The last ones are expresses through operations

$$\left\{\bullet_{\Delta}^{k}\right\}\subseteq\Delta^{\Delta\times\Delta}$$

and mappings:

$$\left\{\bullet_{\nabla}^{k}\right\}\subseteq\left(\nabla^{\left(\Delta^{Events}\right)}\right)^{\nabla\times\nabla}$$

as follows:

$$\langle\alpha,\delta\rangle\bullet_{\Psi}^{k}\langle\beta,\gamma\rangle=\kappa\left(\left\langle\alpha,\beta,\bullet_{\nabla}^{k},\tau\left(\left\langle\delta,\gamma,\bullet_{\Delta}^{k},Events\right\rangle\right)\right\rangle\right),$$

where

$$\kappa\left(\langle\alpha,\beta,\varphi,\varepsilon\rangle\right)=\left\langle\varphi\left(\langle\alpha,\beta\left(\varepsilon\right)\rangle\right),\varepsilon\right\rangle.$$

In turn, the remaining operations and mappings are expressed by:

$$\left\{\bullet_{D}^{k}\right\}\subseteq D^{D\times D},$$

$$\zeta\in\left(\bigcup_{p}^{p\in\mathbf{Z}_{+}}D^{\left(D^{p-1}\right)}\right)^{\left(\mathbf{Z}_{+}\times D\right)},$$

$$\nu\in\mathbf{Z}_{+}^{\left(\bigcup_{p}^{p\in\mathbf{Z}_{+}}D^{\left(D^{p-1}\right)}\right)},$$

$$\mu\in D^{\left(\bigcup_{p}^{p\in\mathbf{Z}_{+}}D^{\left(D^{p-1}\right)}\right)}$$

as follows:

$$\alpha\bullet_{\nabla}^{k}\beta=\zeta\left(\left\langle\nu\left(\alpha\right)\cup_{\mathbf{Z}_{+}}\nu\left(\beta\right),\mu\left(\alpha\right)\bullet_{D}^{k}\mu\left(\beta\right)\right\rangle\right),$$

where $\cup_{\mathbf{Z}_{+}}=\max$, and, for example, $\bullet_{D}^{\cup}=\max$, $\bullet_{D}^{\cap}=\min$, $\bullet_{D}^{\otimes}=*$, $\alpha\bullet_{D}^{\oplus}\beta=\left(\alpha\cap^{+}\left(1-\beta\right)\right)\cup^{+}\left(\left(1-\alpha\right)\cap^{+}\beta\right)$.

Moreover:

$$\forall\varphi\forall p\left(\left(\varphi\in D^{\left(D^{p-1}\right)}\right)\rightarrow\left(\nu\left(\varphi\right)=p-1\right)\right),$$

$$\forall\sigma\forall p\forall\varepsilon\left(\left(\sigma\in D^{p-1}\right)\rightarrow\left(\zeta\left(\langle p,\varepsilon\rangle\right)\left(\sigma\right)=\pi\left(\sigma+\langle\varepsilon\rangle\right)\right)\right),$$

$$\forall\sigma\left(\left(\sigma\in\{1\}^{\nu(\chi)-1}\right)\rightarrow\left(\mu\left(\chi\right)=\chi\left(\sigma\right)\right)\right),$$

where:

$$\chi\in D^{\left(D^{\nu(\chi)-1}\right)},$$

$$\pi\left(\langle\rangle\right)=1,$$
$$\pi\left(s+\langle e\rangle\right)=e\cdot\pi\left(s\right)+\left(1-e\right)\cdot\left(1-\pi\left(s\right)\right)$$

or in a non-recurrent form:

$$\pi\left(s\right)=\frac{1+(-1)^{\dim(s)}\cdot\sum_{i}^{i\in\mathbf{N}\cup\{0\}}\frac{(-2)^{i}}{i!}\cdot\sum_{m}^{m\in\left\{s_{j}|j\right\}^{i}}\prod_{j}^{m_{j}}m_{j}}{2}.$$

Structurally, the concept of a distensible set can be expressed within the formal reflexive and descriptive semantics of the languages for the unified semantic representation of knowledge model, with the involvement of more fundamental concepts of the becoming of the actual and phantom (event) [12], [3], [37].

Key elements of the ***SC-Code Core*** [8]:

- an ***sc-sign*** is the designation of the sc-set and the element (component) of any sc-set designated by the sc-element [3], [17];
- ***sc-set***;
  - the sc-set is a distensible set, any sc-element is a designation (sign) of the sc-set;
- ***node sc-set***;
  - a node sc-set is a distensible set that is not an sc-pair of fuzzy membership, any sc-node is a designation (sign) of a node sc-set;
- ***sc-pair of fuzzy membership***;
  - an sc-pair of fuzzy membership is an sc-pair, the designation of which belongs to the relation of fuzzy membership, sc-arc of fuzzy membership is the designation (sign) of the sc-pair, the first component of which is the designation (sc-sign) of the sc-set, denoted by the sc-element from which this sc-arc goes out, and the second component which is the designation (sc-sign) of the sc-set denoted by the sc-element in which this sc-arc comes [12];
- ***sc-pair of permanent membership***;
  - an sc-pair of permanent membership is an sc-pair, the designation of which belongs to the relation of permanent membership, an sc-arc of permanent membership is the designation (sign) of an sc-pair, the first component of which is the designation (sc-sign) of the sc-set S, denoted by the sc-element from which this sc-arc goes out, and the second component of which is permanently (as long as it exists) belonging to the sc-set S designation (sc-sign) of the sc-set designated by the sc-element in which this sc-arc comes [12];
- ***sc-pair of temporal actual membership***;
  - an sc-pair of temporal actual membership is an sc-pair whose designation belongs to the relation of temporal actual membership, an sc-arc of temporal actual membership is the designation (sign) of the sc-pair, the first component of which is the designation (sc-sign) of the sc-set S, denoted by the sc-element from which this sc-arc goes out, and the second component of which is temporarily currently belonging to the sc-set S designation (sc-sign) of the sc-set designated by the sc-element in which this sc-arc comes [12];
- ***sc-pair of temporal phantom membership***;
  - an sc-pair of temporal phantom membership is an sc-pair whose designation belongs to the relation of temporal phantom membership, an sc-arc of temporal phantom membership is the designation (sign) of the sc-pair, the first component of which is the designation (sc-sign) of the sc-set S, denoted by the sc-element from which this sc-arc goes out, and the second component of which is temporarily belonging to the sc-set S designation (sc-sign) of the sc-set designated by the sc-element in which this sc-arc comes [12];

- ***sc-pair of temporal phantom non-membership***;
  - an sc-pair of temporal phantom non-membership is an sc-pair whose designation belongs to the relation of temporal phantom non-membership, an sc-arc of temporal phantom non-membership is the designation (sign) of the sc-pair, the first component of which is the designation (sc-sign) of the sc-set S, denoted by the sc-element from which this sc-arc goes out, and the second component of which is temporarily not belonging to the sc-set S designation (sc-sign) of the sc-set designated by the sc-element in which this sc-arc comes [12];
- ***sc-pair of temporal actual non-membership***;
  - an sc-pair of temporal actual non-membership is an sc-pair whose designation belongs to the relation of temporal actual non-membership, an sc-arc of temporal actual non-membership is the designation (sign) of the sc-pair, the first component of which is the designation (sc-sign) of the sc-set S, denoted by the sc-element from which this sc-arc goes out, and the second component of which is temporarily currently not belonging to the sc-set S designation (sc-sign) of the sc-set designated by the sc-element in which this sc-arc comes [12];
- ***sc-pair of permanent non-membership***;
  - an sc-pair of permanent non-membership is an sc-pair whose designation belongs to the relation of permanent non-membership, an sc-arc of permanent non-membership is the designation (sign) of the sc-pair, the first component of which is the designation (sc-sign) of the sc-set S, denoted by the sc-element from which this sc-arc goes out, and the second component of which is permanently (as long as it exists) not belonging to the sc-set S designation (sc-sign) of the sc-set designated by the sc-element in which this sc-arc comes [12];
- ***node sc-pair***;
  - node sc-pair is an sc-pair designated by the sc-node;
- ***sc-pair***;
  - sc-pair is an sc-set, to which there are only two memberships of different sc-elements or of the same sc-element;
- ***sc-connective***;
  - sc-connective is an sc-set whose sc-subset is an sc-pair;
- ***sc-relation***;
  - sc-relation is an sc-set of sc-connectives;
- ***binary sc-relation***;
  - binary sc-relation is an sc-set of sc-pairs;
- ***slot sc-relation***;
  - slot sc-relation is an sc-set of sc-pairs that are not node sc-pairs;
- ***attributive sc-relation***;
  - attributive sc-relation is an sc-set of sc-pairs of membership (permanent, temporal, actual, or phantom);
- ***sc-file***;
  - sc-file is an entity designated by an sc-element whose contents is an sc-file (a finite dynamic or static data structure);
- ***sc-structure****;
  - sc-structure* is an sc-set in which there is an sc-subset-carrier (the set of primary elements of the sc-structure);
- ***perception ****;
  - perception* is a binary sc-relation between an sc-element and an sc-set of its images;
- ***explanation****;
  - explanation* is a binary sc-relation between an sc-element and an sc-set of its explanations;
- ***becoming****;
  - becoming* is a binary sc-relation between events (states) or phenomena.

Each ***sc-element*** is a sign (designation) of some described entity [5].

Any entity can be designated by an ***sc-element*** and, accordingly, described as a construction of the ***SC-code Core*** [5].

With the help of the sc-elements, it is possible to represent any connections between sc-elements and/or between entities that are designated by these sc-elements. In this case, these connections are considered as extensible sets of connected sc-elements and are designated by sc-arcs, and in the case of non-binary connections – by sc-nodes [5].

Since each sc-connector is semantically interpreted as the designation of a pair of sc-elements connected (linked) by this sc-connector, each pair of incidence of the sc-connector is semantically interpreted as an membership pair connecting the sc-connector with one of the elements of the pair of sc-elements designated by it, and the sc-connector itself is its designation [5]. Any described entity can be designated by an sc-element of an uncertain type, however, the reverse is not true, since some entities can only be designated by sc-arcs of a general type, basic sc-arcs [5].

Basic denotational semantics of the internal language of ostis-systems (SC-code). The basic denotational semantics of the ***SC-code*** (the internal language of ostis-systems) basically corresponds to the basic denotational semantics of the ***SC-Code Core***, due to the preservation of all the key elements of the ***SC-Code Core*** in the SC-code. However, the semantics of the sc-texts of the ***SC-code*** differs from the semantics of the sc-texts of the ***SC-code Core***, since a more precise semantic interpretation can be set by the membership of the sc-text component to the corresponding syntactic class that does not belong to this sc-text, in contrast to its assignment by belonging to the key element, belonging to sc-text of which is required [5], [8].

The semantic proximity of the ***SC-code*** and the ***SC-code Core*** is a consequence of the fact that the ***SC-code*** is a syntactic extension of the ***SC-code Core*** [5].

V. Semantic Space

A. Review of approaches

The word "space" takes its etymological roots in Proto-Indo-European word with the meaning "to stretch" or "to pull". The word "semantics" originates from Anc. Greek "semantikos" that is created by union of "semaino" (to indicate, to sign) and suffix "ikos". On the other hand, "meaning" rooted in "semaino" originates from the Proto-Indo-European word with the meaning "to think" or "to change". So, "semantic space" means "stretched thought" and is interpreted as a phenomenon of thinking (i.e. movement of thoughts). The need for considering such a concept is related to the need for analysis of structural and quantitative (metrical) features aiming to detect limits, assess completeness of thinking processes, and to optimize costs needed per their modeling having finite resources.

It probably might seem, that the concept of "semantic space", or something close or similar to this, originates from ancient times by philosophers, including idealists and dualists, e.g. Platonic "world of ideas" [38]. However, philosophic concepts change along their development and are often not well defined so that one can make unambiguous statements and be definitely certain about them, particularly being in the conditions of incomplete information about historical "facts" and inaccuracy of historical evidence. In addition, the concept of "space" often related to the concept of "time", for it most likely is a "container" ("substance") for something changeable, impermanent, i.e. – materialistic, things (from "thinking") rather than constant, "perpetual" "ideas". Perhaps, R. Descartes was one of the first in European history who showed signs of intentions, which came down to us, to connect "thinking" and "material" ("things") "space" via the God and relationism [39] that was stood by G. W. Leibniz afterwards [40], and together with the similarity of properties of "thinking" to natural "extension" is reflected by D. Hilbert [41]. Reasoning about "space" and "thinking", D. Hilbert had been rejecting their infinity. Also, it could be that Hegel's concepts [37] are closer to the concept of "semantic space". In dialectical materialism in accordance with the definition for "matter" the existence of meaning should be accepted only materialistically in the space and time.

Largely due to the development and definition of mathematical concepts for various mathematical structures, in modern science, "space" is understood not only as "material", "physical space".

Let us take a look at the history of the usage of the term "space" in mathematics. It is believed that infinite constructions are not considered in the beginnings of Euclid, which is also considered characteristic of ancient mathematicians, therefore there are no sufficient grounds to speak of an "ancient concept" of"Euclidean space". Modern science II defines a lot of mathematical structures (Fig. 10) up to hypothetical ones ("antispace" [47]) in the name of which the term "space" is used:

- vector (linear) space (/finite-dimensional) [48];
- affine space (/finite-dimensional);
- topological space [49];
- linear topological space;
- pseudometric space [50];
- metric space [51];
- locally convex space;
- semi-normed space;
- normed space [54];
- Banach space;
- pre-Hilbert space;
- Hilbert space;
- function space;
- Euclidean space;
- pseudo-Euclidean space;
- space with measure;
- probability space.

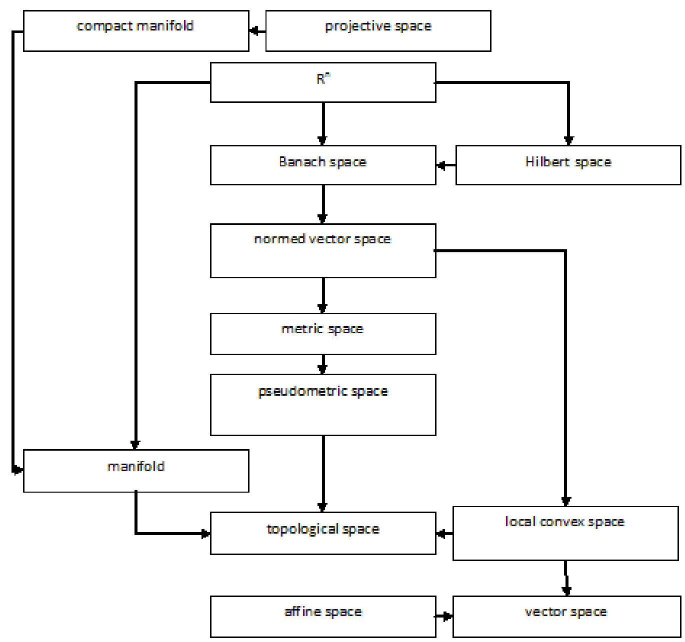

Fig. 10. A conceptual scheme of "spaces".

In other sciences, it is possible to find "phase space" [52], "object space" [53], etc. Among the representatives of more exact sciences, the works of D. Bohm and V. V. Nalimov can be highlighted.

The works of D. Bohm [43], [42] raise questions of interpretation of physical phenomena at the level of the microworld within the subject of quantum mechanics, questions related to the mutual influence of an observer (subject) on an object in an experiment, questions related to integrity and partiality of perception, the physical nature of consciousness and the duality of the properties of the objects of the microcosm, revealed in the experiments II. For D. Bohm, "holonomic movement" and "implicate order" are conceived as a principle and substance capable of linking "part" and "whole", "consciousness" associated with "field of knowledge" (thougths), considered as a process, and "matter" in space, which has higher dimensions. The most complete work in this regard is his work "Wholeness and the Implicate Order" [43].

In the works of V. V. Nalimov [44], [45], the "probabilistic space of meanings" is considered within the "probabilistic theory of meanings". According to the position of one of the founders of the theory of probability, A. N. Kolmogorov, one of its fundamental questions is the question of the connection and mutual influence of the subject on the object [71]. Despite the fact that the concept of a probability space could hardly be considered as a model for the world of ideas in Plato's time, V. V. Nalimov considers himself a Platonist. Such a "probability space of meanings" is assumed to be different from the "physical space" and not included in it II:

"Physical space, according to Bastin's ideas, represents a finite series of points, for which the rules for constructing new points are postulated, creating a hierarchy of points (see [215], p. 99)." [46]

Table II
Comparison of approaches to investigate «semantic spaces»

	exterior (physical) approach	***interior (abstract, logic-semiotic) approach based on analysis of***	
		quantitative features (probabilistic (additive) measures)	***struc-ture-dynamic features***
cognitive process analysis (introspection)	+	-	?
adaptation	+	-	+
unification	-	+	+

In modern works in the technical sciences [14], perhaps the closest concepts are those expressing the meaning of the term "semantic space" (interior approach II). Common in many approaches to working with "semantic space" is the consideration of word forms or lexemes (sets of word forms) and their features (II). The following approaches III are found in the literature [14]:

- an approach based on semic axes and feature space (binary $\{0,1\}^n$, unipolar $[0;1]^n$, bipolar (bisemic) $[-1;1]^n$);
- an approach based on semic axes and neural encoding of a place field recognition of meanings (while words and phrases have areas (subsets) of meanings being connected by other parts of speech as inclusion and intersection, texts correspond to the path of connected areas, and binary coding is used for groups of neurons recognizing meanings);
- an approach based on the "sense-text" [55] model (reflection of the incompleteness of semantic scales and analysis of syntagmas and surface-syntactic structure);
- an approach based on neurolinguistic data that reflects the processes of production and perception of speech in neural networks (lexical production network), is close to the "meaning-text" model;
- models built on the basis of static analysis (of corpora) of texts (vector space model).

The statistical approach to natural language processing is opposed to the intuition and communicative experience of scientists [14].

An approach is based on the semantic statistical hypothesis that the meaning of words (lexemes) is determined by the context of usage (its statistical pattern) in the language (with a communicative structure) [14].

Vector space models of semantics [14]. A concrete model is considered for two cases: a large vocabulary case ($N \leq M$) and an information retrieval case ($M \leq N$), where M is the dictionary size, N is the number of contexts.

On the basis of statistics, a matrix of dimensions $M \times N$ of frequencies p_{ij} of occurrences of a lexeme (word) w_i in document (context, subtexts that may overlap) c_j.

$$x_{ij} = \max\left(\{0\} \cup \left\{\log\left(\frac{p_{ij}}{(\sum_j p_{ij}) * (\sum_i p_{ij})}\right)\right\}\right)$$

The denominator contains word and context probability estimates, respectively.

In the case of a nondegenerate matrix with its rank $r = N$, each such matrix defines a point in the Grassmannian of N-dimensional subspaces of a M-dimensional space ($N \leq M$).

In the case of a nondegenerate matrix with its rank $r = M$, each such matrix defines a point in the Grassmannian of M-dimensional subspaces of a N-dimensional space ($M \leq N$).

Each text is a point in the Grassmannian [56] corresponding to the projective space $P^{M-1} = Gr\left(\langle 1, M\rangle\right)$, relative to one selected context. For all contexts, in accordance with the order of contexts in texts and the resulting oriented N-tuple, a route (path) can be constructed by connecting adjacent points in the N-tuple with geodesics. For two texts T and T', these paths will be two polygonal curves. The Frechet metric between these curves [18] can be calculated using the Fubini-Study metric [19] in P^{M-1}. To calculate it, the paths $\Gamma\left(\mathrm{T}\right)$ and $\Gamma\left(\mathrm{T}'\right)$ should be parameterized through t ($\gamma \in \Gamma(\mathrm{T})^{[0;1]}$, $\gamma' \in \Gamma(\mathrm{T}')^{[0;1]}$):

$$\delta\left(\langle\Gamma\left(\mathrm{T}\right), \Gamma\left(\mathrm{T}'\right)\rangle\right) = \inf_{\gamma,\gamma'} \max_{t\in[0;1]} \left\{d_{FS}\left(\langle\gamma\left(t\right), \gamma'\left(t\right)\rangle\right)\right\}.$$

Another way to specify a linear order is to consider the flag (filtering (flag manifold)) [57] in $\mathbf{R}^M$ defined by expanding (embedding) contexts. As a result, for the text we get points (flags) in the flag manifold. For flag varieties, one can also calculate the Fubini-Study metric [19].

This order corresponds to the temporal dimension (communication process in time), which can be significant. Other ordering may be independent of this, such as alphabetical or Zipf's order [58], [59].

The issues of formalizing meanings, their correlation, the genesis (in space and time) of languages are considered in the works of V.V. Martynov [60], [61], [62].

To solve the identified problems, it may be worth turning to an alternative approach: to explore not only the communicative

Table III
Comparison of "semantic spaces" construction approaches

	semic axes and		*"meaning-text" model*	*neurolingvistic coding*	*statistical model (semantic vector space model)*
	feature spaces	*neural encoding place field recognition*			
defined semic axes	+	+	-	-	-
dynamic (computing) decomposition	-	+	-	+	-
cognitive process analysis (introspection)	-	+	-	+	-
accounting of NON-factors (incompleteness)	-	-	+	+	+

structures of the language but also the cognitive-representational structures of the language [70].

The prerequisites for building knowledge representation models that claim to be universal have been created in the works carried out in accordance with the graph-dynamic paradigm of information representation and processing [17], [5], [8] .

B. SC-space

The concepts of ***SC-space*** and ***SC-code*** are necessary to clarify and formalize the meaning of information structures with the unification of the semantic representation of information. The meaning of an information construction is ultimately determined by (1) the internal connections of all elementary fragments of this construction and (2) its external connections with the elements of the semantic space (its position in the context). The semantic space is the result of the natural formation of knowledge in the process of their integration.

The most important advantage of ***SC-space*** is the possibility of clarifying such concepts as the concept of similarity (similarities and differences) of various described "external" entities, similarity of unified semantic networks (texts of the ***SC-code***), the concept of semantic proximity of the described entities (including texts of the ***SC-code***).

It should be noted that it cannot be ruled out that the union of two arbitrary texts of such languages will not be the text of the language of the unified semantic knowledge representation model due to the abstractness of the languages of the unified semantic knowledge representation model and the conventionality of the choice of labels for the elements of their texts. To avoid the results of such eclectic combinations in terms of syntax or semantics, a set of "semantic spaces" should be considered for abstract languages. However, it may be sufficient to consider one "semantic space" for specific (real) languages.

Next, consider:

- possibility of transition from sc-texts to graph structures and from them to topological space;
- the ability to move from sc-texts to graph structures and from them to a manifold (topological space);
- possibility of transition from sc-texts to graph structures and from them to metric space.

On the set of elements that form ***SC-space***, it is possible to study topological properties and consider ***SC-space*** as a topological space. It should be noted that despite the fact that the ***SC-code*** is focused on the semantic representation of knowledge, due to the presence of non-factors, not all meanings can be represented at some point in time while the structure of the corresponding representation is unknown. Therefore, the structural and topological properties of the texts of the knowledge representation language rather determine the syntactic space than the semantic (meaning) one. Although both can approach each other as the uncertainties caused by non-factors are eliminated.

It is necessary to make several transitions to get the topological space as a transformation of the sc-text, which could result from the integration of many smaller sc-texts:

- transition from texts with syntax of sc-languages to a pseudograph (oriented or unoriented) (Fig. 6, Fig. 7, and Fig. 8);
- transition from an oriented pseudograph to a transitive oriented pseudograph;
- transition from an oriented (transitive) pseudograph to an oriented (transitive) graph;
- transition from an oriented transitive pseudograph to a topological space;
- transition from an oriented pseudograph to an unoriented graph;
- transition from an unoriented graph to a manifold (topological space).

There is a transition from an oriented pseudograph to an oriented bipartite graph (Fig. 11). During this transition, the following occurs: mapping vertices to edges and arcs; mapping arcs to connections (connectives) in accordance with the direction of arc orientation.

Fig. 11. Transformation of an oriented pseudograph to a bipartite orgraph.

There is a transition from an oriented pseudograph to a transitive oriented pseudograph. (Fig. 12).

In this transition, transitive closure of arcs is performed.

There is a transition from a (transitive) oriented pseudograph to a (transitive) oriented graph (Fig. 13).

With this transition, loops are eliminated.

There is a transition from an oriented pseudograph to an unoriented graph (Fig. 14).

Fig. 12. Transformation of an oriented pseudograph to a transitive oriented pseudograph.

Fig. 13. Transformation of a (transitive) oriented pseudograph to a (transitive) orgraph.

In this transition, the following is carried out: matching vertices to vertices; matching pairs of triples of vertices and triples of edges to arcs.

Fig. 14. Transformation of an oriented pseudograph to an unoriented graph.

There is a transition from an unoriented graph to a manifold (topological space) (Fig. 15).

The following is carried out in this transition: matching figures (points) to vertices; matching figures (lines, two-point sets) to edges.

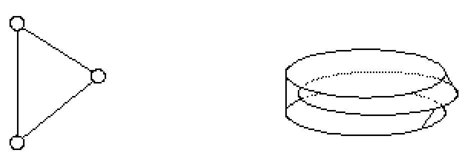

Fig. 15. Transformation of an unoriented pseudograph to a manifold.

The transition from a transitive oriented graph $G = \langle V, E \rangle$ to a topological space T is [63]:

$$G \to G \downarrow \to \overline{T} \to T;$$

$$G \downarrow = \langle V, E \downarrow \rangle ;$$

$$E \downarrow = \{ g \downarrow | g \in E \} ;$$

$$E \downarrow = \left\{ E^{-1} (v) \cup \{v\} | v \in V \right\} .$$

Let us consider the concept of specialization for the reverse transition as a transitive relation (x is the specialization of y):

$$x \in \overline{\{y\}}^{T},$$

that is, x belongs to the closure of $\{y\}$ in T.

There is a reverse transition:

$$E = \left\{ \langle x, y \rangle \middle| x \in \left(\overline{\{y\}}^{T} / \{x\} \right) \right\} .$$

Let us study the question of the possibility of considering ***SC-space*** as a metric space.

The syntactic metric is specified on the lines of the generalized formal language in accordance with the metric tensor over the identified matches, transpositions, exhanges, duplications, fusions, generations, and deletions [13], [3]. The distance ρ_p between the generalized strings α and β is:

$$\rho_p \left(\langle \alpha, \beta \rangle \right) = \rho_p^0 \left(\langle \alpha, \beta \rangle \right) ; (16)$$

$$\rho_p^k \left(\langle \alpha, \beta \rangle \right) = \begin{cases} 1 \left| \langle \alpha, \beta \rangle \in A \times A \right. \\ \psi_k^{\beta} \left(\rho_p^k \left(\langle \alpha, \langle \beta \rangle \rangle \right) \right) \left| \langle \alpha, \beta \rangle \in \left(A^{(*^*)} / A \right) \times A \right. \\ \psi_k^{\alpha} \left(\rho_p^k \left(\langle \langle \alpha \rangle, \beta \rangle \right) \right) \left| \langle \alpha, \beta \rangle \in A \times \left(A^{(*^*)} / A \right) \right. \\ \varphi_p^k \left(\langle \alpha, \beta \rangle \right) \left| \langle \alpha, \beta \rangle \in \left(A^{(*^*)} / A \right) \times \left(A^{(*^*)} / A \right) \right. \end{cases}$$

$$\psi_k^{\chi} (\gamma) = \upsilon_k^{\chi} * \gamma ; (17)$$

$$\varphi_p^k \left(\langle \alpha, \beta \rangle \right) = \sqrt[p]{\sum_{i=1}^{\dim(\alpha)} \sum_{j=1}^{\dim(\beta)} \psi \left(\left\langle \varepsilon_{\beta}^{\alpha} (i) , \left(\rho_p^{k+1} \left(\langle \alpha_i, \beta_j \rangle \right) \right)^p , \omega_{ijk}^{\varepsilon_{\beta}^{\alpha}(i)} \right\rangle \right)};$$

$$\psi \left(\langle \delta, \lambda, \chi \rangle \right) = \chi * \left(1_{\{M,R,X,T,P\}}^{\{\delta\}} * \lambda + 1_{\{I,D,G,C,E,F\}}^{\{\delta\}} \right) ; (18)$$

$$1_{\lambda}^{\gamma} = \begin{cases} 0 \left| \emptyset = (\lambda \cap \gamma) \right. \\ 1 \left| \emptyset \subset (\lambda \cap \gamma) \right. \end{cases} , (19)$$

$$\varepsilon_{\beta}^{\alpha} \in \{M, T, R, I, D\}^{\{ i | (i \in (\mathbf{N} / \{0\})) \wedge (i \leq \max(\{\dim(\alpha)\} \cup \{\dim(\beta)\})) \}} ,$$

where υ_k^{χ} , $\omega_{ijk}^{\varepsilon(\langle \alpha, \beta \rangle)(i)}$ are weight coefficients, p is a parameter.

Therefore, the syntactic metric and the metric space are defined in a natural way in the case of representation of sc-texts by texts of a generalized formal language.

It is necessary to take into account the semantics of sc-elements and structures from them in all its forms, i.e. denotational, operational, and others [9], [23], [24], for the purpose of constructing a "semantic metric" ("semantic space metric"). Thus, it is also necessary to consider not only structures and their elements (sc-elements) but also the becoming [12] of structures and their elements in the processes of accumulation and integration of knowledge as well as models in which their specification is possible [15], [16]. To do this, we turn to models of knowledge specification and knowledge integration as well as models that can generalize these models for calculating semantic metrics for limit type structures.

C. Knowledge specification model

The knowledge specification model [8] is given by a set of (finite) formal models of (finite) fragments ontologies [4] of knowledge bases (KB) Z:

$$Z = \{ \langle G, R, O \rangle \} ,$$

G is a finite non-empty set of designations in a KB fragment, R is an oriented finite set of relations on designations in a KB fragment, O is an oriented finite set of designations interpretation functions in a KB fragment.

$$\left\langle Z \cup 2^{\left\{\omega(z)|z\in Z^2\right\}}, 2^{\Omega} \right\rangle,$$

where 2^{Ω} is a set of relations of the specification model and ω is a function of ontological model elements:

$$\Omega = \bigcup_{\{x\}\cup\{y\}\subseteq Z} \left(\{x\} \times \{y\}\right) \times 2^{\omega(x)\times\omega(y)};$$

$$\omega\left(z_i\right) = G_i \cup \{r\,|r = R_{ij}\} \cup \{o\,|o = O_{ij}\} \cup \{k\,|k \in R_{ij}\} \cup \{p\,|p \in O_{ij}\} \cup \{a\,|\langle a, v\rangle \in O_{ij}\}.$$

The knowledge specification model considers the semantics of knowledge for pairs of knowledge base fragments on finite structures within knowledge specification relations, the power of which can be unlimited. As a result, all problems of semantics analysis within the knowledge specification model are solvable for any pair of knowledge base fragments.

D. Knowledge integration

The [8] knowledge integration model is aimed at solving problems of continuous horizontal-profile knowledge integration and concludes a set of (finite) knowledge base fragments $J \subseteq 2^{V\cup E}$, where V is a set of designations (sc-elements), E is a set of connectives of designations incidence relations V.

The integration model concludes four types of relations:

- relations of ontological comparison, which for a pair of KB fragments allow obtaining a set of relations (sc-relations, comparisons) of similarity and difference, which have the property of reflexivity or irreflexivity $R_M \subseteq (J \times J) \times \Xi$;
- the fusion relation $R_F \subseteq (J \times J) \times 2^{V\times V}$;
- the (designations) mapping relation (embedding, inclusion) $R_O \subseteq J \times J$;
- the integration relation $R_I \subseteq (J \times J) \times J$;

These four types of relations define the order of solving knowledge integration problems:

- first, the similarity and difference of designations in the original fragments (texts) should be determined by alignment and comparison (in accordance with the knowledge specification model);
- then, the pairs of matching designations must be localized and fused;
- then, the mapping must be found as a mapping of each original fragment to a fragment containing the designations resulting from the merger and the remaining designations of the original fragments;
- then a fragment should be formed, which is the result of integrating a pair of original fragments, that is, a fragment onto which each of the original fragments is mapped.

The process of solving this problem is expressed in the becoming of designations as a result of merging and in the becoming (formation) of integrated fragments (texts) resulting from the integration of the original fragments.

If we trace the branches of the relation of becoming of integrated fragments (texts) from the original fragments along the mapping relation (embedding), then we can see that the formation of integrated texts generates the movement in the direction of knowledge accumulation as a natural "arrow" that regulates memorization processes. The later allows defining an ordinal scale on such texts, which sometimes may be internal: «arrow of time».

The formation (becoming) of integrated texts preserves the connectives of incidence in their structure, ensuring the convergence of structures to some integrated substructures (of a "space"), which allows proposing and establishing a (semantic) metric for such substructures which are the limit type substructures.

E. Semantic space metamodel

In accordance with the model of events and phenomena [12] and the relation of becoming [3], [37], let us consider linearly ordered sequential unions of non-intersecting chains of the generalized relation of the designations mapping in the unions of texts of languages of the unified semantic knowledge representation model.

To each sequential union, let us associate the numbering function on the universal linear scale $\square$ of the elements F ($F \subseteq J$) connected by the edges of the chain, in the order of the edges of this chain.

Let us single out a subclass of historically finite linearly ordered unions of disjoint chains of the generalized designations mapping relation.

Let us single out a subclass of locally finite linearly ordered unions of disjoint chains of the generalized designations mapping relation.

A subclass of finite linearly ordered unions of disjoint chains of a generalized relation is the intersection of the subclasses of historically finite linearly ordered unions of disjoint chains of the generalized designations mapping relation and locally finite linearly ordered unions of disjoint chains of the generalized designations mapping relation.

For the subclasses, we also consider the same numbering function as for the entire class of sequential unions. The corresponding functions will be called «histories»: $H = F^{\square}$. Accordingly, let us single out the historically finite histories $R_{HISTORICFINITE} \in S_H$, the local finite histories $R_{LOCALFINITE} \in S_H$ and the finite histories $R_{FINITE} \in S_H$; $R_{FINITE} = R_{HISTORICFINITE} \cap R_{LOCALFINITE}$. Let us also single out:

- the subclass of well-ordered histories to the beginning $R_{TOTALBACKWARD} \in S_H$;
- the subclass of well-ordered histories to the end $R_{TOTALFORWARD} \in S_H$;
- the subclass of well-ordered histories (two-sided) $R_{TOTAL} \in S_H$; $R_{TOTAL} = R_{TOTALFORWARD} \cap R_{TOTALBACKFORWARD}$.

The metamodel is given by the pair:

$$\langle H, S_H \rangle,$$

where $S_H \subseteq 2^{(H^*)}$. Let us single out the following relations of the semantic space metamodel:

- the subhistory relation $R_{SUB} \in S_H$;
- the superhistory relation $R_{SUPER} \in S_H$ (inverse to the subhistory relation $R_{SUPER} = (R_{SUB})^{-1}$);
- the continuous subhistory relation $R_{SUBCONTINOUS} \in S_H$; $R_{SUBCONTINUOUS} \subseteq R_{SUB}$;
- the continuous superhistory relation $R_{SUPERCONTINUOUS} \in S_H$ (inverse to the continuous subhistory relation $R_{SUPERCONTIONUOUS} = (R_{SUBCONTIONUOUS})^{-1}$);
- the relation of the initial subhistory $R_{START} \in S_H$;
- the relation of the final subhistory $R_{FINAL} \in S_H$;
- the history convergence relation $R_{CONVERGENCE} \in S_H$ (two sequences of becoming (of) integrated texts converge if they have a common final history $R_{CONVERGENCE} = \Big((R_{FINAL})^{-1} \circ R_{FINAL}\Big) \cup \Big(R_{FINAL} \circ (R_{FINAL})^{-1}\Big)$);
- the relation of the maximum well-ordered subhistory to the beginning $R_{FORWARD} \in S_H$; $R_{FORWARD} \subseteq R_{START}$;
- the relation of the maximum well-ordered subhistory to the end $R_{BACKWARD} \in S_H$; $R_{BACKWARD} \subseteq R_{FINAL}$;
- the relation of maximal linearly ordered strict subhistory $R_{EDGE} \in S_H$ (the edge relation $R_{EDGE} \subset R_{SUB}$);
- the relation of minimal linearly ordered strict superhistory $R_{FACE} \in S_H$ (the face relation $R_{FACE} \subset R_{SUPER}$, inverse to the edge relation $R_{FACE} = (R_{EDGE})^{-1}$);
- the enclosure relation $R_{ENCLOSE} \in S_H$;
- the disclosure relation $R_{DISCLOSE} \in S_H$, inverse to the enclosure relation $R_{DISCLOSE} = (R_{ENCLOSE})^{-1}$;
- the relation of possibility of interaction (interoperability) $R_{INTEROPERABILITY} = R_{DISCLOSE} \circ R_{ENCLOSE}$.

The following relations are reflexive: $R_{ENCLOSE}$, $R_{DISCLOSE}$, R_{START}, R_{FINAL}, R_{SUB}, R_{SUPER}, $R_{SUBCONTINUOUS}$, $R_{SUPERCONTINUOUS}$.

For locally finite histories h, the following is true:

$$\forall \iota \left((\iota \in \Box) \to \left(|h(\iota+1)/h(\iota)| \in \mathbf{N} \cup \{0\} \right) \right).$$

Sequences of integrated texts (histories) can be embedded in some (unified) "semantic space".

Such substructures as subspaces can be distinguished within spaces of various kinds (topological, vector, metric).

The metric of the metric semantic space can be constructed for the structures of the metamodel of the semantic space. For ontologies in which all NON-factors of extensional knowledge (the closed world assumption) can be eliminated during some finite history, it is possible to construct a space with a metric by introducing for each designation of a set (sc-set) a characteristic vector (or matrix (i.e. vector of vectors or matrices)). Such a vector characterizes inclusion in the history of the designation of the set (sc-set) or the possible occurrence of the designation of this set (sc-set) and contains all the corresponding components for each designation of the set (sc-set) or the possible occurrence of its element or attribute of such an occurrence. The order of the components of the vector is coordinated with the order of the becoming of designations in history. Events, values of fuzzy measures are considered as attributes of occurrences. When calculating the semantic metrics (metrics of the semantic space) for vectors, the vector of modules of the difference of their components is calculated. Next, a certain norm of the vector is calculated, which allows introducing a semantic metric. For such ontologies with a (finite) set of attributes, the metric can be calculated similarly due to the one-to-one correspondence:

$$\left(A^B\right)^C = A^{B \times C}.$$

The above is also true if the components of the characteristic vector take values on the interval $[0;1]$, that is, if there is such a non-factor as fuzziness (according to L. Zade). When attribute values do not belong to this interval, it is required to reduce attributes (sc-sets) to attributes (sc-sets) in canonical form, whose values belong to this interval.

In the case of the presence of uncertainty (the open world assumption) as a non-factor of knowledge, one can turn to the apparatus of rough sets [6]. Due to the presence of necessary operations in the algebra of distensible sets (sc-sets), for a pair of sets and for each component of the characteristic vector, it is possible to define an upper and lower estimates (by analogy with rough sets [6], [7]), which correspond to a component of some distensible subset (of an distensible superset). Further, according to the approaches for rough sets, it is possible to calculate the maximum (upper) and minimum (lower) vectors of the modules of the differences and calculate their norms. It should be noted that if one can obtain a metric space based on the norm of the upper vector then, in the worst case, one can obtain only a family of pseudometrics based on the lower vector and its norm [50]. However, one can find some pseudometric not exceeding of the value of none of this family for any pair of elements. Thus, in this case, one (or more) pairs of spaces can be obtained on one carrier, the first of which will be metric (and pseudometric) (the metric estimate is from above), and the second is pseudometric (the metric estimate is from below). With further accumulation of knowledge, as the mentioned non-factor (uncertainty) is eliminated, both spaces may converge to each other in terms of the value of pseudometrics until the values of the pseudometric (lower estimate) coincide with the values of the metric (upper estimate).

Since the transition from finite structures to non-finite structures consists of a sequence of steps, it takes time (potential infinity). This formation, in turn, is connected with the formation of integrated texts, which, in turn, is associated with their operational semantics. Thus, the very process of formation (becoming) of integrated texts (integration) determines the metrics of the semantic space.

Is the metric for designations with potentially unbounded (reflexive) semantics computable?

In the case when there is confidence that the histories are not represented either as historically finite histories or as locally finite histories, but the rules or the mechanism for generating histories to calculate (possibly only with the engagement of hypercomputing) the metrics are known, the following grammatical rules can be used.

$$0 \to XXX$$
$$1 \to YYY$$

$$YYYXXX \to YXYXYX$$
$$XXXYYY \to YXYXYX$$

$$YXYXYXYXYXYX \to YXYXYX$$
$$YYYYXYXYXXXX \to YXYXYX$$
$$XXXYXYXYXYYY \to YXYXYX$$

$$YYYYXYXYXYYY \to YYYXXXYYY$$
$$XXXYXYXYXXXX \to XXXYYYXXX$$

$$YXYXYX \to \varepsilon$$

$$XXX \to \varepsilon$$
$$YYY \to \varepsilon$$

. They can be used as rules of game [23], [24]. It is possible to calculate some value of the metric by counting the number and remembering the order in which these rules are applied to obtain a given set of vectors in accordance with the semantics of such a game.

In a finite time, it is practically impossible without hyper(super)computations to distinguish (semantically) a knowledge base with finite semantics from a knowledge base with potentially unlimited semantics.

Designations with potentially unlimited semantics can be represented as structures with finite semantics, for which the metric is defined with the accuracy of a given interval.

F. Space-time and semantic space order

The objects and connections of the subject domain, as well as the texts themselves, are assumed to be located in physical time and space (space-time [64], [65], [14]). It is considered inappropriate if the complexity of the spatio-temporal structures presented in the texts significantly exceed the complexity of the structure of physical space and time. Elementary events are connected by the relation of becoming. These events are assumed to correspond to the elements (points) of space-time. Thus, they are represented and explicitly expressed in the semantics of designations in the texts of the languages of the model of the unified semantic representation of knowledge.

This curve can be a curve of the second order (a quadric). Any graph can be represented by geometric shapes in three-dimensional space without intersections of shapes that would correspond to its non-incident vertex or edge. For example, it is in the case of (1) representation of graph vertices in three-dimensional space by straight lines intersecting a parametrically given convex curve lying in a plane along a given (not forming a cycle) step of the parameter and also perpendicular to this plane at intersection points and (2) representation of arcs by straight lines intersecting pairs of straight lines representing vertices. Each line lies in its own plane parallel to the plane in which the convex curve lies. If there is a metric space for the physical space-time then it is possible to define the metric of elementary events with the metric space on them for the knowledge base.

Distensible sets can be considered as the average of the set of points of elementary events, then the space-time metric for the pair of designations for distensible sets s and n:

$$\mathrm{M}\left(\langle s, n\rangle\right) = \left(\frac{\sum_{x=1}^{|V(s)|}\sum_{y=1}^{|V(n)|} d\left(\left\langle V(s)_x, V(n)_y\right\rangle\right)}{|V\left(s\right)| * |V\left(n\right)|}\right),$$

where V is a function of the feature vector (points of elementary events), d is the metric of points of elementary space-time events.

In general, the relation of becoming is not antisymmetric, but its condensation can set the order, which can be represented by structures formed in the processes of semantic logging [3], [66]. Recording the processes [66] of integration (fragments) of texts sets the internal (temporal) order and allows you to reverse these processes along each of the branches of the formation of integrated (fragments) of texts. It is assumed that this order corresponds to the temporal order in space-time.

VI. Application for problem solving of taxonomy optimization

Let's consider the application of concepts related to the concept of a metric space to solve the problem of optimizing the representation of taxonomies This task lies in the fact that it is necessary to minimize the number of operations on the states of the taxonomy knowledge processing model and the number of permanently stored classes. If we consider taxonomy classes as classical sets, then for their expression we can choose the operational basis of set-theoretic operations. They are the operation of removing the taxonomy class and constructive operations of the algebra of sets: intersection and symmetric difference ($\cap$, $-$). Examples of other operational bases of set algebra are set difference with union or intersection of sets. In the mentioned basis, the union of sets is expressed:Examples of other operational bases of set algebra are set difference with union or intersection of sets. The union of sets is expressed through the mentioned basis as follows:

$$A \cup B = (A \cap B) - A - B.$$

In addition to the operational basis, a model basis is considered. The model basis is the minimum number of taxonomy classes through which any of its classes can be expressed. Thus, the taxonomy is defined in accordance with the structural approach by two sets: an initial state, a set of classes and a set of rules (operations), with which it is possible to get any family of taxonomy classes Let's define the concept of closure of a taxonomic structural model (operational-model closure) in order to formulate the problem: $\overline{\langle\{\sigma\}, \lambda\rangle} = \left\langle\left(\bigcup_{\varphi\in\lambda}\varphi\right)^{\circ}(\sigma), \left(\bigcup_{\varphi\in\lambda}\varphi\right)^{\circ}\right\rangle$, where $\sigma \subseteq \theta$ is a subset of the set of all taxonomy classes θ, $\lambda \subseteq \left(2^{\theta}\right)^{\left(2^{\theta}\right)}$ is a set of operations on taxonomy classes.

Let's consider the set of all possible transitions $\bigcup_{\varphi\in\lambda}\varphi$ on the set of operations λ.

Task.

Given:

$$\Gamma \subseteq 2^{\bigcup_{\varphi\in\lambda}\varphi};$$

function of useful output from job (information capacity):

$$\rho \in \delta^{2^{\theta}\times 2^{\theta}};$$

cost function of (time) resources for job:

$$\tau \in \delta^{2^{\theta}\times 2^{\theta}};$$

functions:

$$\pi \in \delta^{\{|\gamma| | \gamma\in\Gamma\}};$$

$$\psi \in \delta^{\{|\gamma| | \gamma\in\Gamma\}};$$

α, β.

Required:

$$\alpha * |\sigma| + \beta * |\lambda| \to \min;$$

$$\overline{\langle\sigma,\lambda\rangle} \to \max;$$

$|\Gamma| \to \max$ for any $\gamma\in\Gamma$ to satisfy

$$\sum\nolimits_{\chi\in\gamma}\tau(\chi) \le \pi(|\gamma|) * \sum\nolimits_{\chi\in\gamma}\rho(\chi);$$

$$\sum\nolimits_{\chi\in\gamma}\tau(\chi) \ge \psi(|\gamma|) * \sum\nolimits_{\chi\in\gamma}\rho(\chi).$$

Note that if the metric is defined $\mu(\langle A,B\rangle) = |A-B|$ then the cardinality (norm) of the set can be expressed inversely $|S| = \mu(\langle S,\emptyset\rangle)$.

Assume that the inequalities with respect to π and ψ are always satisfied. Taking on such additional requirements as $\overline{\langle\{\sigma\},\lambda\rangle} = \left\langle 2^{\theta}, 2^{\theta}\times 2^{\theta}\right\rangle$ we are considering the following concepts in order to establish the basis of the model.

Introscalar product of sets is [8]:

$$is(\langle A,B,S\rangle) = |S-(S\cap(A-B))| - |S\cap(A-B)|;$$

$$is(\langle A,B,S\rangle) = |S| - 2*|(S\cap A)-(S\cap B)|;$$

$$is(\langle A,B\rangle) = is(\langle A,B,A\cup B\rangle).$$

Introcoscalar product of sets is:

$$ics(\langle A,B,S\rangle) = \pm 2*\sqrt{|S-(S\cap(A-B))| * |S\cap(A-B)|};$$

$$ics(\langle A,B\rangle) = ics(\langle A,B,A\cup B\rangle).$$

These concepts make it possible to define an analogue of trigonometric functions for sets and a formal analogue of the Euler formula:

$$icos(\langle A,B,S\rangle) = \frac{is(\langle A,B,S\rangle)}{|S|};$$

$$isin(\langle A,B,S\rangle) = \frac{ics(\langle A,B,S\rangle)}{|S|};$$

$$iexp(\langle A,B,S\rangle) = icos(\langle A,B,S\rangle) + i * isin(\langle A,B,S\rangle);$$

$$iexp(\langle A,B\rangle) = iexp(\langle A,B,A\cup B\rangle).$$

Two sets are subintroorthogonal if and only if:

$$iexp(\langle A,B,S\rangle) = \pm i;$$

$$iexp(\langle A,B\rangle) = \pm i.$$

Two sets are subintroorthogonal if and only if the square of the introscalar product is minimal:

$$is(\langle A,B,S\rangle)^2 \to \min;$$

$$is(\langle A,B\rangle)^2 \to \min.$$

A family of sets is called a introorthogonal basis if and only if any different of them are pairwise introorthogonal.

A family of sets is called a subintroorthogonal basis [8] iff any different of them are pairwise subintroorthogonal.

Thus, the subintroorthogonal basis can be chosen as the basis of the model (Fig. 16).

In the case of stronger restrictions it is required to additionally determine the maximum possible value of the area Γ.

The introduced models and concepts make it possible to transfer the obtained results to an distensible taxonomy, which has a distensible set of classes. They also provide an opportunity to simulate the adaptation of the model basis of an distensible taxonomy when new classes are added to it. It can also be used when distensible sets (kinds) are used instead of taxonomy classes.

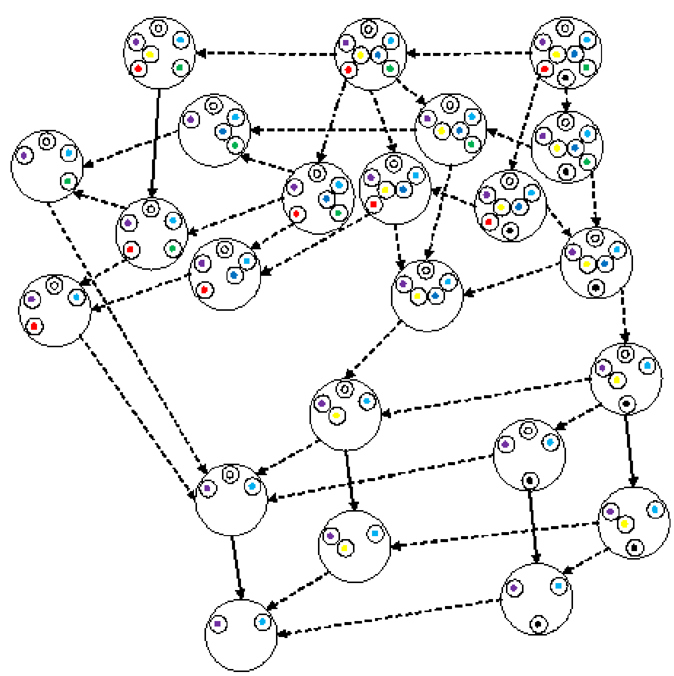

Fig. 16. Taxonomy states-transition diagram.

VII. Conclusions

The proposed models and approaches form the basis for solving problems in knowledge-driven systems aimed at ensuring interoperability and convergence of OSTIS ecosystem users and agents [69], [3], [9].

It seems promising to further study the properties of the semantic space and develop the OSTIS [5], [2] standard and technology based on the results obtained, including solving the problems of quality analysis and management of knowledge base and agents of intelligent computer systems.

References

[1] A.S. Narinyani. NE-faktory: netochnost' i nedoopredelennost' – razlichie i vzaimosvyaz' [Non-factors: inaccuracy and underdetermination – difference and interrelation]. Izv RAN (RAS). Ser. Teoriya i sistemy upravleniya 5, 2000. pp. 44—56.

[2] V.V. Golenkov. Otkrytyi proekt, napravlennyi na sozdanie tekhnologii komponentnogo proektirovaniya intellektual'nykh sistem [An open project aimed at creating a technology for the component design of intelligent systems], Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems], 2013, pp. 55—78.

[3] V.P. Ivashenko. Modeli resheniya zadach v intellektual'nykh sistemakh. V 2 ch. Ch. 1 : Formal'nye modeli obrabotki informatsii i parallel'nye modeli resheniya zadach : ucheb.-metod. posobie [Models for solving problems in intelligent systems. In 2 parts, Part 1: Formal models of information processing and parallel models for solving problems: a tutorial] Minsk, BGUIR, 2020. 79 p.

[4] T.A. Gavrilova, V.F. Khoroshevsky. Bazy znanii intellektual'nykh sistem [Knowledge bases of intelligent systems], Saint Petersburg, Piter, 2001. 384 p.

[5] V.V. Golenkov, N.A. Gulyakina, D.V. Shunkevich. Otkrytaya tekhnologiya ontologicheskogo proektirovaniya, proizvodstva i ekspluatatsii semanticheski sovmestimykh gibridnykh intellektual'nykh komp'yuternykh system [Open technology for ontological design, production and operation of semantically compatible hybrid intelligent computer systems], Minsk, Bestprint, 2021. 690 p.

[6] Z. Pawlak. Rough sets. International Journal of Parallel Programming, 1982, vol. 11, no. 5, pp. 341—356.

[7] D. Dubois, H. Prade. Rough fuzzy sets and fuzzy rough sets International Journal of General Systems, 1990, vol. 17, no. (2-3), pp. 191-–209.

[8] V.P. Ivashenko. Modeli i algoritmy integratsii znanii na osnove odnorodnykh semanticheskikh setei (disc. na soiskanie stepeni kand. tekhn. nauk: 05.13.17) [Models and algorithms for knowledge integration based on homogeneous semantic networks (thesis for the degree of Candidate of Technical Sciences: 05.13.17)] , Minsk, BGUIR, 2014, 152 p.

[9] V.P. Ivashenko Operatsionnaya semantika mnogoagentnykh sistem obrabotki znanii [Operational semantics of multi-agent knowledge processing systems], Information Technologies and Systems, 2020, Minsk, BGUIR, pp. 78–79.

[10] V.P. Ivashenko. Modeli i algoritmy integratsii znanii na osnove odnorodnykh semanticheskikh setei [Models and algorithms for knowledge integration based on homogeneous semantic networks], Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems], 2015, Minsk, BGUIR, pp. 111—132.

[11] V.P. Ivashenko Tekhnologiya razrabotki programmnykh komponentov intellektual'nykh sistem na osnove integratsionnoi platformy [Technology for the development of software components of intelligent systems based on an integration platform], Information Technologies and Systems, 2021, Minsk, BGUIR, pp. 84–85.

[12] V.P. Ivashenko. Ontologicheskaya model' prostranstvenno-vremennykh otnoshenii sobytii i yavlenii v protsessakh obrabotki znanii [Ontological model of the space-time relations of the event and the phenomenon in the processes of knowledge processing] 2017, vol. 5, no. 107, Minsk, BSUIR, pp. 13–17.

[13] V.P. Ivashenko. String processing model for knowledge-driven systems. Minsk, Doklady BGUIR, 2020, vol. 18, no. 6, pp. 33–40.

[14] Yu. Manin, M. Marcolli. Semantic spaces. Published, Location, 2016. 32 p. (arXiv)

[15] M.A. Perez, D.I. Spivak. Toward formalizing ologs, Preprint arXiv:1503.08326, 2015. 35 p.

[16] D.I. Spivak, R.E. Kent. Ologs: a categorical framework for knowledge representation. Preprint arXiv:1102.1889, 2011. 52 p.

[17] V.V. Golenkov, O.E. Eliseeva, V.P. Ivashenko. Predstavlenie i obrabotka znanii v parallel'nykh grafodinamicheskikh assotsiativnykh mashinak [Representation and processing of knowledge in parallel graphodynamic associative machines], Minsk, BGUIR, 2001. 412 p.

[18] H. Alt, M. Godau, Computing the Fr ́echet distance between two polygonal curves, Int. J. Comput. Geom. Appl., 1995, vol. 5, pp. 75–91.

[19] E. Study. Kürzeste Wege im komplexen Gebiet. Mathematische Annalen (in German). Springer Science and Business Media LLC, 1905, vol. 60 no. 3, pp. 321–378.

[20] G. Rozenberg, A. Salomaa. Handbook of Formal Languages, Volume 1: Word, Language, Grammar. Verlag, Berlin, Heidelberg, Springer, 1997. 873 p.

[21] B. Smith. Metody i algoritmy vychislenii na strokakh [Computing Patterns in Strings], Moscow, OOO "I.D. Williams", 2006. 496 p.

[22] D.A. Pospelov. Situatsionnoe upravlenie: teoriya i praktika [Situational management: theory and practice], Moscow, Nauka, 1986. 288 p.

[23] A. Blass. A game semantics for linear logic. Annals of Pure and Applied Logic, 1992, vol. 56, pp. 183–220.

[24] J.H. Conway. On Numbers and Games. AK Peters/CRC Press, 2000. 242 p.

[25] G. Plotkin. Call-by-name, call-by-value and the lambda-calculus. Theoretical Computer Science, 1975, vol. 1, no. 2. pp. 125–159.

[26] D. Scott, Ch. Strachey. Toward a mathematical semantics for computer languages. Proceedings of the Symposium on Computers and Automata, Microwave Research Institute Symposia Series, Polytechnic Institute of Brooklyn Press, New York, 1971, vol. 21, pp. 19—46.

[27] F. Daneš. On Prague school functionalism in linguistics. Functionalism in Linguistics, 1987, pp. 3–38.

[28] A. R. D. Mathias. Unordered pairs in the set theory of Bourbaki 1949, Archiv der Mathematik, 2010, vol. 94, pp. 1–10.

[29] W. Quine. On ordered pairs. Journal of Symbolic Logic, 1945, vol. 10, no. 3, pp. 95—96.

[30] J. Heijenoort. From Frege to Gödel. A source book in mathematical logic, 1879–1931. Cambridge, Mass., Harvard University Press, 1967, 664 p.

[31] C. Kuratowski. Sur la notion de l'ordre dans la Théorie des Ensembles. Fundamenta Mathematicae, 1921, vol. 2, no. 1,pp. 161–171.

[32] A.P. Morse. A Theory of Sets. Academic Press, 1965, 130 p.

[33] A. Levy. Basic Set Theory, Springer-Verlag, 1979, 391 p.

[34] K.J. Devlin. Fundamentals of contemporary set theory. Universitext. Springer-Verlag, 1979, 182 p.

[35] J.M. Lee. Introduction to Topological Manifolds, Springer, 2011, 452 p.

[36] J. McCarthy. Recursive Functions of Symbolic Expressions and Their Computation by Machine, Part I. Communications of ACM, New York, 1960, vol. 3, no. 4, pp. 184–195.

[37] G. W. F. Hegel. Nauka logiki [Science of Logic]: v 3-h t. T.1, Moscow, "Mysl"', 1970. 501 p.

[38] Platon. Sobr. soch. v 4-kh tomakh. [Collected works in 4 volumes.]: T. 3, Moscow, "Mysl"', 1994. 654 p.

[39] J. Cottingham, René Descartes: Meditations on First Philosophy: With Selections from the Objections and Replies, Cambridge, Cambridge University Press, 2015. 282 p.

[40] R.T.W. Arthur. Leibniz on Time, Space, and Relativity, Oxford University Press, 2022, 432 p.

[41] D. Hilbert. On the infinite. In Philosophy of Mathematics (1984), Selected Readings, Cambridge, Cambridge University Press, 2012, pp. 183–201.

[42] D. Bohm and B.J. Hiley. The Undivided Universe: An Ontological Interpretation of Quantum Theory, London, Routledge, 1993. xii + 397 p.

[43] D. Bohm. Wholeness and the Implicate Order, London, Routledge, 2002. 284 p.

[44] V.V. Nalimov, Zh.A. Drogalina. Real'nost' nereal'nogo. Veroyatnostnaya model' bessoznatel'nogo [The reality of the unreal. Probabilistic model of the unconscious], Moscow, Mir idei, AO Akron, 1995. 432 p.

[45] V.V. Nalimov. Spontannost' soznaniya. Veroyatnostnaya teoriya smyslov i smyslovaya arkhitektonika lichnosti [Spontaneity of consciousness. Probabilistic theory of meanings and semantic architectonics of personality], Moscow, Prometei, 1989. 287 p.

[46] V.V. Nalimov. Veroyatnostnaya model' yazyka. O sootnoshenii estestvennykh i iskusstvennykh yazykov [Probabilistic model of language. On the relationship between natural and artificial languages] Moscow, Nauka, 1979. 304 p.

[47] I.N. Taganov, Yu.I. Babenko. Antivremya i antiprostranstvo [Anti-time and anti-space], Saint Petersburg, RAN, 2016. 200 p.
[48] S. Roman, Advanced Linear Algebra, Graduate Texts in Mathematics, vol. 135 (2nd ed.), Berlin, New York, 2005. 488 p.
[49] M.A. Armstrong, Basic Topology [1979], Undergraduate Texts in Mathematics, Springer, 1983. 263 p.
[50] L. Collatz. Functional Analysis and Numerical Mathematics, New York, San Francisco, London, Academic Press, 1966. xx + 473 p.
[51] J. Heinonen. Lectures on analysis on metric spaces, New York, Springer, 2001. X+141 p.
[52] A.A. Adronov, A.A. Vitt, S.E. Haikin. Teoriya kolebanii (2-e izd.) [Oscillation theory (2nd ed.)], Moscow, Nauka. 1981. 586 p.
[53] GOST 7427-76. Geometricheskaya optika. Terminy, opredeleniya i bukvennye oboznacheniya. S izmeneniem №1, utverzhdennym v iyule 1982 g. [GOST 7427-76. Geometric optics. Terms, definitions and letter designations. With change No. 1, approved in July 1982]. Moscow, Izdatel'stvo standartov, 1988. 19 p.
[54] H.H. Schaefer, M.P. Wolff. Topological Vector Spaces, New York, Springer New York Imprint Springer, 1999. 361 p.
[55] I. Mel'˘cuk. Language: from Meaning to Text. Ed. by D. Beck. Moscow & Boston, 2016.
[56] J. Harris. Algebraic Geometry: A First Course, New York, Springer, 1992. 330 p.
[57] A. Kostrikin, Yu. Manin. Linear Algebra and Geometry, Gordon and Breach Science Publishers, 1997. 320 p.
[58] W. Lowe. Towards a theory of semantic space, in Proceedings of the 23rd Conference of the Cognitive Science Society, 2001, pp. 576–581.
[59] Yu. I. Manin. Zipf's law and L. Levin's probability distributions. Functional Analysis and its Applications, 2014, vol. 48, no. 2. Preprint arXiv:1301.0427.
[60] V.V. Martynov. V tsentre soznaniya cheloveka [At the center of human consciousness], Minsk, BGU, 2009. 272 p.
[61] V.V. Martynov. Yazyk v prostranstve i vremeni: k probleme glottogeneza slavyan (2-e izd.) [Language in space and time: to the problem of glottogenesis of the Slavs (2nd ed.)], Moscow, Editorial URSS, 2004. 106 p.
[62] A.N. Gordey. Teoriya avtomaticheskogo porozhdeniya arkhitektury znanii (TAPAZ-2) i dal'neishaya minimizatsiya semanticheskikh ischislenii [The theory of automatic generation of knowledge architecture (TAPAZ-2) and further minimization of semantic calculus]. Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems], 2014, Minsk, BGUIR. pp. 49–64.
[63] C. Marijuan. Finite topologies and digraphs. Proyecciones (Antofagasta), 2010, vol. 29, pp. 291–307.
[64] R. Penrose. The Road to Reality, Oxford, Oxford University Press, 2004. 1094 p.
[65] A.K. Guts. Khronogeometriya. Aksiomaticheskaya teoriya otnositel'nosti [Chronogeometry. Axiomatic theory of relativity], Omsk, OOO "UniPak", 2008. 340 p.
[66] V. Ivashenko, N. Zotov, M. Orlov. Semantic Logging of Repeating Events in a Forward Branching Time model. Pattern Recognition and Information Processing (PRIP'2021), United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk, 2021, pp. 149–152.
[67] M. Bansal, D. Burkett, G. de Melo, Gerard, D. Klein, Dan. Structured Learning for Taxonomy Induction with Belief Propagation. 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 – Proceedings of the Conference, 2015, vol. 1, pp. 1041–1051.
[68] Yu.I. Zhuravlyov. Ob algebraicheskom podkhode k resheniyu zadach raspoznavaniya ili klassifikatsii [On an algebraic approach to solving problems of recognition or classification]. Problemy kibernetiki, 1978, vol. 33. pp. 5–68.
[69] V.V. Golenkov, D.V. Shunkevich. Agentno-orientirovannye modeli, metodika i sredstva razrabotki sovmestimykh reshatelei zadach intellektual'nykh sistem [Agent-based models, methodology and tools for developing compatible problem solvers for intelligent systems]. Programmnye produkty i sistemy, 2020, vol. 33, no. 3. pp. 404–412.
[70] J.A. Fodor. The Language Of Thought. Crowell Press, 1975. 214 p.
[71] A.N. Kolmogorov. Problems of Probability Theory. Theory of Probability & Its Applications, 1994, vol. 38, no. 2, pp. 177–178.

Универсальный язык смыслового представления знаний и смысловое пространство

Ивашенко В.П.

Статья рассматривает модели и средства, обеспечивающие унифицированное представление знаний и их интеграцию в рамках «смыслового пространства». Для этого вводится понятие «обобщённого формального языка», позволяющего выявить с целью анализа взаимоотношение формальных языков и известных языков представления знаний, включая семантические сети.

На основе этого анализа уточняется семантика языков модели унифицированного представления знаний, вводится язык, являющийся основой стандарта для технологии разработки интеллектуальных систем, и даётся концепция «смыслового пространства», ориентированного использование в целях оценки качества интеллектуальных компьютерных систем в рамках технологии OSTIS. Рассматриваются прикладные задачи на основе предложенных моделей и дальнейшие перспективы развития технологии и её компонентов.

Received 11.11.2022