The presentation of evolutionary concepts in problems of information support to implement the best available technologies

Sergey V.Kosikov[1], Viacheslav E. Wolfengagen[2], Larisa Yu.Ismailova[2]

[1] Institute for Contemporary Education "JurInfoR-MGU",
Moscow, 119435 RF,
kosikov.s.v@gmail.com

[2] National Research Nuclear University "MEPhI"
(Moscow Engineering Physics Institute),
Moscow, 115409 RF
{jir.vew@gmail.com, lyu.ismailova@gmail.com}

The paper considers an approach to solving the problem of supporting the semantic stability of information system (IS) objects. A set of IS objects is addressed as a semantic network consisting of concepts and frames. The interpretation that assigns intensional (meaning) and extensional (value) characteristics to network designs is connected to the constructions of the semantic network. The interpretation in the general case depends on the interpreting subject, time, context, which can be considered as parameters. The possibility to preset a consistent interpretation for a given semantic network is regarded as a semantic integrity, and the possibility to control changes in interpretation when the parameter is changed is regarded as semantic stability. Among the tasks related to supporting semantic stability, the problem of modelling evolutionary concepts (EC) is highlighted. It is proposed to construct a computational model of EC based on the theory of categories with a significant use of the concept of variable domain. The model is constructed as a category of functors, and it is shown that the Cartesian closure of the basic category implies Cartesian closure of the category of models. The structure of the exponential object of the category of models has been studied, and it is shown that its correct construction requires taking into account the evolution of concepts. The testing of the model's constructions was carried out when lining the means of semantic support for the implementation of the best available technologies (BAT).

Keywords: information system, semantic network, semantic modeling, semantic stability, data model, computational model, theory of categories

1. Introduction

The Internet technologies development brings to the change in the principles of working with information. Information systems, the content of which has been previously prepared and verified, are replaced by systems, in which the content is available to everyone for editing. The fact that there is an undefined circle of persons capable of creating and changing information objects within the system leads to a number of problems [2], the support of the semantic integrity of the system being the most important among them.

The most general representation of the content of the information system involves the identification of objects and their links. Objects (links) within the information system, as a rule, correspond to real or conceivable objects (links) in the real world - the subject domain of the system. Establishing such a correspondence, as a rule, is considered as sensing the information object within the system (we temporarily abstract ourselves from the difference between meaning and value) or setting its semantics.

Further steps for detailing the representation of the content of the information system are associated with the identification of individuals that correspond to specific objects of the domain and general notions (concepts) corresponding to the classes of individuals grouped according to some principle. The set of individuals assigned to the concept corresponds to the classical notion of the volume (or meaning) of the concept, and the set of properties and links of the concept defines its meaning or content [12].

Semantic integrity in the most general form can be understood as follows. First, the various subjects interacting with the system must give the same object the same meaning. Secondly, the object can appear in different places of the system's content, i.e., in different contexts. In this case, all occurrences of the object should also be given the same meaning.

The above informal considerations can be specified by various ways. The representation of the content of the information system is refined with the help of conceptual modelling methods. The conceptual model is made, as a rule, of concepts corresponding to information objects or entities, and frames corresponding to the connections of entities. Further specification is connected to the adoption of a formal system, which describes the concepts and frames. The informal semantics of entities and connections can be specified by constructing the formal semantics of the considered system, i.e., setting the interpretation of the formal system.

The provision of the semantic integrity of the information system requires a model that can reflect both the loss of such integrity and its restoration. If it is necessary to take into account the subjective view on the content of the system, the modelling of the dependence of the system's constructions interpretation on the subject is needed. This requirement imposes restrictions on the formal systems used. For example, the models of classical logic predicates do not contain tools for modeling such dependencies.

Similarly, accounting for the interpretation of various occurrences of the information object requires modelling the dependence of the interpretation on the context. Such models are also difficult to make when using classical logic. As a rule, they are made within the framework of pragmatics, which takes into account the situation of its use when interpreting the expression.

Specific problems appear when the information system is used for a long time. In this case, it becomes necessary to take into account the dependence of the information objects interpretation on time [5]. As time goes, both the volume of entities, represented within the information system, and the content of entities can change. Accordingly, the connections, in which the entities are nested, can also change. Modelling such changes also suggests the use of special logical systems, as a rule, different variants of temporal logic.

All considered sources of changes assume that the dependence of the information system objects interpretation on one or another factor that can be addressed as a parameter. Consideration of parameterized interpretations that can be associated both with one way of parameterization and with several different methods is of great interest. In this case, the methods of matching the parameterization on various bases are of interest.

Saving the semantic integrity of the information objects system – concepts and frames – when changing the parameters on which the interpretation depends, can be considered as the semantic stability of the system [3, 8, 10]. Dependence of the concepts and frames interpretation on the parameter brings to the need to consider evolutionary interpretations [11]. Thus, the task of modelling evolutionary concepts is the central part of the means of supporting the system’s semantic integrity and stability, which makes grounds for its relevance.

2. Task to develop a model of evolutionary concepts

The need to support semantic integrity at the informal level is taken into account when supporting the practical information systems, especially those systems that are oriented to changing the information by an undefined circle of persons. So, in electronic encyclopedias such as Wikipedia special pages are created, on which the multi-valued terms are given the variants of their interpretation. In case of creating a reference to such a term, the necessary interpretation can be chosen among the proposed ones on the technical page or created anew. It should be noted that the choice is made entirely in manual mode, there are no automation options.

Up to the present moment the frame languages are the most popular languages for the representation of the subject domain, and the frame itself is understood as a “hierarchically ordered representation of the standard reality situation”. At the same time, the representation of situations in which an individual changes its former properties and begins to manifest itself as an individual with new properties, becoming indistinguishable from already existing individuals with these latter properties, does not receive a proper solution within the framework of known formalisms.

The need to develop consistent methods for describing the changing notions about the subject domain leads to setting the task of developing modelling tools for evolutionary concepts that provide for:

- Expressions of the classical means for constructing typed descriptions of the subject domain (the Cartesian products, sums, etc.);
- Descriptions of the various types of parameterization (according to subject, time, context), as well as a combination of different methods of parameterization;
- Possibility to integrate the modelling environment with the computing environment.

An important requirement for the proposed family of models is the possibility of its integration with a supporting computing environment of applicative type. Such integration provides the possibility of software support for the means of describing evolution through the use of higher order functions and makes an element of novelty of the proposed model.

3. Variable domains in the theory of categories. Development of the model for support to evolutionary concepts

3.1. Cartesian closed category

Cartesian product is an abstraction of the set of ordered pairs. A strict definition looks like this. The cartesian product of objects a and b in category C is the object d with the pair of arrows $p : d \to a$, $q : d \to b$ that for any object c and arrows $f : c \to a$, $g : c \to b$ there is a unique arrow $h : c \to d$, that:

$$f = p \circ h, \qquad g = q \circ h.$$

Object d is denoted below as $a \times b$. Arrows p and q are called projections for $a \times b$. Since the arrow h, built by f and g, is the unique arrow with the specified property, it can be considered as a function of f and g. This function is denoted as

$$\langle f, g \rangle.$$

In new denotations the properties of cartesian product look like

$$f = p \circ \langle f, g \rangle,$$
$$g = q \circ \langle f, g \rangle,$$
$$h = \langle p \circ h, q \circ h \rangle,$$

where $h : c \to a \times b$. Sometimes it is convenient to consider an arrow $\langle f \circ p, g \circ q \rangle$ for arrows $f: a \to c$ and $g : b \to d$. This arrow is named the product of f and g and is denoted as $f \times g$.

Not every category have the cartesian product for any two objects. If it is so, the category is named decart.

Category Set of sets is decart. Decart products in it are built as usual as sets of pairs: if A and B are two sets (objects of category Set), their decart product is a set $\{(a, b) | a \in A, b \in B\}$.

Projections are defined by the natural way:

$$p\,(a, b) = a, \qquad q\,(a, b) = b.$$

The arrow of pair evaluation $\langle f, g \rangle$ for $f : C \to A$, $g : C \to B$ is defined so:

$$\langle f, g \rangle(c) = (fc, gc),$$

where $c \in C$. It is easy to check that the product of arrows is evaluated so:

$$(f \times g)(a, b) = (fa, gb).$$

3.2 Cartesian product in a functor category.

We show now that the category $Set^{C^{op}}$ is a decart category. Really, decart product can be built "pointwise". We consider now the construction in more details.

Let U and V be two functors from $Set^{C^{op}}$. Then for defining of their decart product it is necessary to define a functor $U \times V$ and projections (which must be the natural transformations) $p : U \times V \to U$ and $q : U \times V \to V$. Then it is necessary for all natural transformations $\mu : W \to U$ and $\nu : W \to V$ define the natural transformation $<\mu, \nu>$ and to check the properties of decart product.

We define the functor $U \times V$ as follows:

$$(U \times V)_A = U_A \times V_A, \qquad (U \times V)f = Uf \times Vf.$$

Then for every A in Set are defined the usual projections $p: U_A \times V_A \to U_A$ and $q: U_A \times V_A \to V_A$.

We use these projections as the components of a natural transformation – projection as follows:

$$p_A = p, \ \ q_A = q.$$

We have $(U \times V)f : (U \times V)_A \to (U \times V)_A$, so in the definition of the restriction mapping it is convenient to denote an element of $(U \times V)_A$ on which we define the value, as (a, b). We apply now $(U \times V)f$ to the element (a, b) from $(U \times V)_A$. We have $(U \times V)f\,(a, b) = (Uf \times Vf)\,(a, b) = (Ufa, Vfb)$, or

$$(a, b) \uparrow f = (a \uparrow f, b \uparrow f).$$

Now we have to verify the correctness of the given definition, that is to verify the properties of the functor and the natural transformation. We verify first that $U \times V$ defined above is a functor.

1. $U \times V$ preserves the composition. Really,

$$(U \times V)(f \circ g) = U(f \circ g) \times V(f \circ g) = (U(f) \circ U(g)) \times (V(f) \circ V(g)) =$$
$$= <U(f) \circ U(g) \circ p, V(f) \circ V(g) \circ q> = <U(f) \circ p, V(f) \circ q> \circ <U(g) \circ p, V(g) \circ q> =$$
$$= (Uf \times Vf) \circ (Ug \times Vg) = (U \times V)f \circ (U \times V)g.$$

2. $U \times V$ preserves the unit arrows. Really,

$$(U \times V)1 = U1 \times V1 = 1 \times 1 = <1 \circ p, 1 \circ q> = <p, q> = 1.$$

So $U \times V$ is really a functor.

We verify now that p and q are natural transformations. So we check the condition of naturality

$$p_B \circ (U \times V)f = p \circ (Uf \times Vf) = p \circ <Uf \circ p, Vf \circ q> = Uf \circ p = Uf \circ p_A.$$

The condition satisfies so p is really a natural transformation. The check for q is analogous. This check shows that projections in Set are defined really naturally.

We can make the evaluations above also in terms of restriction mappings. The check of composition preservation is:

$$(a, b) \uparrow (f \circ g) = (a \uparrow (f \circ g), b \uparrow (f \circ g)) = ((a \uparrow f) \uparrow g, (b \uparrow f) \uparrow g) =$$
$$= (a \uparrow f, b \uparrow f) \uparrow g = ((a, b) \uparrow f) \uparrow g.$$

The check of the unit is analogous. The check of the naturality of the projection is:

$$p_B\,((a, b) \uparrow f) = p_B\,(a \uparrow f, b \uparrow f) = a \uparrow f = (p_A\,(a, b)) \uparrow f.$$

The indexes of objects for projection can be reconstructed in a unique way, and so they can be omitted because of natural property of the projection.

We define now the arrow of pair evaluation. Let $\mu : W \to U$ and $\nu : W \to V$ be arrows in the category $Set^{C^{op}}$. We define $<\mu, \nu>$ as follows:

$$<\mu, \nu>_A = <\mu_A, \nu_A>.$$

We check that defined arrow is a natural transformation, i.e. check the naturality condition:

$$\langle\mu, \nu\rangle_B \circ Wf = \langle\mu_B, \nu_B\rangle \circ Wf = \langle\mu_B, \nu_B\rangle \circ Wf = \langle\mu_B \circ Wf, \nu_B \circ Wf\rangle = \langle Uf \circ \mu_A, Vf \circ \nu_A\rangle =$$
$$= \langle\langle Uf \circ p \circ \langle\mu_A, \nu_A\rangle, Vf \circ q \circ \langle\mu_A, \nu_A\rangle\rangle = \langle Uf \circ p, Vf \circ q\rangle \circ \langle\mu_A, \nu_A\rangle =$$
$$= ((U \times V)f) \circ \langle\mu, \nu\rangle_A.$$

The condition is satisfied, so the natural transformation $\langle\mu, \nu\rangle$ is defined correctly.

In the terms of restriction mapping the computation above is represented as follows. We have to demonstrate that the restriction operation “$\uparrow f$” can pass through the application $\langle\mu, \nu\rangle$ to the argument. We have:

$$\langle\mu, \nu\rangle(c \uparrow f) = (\mu(c \uparrow f), \nu(c \uparrow f)) = (\mu\, c \uparrow f, \nu\, c \uparrow f) = (\mu\, c, \nu\, c) \uparrow f = (\langle\mu, \nu\rangle(c)) \uparrow f,$$

where $c \in W_A$. The indices of objects in natural transformations are omitted (due to naturality).

Now we have to check the characteristic properties of the projection and pairing. We show now that $p \circ \langle\mu, \nu\rangle = \mu$. We calculate a component $(p \circ \langle\mu, \nu\rangle)_A$ and see that it is equal to μ_A.
We have

$$(p \circ \langle\mu, \nu\rangle)_A = p_A \circ \langle\mu, \nu\rangle_A = p \circ \langle\mu_A, \nu_A\rangle = \mu_A.$$

So components of $p \circ \langle\mu, \nu\rangle$ are really identical with components of μ and so $p \circ \langle\mu, \nu\rangle = \mu$.

We show now that $\langle p \circ \eta, q \circ \eta\rangle = \eta$, where $\eta : W \to (U \times V)$. We have

$$\langle p \circ \eta, q \circ \eta\rangle_A = \langle p_A \circ \eta_A, q_A \circ \eta_A\rangle = \langle p \circ \eta_A, q \circ \eta_A\rangle = \eta_A.$$

So this characteristic property is also satisfied.

In the terms of restriction mapping the computation above is represented as follows.

$$(p \circ \backslash\mathrm{decC}\{\mu\}\{\nu\})(c) = p\, (\backslash\mathrm{decC}\{\mu\}\{\nu\}(c)) = p\, (\mu\, c, \nu\, c) = \mu\, c.$$

We can see that in fact the restriction mapping is not evaluated. This is due to the pointwise character of the correspondent arrow (like an extensional mapping in the intensional logic). The other computations are analogous.

It is significant that the computations in terms of restriction mappings are in general more easy than in terms of functors. The computations in functors, however, have other advantage – they use only properties of cartesian product and do not use the definition of cartesian product in the category Set. So they can be carried out in the general case of cartesian category D.

For generalization of the restriction mappings to the arbitrary category D it is necessary to propose a generalization of the “element” notion in the category D. Possibilities of such generalization will be discussed below.

Exponential. The exponential in the category is the abstraction of the set of functions. The definition is given below.

The exponential of the objects a and b in the category C is such object d together with an arrow $ev : d \times a \to b$, that for every object c and arrow $f : c \times a \to b$ there is the unique arrow $h : c \to d$, for which
$f = ev \circ \langle h \circ p, q\rangle$.

The object d will be denoted as $a \to b$. The arrow h built by f, similar to the case of cartesian product, is the unique arrow with this property, so it can be considered as the function on f. This function will be denoted $\Lambda(f)$. The arrow $\Lambda(f)$ is named a currying of the arrow f.

In the new notations the properties of the exponential can be rewritten so.

$$ev \circ \langle \Lambda(f) \circ p, q \rangle = f,$$

$$\Lambda(ev \circ \langle h \circ p, q \rangle) = h,$$

where $h : c \to (a \to b)$.

The notion of exponential uses the cartesian product of objects. So the category must be cartesian for the existence of exponentials. The cartesian category where for every two objects there is an exponential is named cartesian closed category. Not every category is cartesian closed.

Example. Building an exponential in the category Set.

The example shows that the arrow ev is in some case the most general of all arrows of the same type. This observation can be made more precise. The arrow ev is a counit of adjunction given by the exponential construction. This determines its "the most general" character.

Exponential in the functor category: fail of the pointwise approach. We consider now the exponentials in the category $Set^{C^{op}}$. First of all we can notice that the straightforward approach to the building of the exponentials do not success. It is impossible to build exponentials in $Set^{C^{op}}$ pointwise.

To show that we present a counterexample. For building it we consider a construction of the exponential $\Phi \to \Phi$ for the functor Φ, where the functor Φ is defined above in the example. We suppose that the exponential can be built pointwise and denote it by ψ. Then $\psi_A = \Phi_A \to \Phi_A$ and $\psi_B = \Phi_B \to \Phi_B$. But $\Phi_A = \{0\}$, so $\psi_A = \{k_0\}$, where k_0 is a unique function from $\{0\}$ to $\{0\}$:

$$k_0 (0) = 0.$$

Further, $\Phi_B = \{0, 1\}$, so ψ_B has four elements (functions from Φ_B to Φ_B) given below:

$k_{00} (0) = 0$ & $k_{00} (1) = 0$
$k_{01} (0) = 0$ & $k_{01} (1) = 1$
$k_{10} (0) = 1$ & $k_{10} (1) = 0$
$k_{11} (0) = 1$ & $k_{11} (1) = 1$.

Now we have to define the restriction mapping for the functor ψ, that is the value of ψ on the arrow f. We can not define now, which values are acceptable, so we denote $k_0 \uparrow f$ as k_1.

The given construction shows that there exists the unique arrow $\psi: I \to \psi$, which components are defined as follows:

$$\psi_A (*) = k_0 \text{ and } \psi_B (*) = k_1.$$

We build now the evaluation arrow ev: $\psi \times \Phi \to \Phi$. The component of this arrow (natural transformation) on the object A can be defined uniquely: $ev_A (k_0, 0) = 0$$. For the component on the object B we evaluate the restriction mapping:

$(ev_A (k\_0, 0)) \uparrow f = ev_B (k\_0 \uparrow f, 0 \uparrow f) = ev_B (k\_1, 0) = 0.$

So we can see that for providing ev to be a natural transformation we must have

$ev_B (k_1, 0) = 0.$

Values of ev_B on the other components of its domain can be set up arbitrary, so we can have

$(k_1, 1) = 0$

or

$ev_B (k_1, 1) = 1$$.

So we have constructed the functor ψ and the arrow ev: $\psi \times \Phi \to \Phi$. They are not unique: in the construction of ψ and ev we have a considerable arbitrariness. Now we define a currying and show that for every definition of arbitrary elements it is impossible to provide the fulfillment of the characteristic properties of the exponential.

Let us consider arrows $I \times \Phi \to \Phi$. Considering the examples of natural transformations we have seen that there is two such arrows.

$f_{0A} (*, 0) = 0$ & $f_{0B} (*, 0) = 0$ & $f_{0B} (*, 1) = 0$

$f_{1A} (*, 0) = 0$ & $f_{1B} (*, 0) = 0$ & $f_{1B} (*, 1) = 1.$

Here we can see that currying is impossible: two arrows $I \times \Phi \to \Phi$ can not be carried by the single arrow $\psi: I \to \psi$. Let us see it in details.

Let the arrow ev be defined so that $ev_B (k_1, 1) = 0$. We consider the arrow f_1 and evaluate for it both parts of characteristic equation of currying mapping component for the object B on the element $(*, 1)$ of object $(I \times \Phi)_B$.

$(ev \circ <\Lambda (f_1) \circ p, q>)_B (*, 1) = (ev_B \circ <\Lambda (f_1)_B \circ p_B, q_B>) (*, 1) =$

$= ev_B (\Lambda (f_1)_B (*), 1) = ev_B (\psi_B (*), 1) = ev_B (k_1, 1) = 0,$

but $f_{1B} (*, 1) = 1$. So the arrow f_1 can not be carried. Similarly in the case of $ev_B (k\_1, 1) = 1$, the arrow f_0 can not be carried.

This contradiction shows that the supposed assumption was wrong, that is the pointwise construction of the exponential is impossible. The given proof is not unique, of course. We can propose a sketch of another proof, which is important for the method of the concordance of evaluation mappings.

We suppose that $(U \to V)_A = U_A \to V_A$ and $(U \to V)_B = U_B \to V_B$. Then $(U \to V)f : (U_A \to V_A) \to (U_B \to V_B)$, where $f: B \to A$. The arrow $(U \to V)f$ must be constructed by the method independent from the structure of the particular category C. For its construction we have the following data: arrows $Uf: U_A \to U_B$ and $Vf: V_A \to V_B$, and evaluation mappings $ev_A : (U_A \to V_A) \times U_A \to V_A$ and $ev_B : (U_B \to V_B) \times U_B \to V_B$$.

The arrow $(U \to V)f$ can be represented as the carrying of some arrow $(U_A \to V_A) \times U_B \to V_B$. This arrow in turn can be represented as the composition of $Vf: V_A \to V_B$ and some arrow $(U_A \to V_A) \times U_B \to V_A$. Such arrow could be obtained from evaluation mapping $ev_A : (U_A \to V_A) \times U_A \to$

V_A, if we have mapping U_B to U_A. But we have no such mapping, we have only the reverse mapping $Uf: U_A \rightarrow U_B$.

For making the given sketch strict it is enough to give for category C an example showing the absence of the mapping from U_B to U_A. Such example can be built in a manner similar to the first proof. So the conclusion is that there is no natural way to include the mapping of object A to $U_A \rightarrow V_A$ into a functor.

The given proof shows the reason of difficulties: for construction of the generalization of the function space it is not enough to consider the evaluation mapping $ev_A : (U_A \rightarrow V_A) \times U_A \rightarrow V_A$. For the building of the correct functional constructions it is necessary to consider the function values for such objects B that there exists the arrows $f: B \rightarrow A$, that is the arrows "on the later stages" of functors U and V. These later stages must be "built in" the functor we have to define. This way leads to the building of the correct exponential in the category $Set^{C^{op}}$. We describe now this construction in details.

Exponential in the functor category: function families. Despite of the impossibility of the approach to the construction of the exponential by analogy with the cartesian product, the category $Set^{C^{op}}$ is still cartesian closed. But successful exponential construction is a little more complicated.

Let U and V be two functors of $Set^{C^{op}}$. We define the functor $U \rightarrow V$ in the following way. The value of the object mapping of the functor on the object A is, according to D.Scott [9], the family of mappings $U_B \rightarrow V_B$, indexed by the arrows $f : B \rightarrow A$. That is, elements of $(U \rightarrow V)_A$ are mappings φ from the set of arrows $f : B \rightarrow A$ to the corresponding functions $U_B \rightarrow V_B$.

The values of the functions corresponding to the different arrows $f:B \rightarrow A$ must be coordinated. Namely, if we consider the arrow $g : C \rightarrow B$, then the composition $f \circ g$ is an arrow $C \rightarrow A$. This composition is also a possible index in the family φ and, if $\varphi_f : U_B \rightarrow V_B$ then $\varphi_{f \circ g} : U_C \rightarrow V_C$.
The condition of the coordination requires:

$$\varphi_{f \circ g} \circ Ug = Vg \circ \varphi_f.$$

For representing this condition now in terms of restriction mapping we apply both arrows to the element $b \in U_B$. We get

$$\varphi_{f \circ g} (b \uparrow g) = \varphi_f (b) \uparrow g.$$

The diagram below clarifies the situation.

$$\begin{array}{ccc} U_B & \rightarrow Ug \rightarrow & U_C \\ \downarrow\varphi_f & & \downarrow\varphi_{f \circ g} \\ V_B & \rightarrow Vg \rightarrow & V_C \end{array}$$

The required condition is a variant of naturality. Below we consider it in detail.

The arrow mapping for functor $U \rightarrow V$ must match each arrow $f: B \rightarrow A$ to the function transforming the mappings φ from arrows $C \rightarrow A$ to the functions $U_C \rightarrow V_C$ to the mappings ψ from arrows $C \rightarrow B$ to the functions $U_C \rightarrow V_C$. Symbolically:

$$(U \rightarrow V)_A = \{\varphi: (h : C \rightarrow A) \rightarrow (U_C \rightarrow V_C)\}$$
$$(U \rightarrow V)_B = \{\psi: (g : C \rightarrow B) \rightarrow (U_C \rightarrow V_C)\}$$
$$(U \rightarrow V)f : (U \rightarrow V)_A \rightarrow (U \rightarrow V)_B \text{ where } f: B \rightarrow A.$$

So the input for our task is the arrow $f: B \rightarrow A$ and the mapping φ which transforms each arrow

$h : C \to A$ to the function $U_C \to V_C$ (it is a set-theoretical function because the functors U and V have their values in the category Set). The output is the mapping ψ, which transforms each arrow $g: C \to B$ to the function $U_C \to V_C$. The natural way to define this mapping is to convert the arrow $g : C \to B$ to the arrow $h : C \to A$ and then apply the mapping φ. For the conversion g to h it is enough to assume $h = f \circ g$.

Now we formalize the construction. We denote

$$((U \to V)f) (\varphi) = \psi.$$

Then the value ψ on the arrow $g : C \to B$ can be defined as

$$\psi_g = \varphi_{f \circ g}.$$

This definition can be represented formally as

$$((U \to V)f (\varphi))_g = \varphi_{f \circ g}.$$

This formula defines a restriction mapping for the functor $U \to V$: $\varphi \uparrow f = \psi$ where $\varphi \in (U \to V)_A$. Here $\psi_g = \varphi_{f \circ g}$, that is

$$(\varphi \uparrow f)_g = \varphi_{f \circ g}.$$

This is the very construction that can not be generalized to the arbitrary category D. But there is rather wide class of categories allowing such generalization.

So we have defined the object and arrow mappings for the functor. Now we check that this is really a functor. We must check the composition preservation property

$$(U \to V)(f \circ g) = (U \to V)f \circ (U \to V)g$$

and the unit preservation property

$$(U \to V)(1_A) = 1_{(U \to V)A}.$$

We begin with the composition. Let $f: B \to A$ and $g: C \to B$. Then

$$(U \to V)(f \circ g): (U \to V)_A \to (U \to V)_C.$$

The elements of $(U \to V)_A$ are the families φ of mappings. For the comparison of these families we apply them to the arrow $h: D \to C$. Then

$$(U \to V)(f \circ g)(\varphi) = \psi, \qquad \psi_h = \varphi_{(f \circ g) \circ h}$$

and

$$((U \to V)f \circ (U \to V)g)(\varphi)_h = (U \to V)f ((U \to V)g)(\varphi)_h) =$$
$$= (U \to V)f (\varphi)_{g \circ h} = \varphi_{f \circ (g \circ h)}.$$

The property of the unit preservation can be checked in the similar way.

Now we define the arrow of the evaluation mapping $\varepsilon: (U \to V) \times U \to V$$. Then $\varepsilon_A: (U \to V)_A \times U_A \to V_A$ and

$$\varepsilon_A (\varphi, a) = \varphi_{1A}(a).$$

As usual we have to check the natuality of the given construction.

Now we consider currying. Let $\psi: U \times V \to W$, then $\psi_A: U_A \times V_A \to W_A$ and $\psi_B: U_B \times V_B \to W_B$. So $\Lambda\, \psi: U \to (V \to W)$ and $(\Lambda\, \psi)_A: U_A \to (V \to W)_A$. We define $(\Lambda\, \psi)_A\, a = \varphi$ where φ $f: V_B \to W_B$ where $f: B \to A$.
Let $b \in V_B$. Then $\varphi_f (b) = \psi_B (Ufa, b)$.

$((\Lambda\ \psi)_A)_f\ b = \psi_B\ (a \uparrow f, b)$

We check the functor character according to the diagram.

$$\begin{array}{ccc} U_A & \rightarrow Uf \rightarrow & U_B \\ \downarrow(\Lambda\ \psi)_A & & \downarrow(\Lambda\ \psi)_B \\ (V \rightarrow W)_A & \rightarrow (V \rightarrow W)f \rightarrow & (V \rightarrow W)_B \end{array}$$

We have

$(\Lambda\ \psi)_B \circ Uf = (V \rightarrow W)f \circ (\Lambda\ \psi)_A$

$((\Lambda\ \psi)((Uf)a))\_g\ (b) = \psi_C\ (Ufa \uparrow g, b) = \psi_C\ (a \uparrow (f \circ g), b)$

$((V \rightarrow W)f \circ (\Lambda\ \psi)_A\ a)\_g\ (b) = (\Lambda\ \psi)_A\ a)\_\{f \circ g\}\ (b) = \psi_C\ (a \uparrow (f \circ g), b)$

Example 1. We consider the exponential in the category $Set^{C^{op}}$ for $C = 2^{\rightarrow}$. This category has two objects A and B and the unique non-unit arrow f: B → A$. Let U and V be two functors from C^{op} to Set. For exponential U → V we have to define its object mapping on A and B and arrow mapping on f.
The value of the functor $(U \rightarrow V)_A$ is a set, elements of which are the families of mappings φ from the arrows B → A to the sets of functions $U_B \rightarrow V_B$. In our case there are two such arrows: $1_A : A \rightarrow A$ and $f : B \rightarrow A$. So every family of mappings φ contains two functions: $$\varphi_{1A}: U_A \rightarrow V_A$ and $\varphi_f: U_B \rightarrow V_B$. For these functions we have the condition of coordination, which has the form

$\varphi\_\{1_B \circ f\} \circ Uf = Vf \circ \varphi\_f,$

or in terms of restriction mapping

$\varphi\_f\ (a \uparrow f) = (\varphi\_f\ a) \uparrow f.$

We can see that the condition of coordination shows that φ_f is the natural transformation. This coincidence is the general case because the exponential is an abstraction of the set of arrows from U to V, and every such arrow is the natural transformation.

The value of the functor $(U \rightarrow V)_B$ is a set, elements of which are the families of mappings ψfrom the arrows C → B to the sets of functions $U_C \rightarrow V_C$$. In our case there are one such arrow: $1_B : B \rightarrow B$. So every family of mappings contains only one function $$\psi_{1B}: U_B \rightarrow V_B$. The condition of coordination in this case is trivial and does not specify any special restriction on ψ_{1B}.

The value $(U \rightarrow V)_f$ is a function transforming every φ from $(U \rightarrow V)_A$ to ψ from $(U \rightarrow V)_B$. The mapping ψ is completely determined by φ according to the formula $\psi_g = \varphi_{f \circ g}$. In our case $g = 1_B$, that is $\psi_{1B} = \varphi_f$. So $(U \rightarrow V)_f$ simply "selects" one function φ_f from the pair $\varphi = (\varphi_{1A}, \varphi_f)$.

Exponential in the functor category: natural transformations. We will show that the exponential in the category $Set^{C^{op}}$$ can be represented as a set of natural transformations between the functors of special kind.

For the given functors F, G: C → Set we define F → G as follows.

$(F \rightarrow G)(a)=Nat[F_a, G_a],$

where F_a and G_a are forgetful functors in the relative category C over a, defined above, and $Nat[F_a, G_a]$ is a set of all natural transformations from F_a to G_a.

For $k : a \to d$ we define $(F \to G)(k)$: $Nat[F_a, G_a] \to Nat[F_d, G_d]$. Let τ: $F_a \to G_a$ and τ': $F_d \to G_d$. Then $\tau'_f = \tau_{f \circ k}$.

5. Testing the constructions of a model

The methods of work with evolutionary concepts have been partially tested when developing the software for informational support to the process of BAT implementation. In particular, the methods of supporting the evolutionary trajectory of an object and its accounting when manipulating an object were used in the editor of conceptual descriptions of the subject domain.

The conceptual descriptions editor is oriented to the description of concepts of a fairly general structure. The concept description is accompanied by a description of the characteristic frames of the given concept. The arguments of characteristic frames may be either simple concepts or the conceptual operations results.

The editor provides for two metalanguages, one of which is intended to describe the structure of the edited concepts, and the other sets the concept representation for users. The evolutionary trajectory of the concept is a part of the concept structure. It is possible to select concepts according to the condition specified at the evolutionary trajectory, as well as to change the concept according to the rule that depends on its evolutionary trajectory.

Separate working mechanisms for evolutionary concepts have also been tested when developing the training and methodological complex on the legal basis for the BAT implementation. One of the information elements of the complex was the training program. Since the BAT implementation is a rapidly developing sector of law, the programme underwent evolution during the complex development. As separate stages of evolution caused methodological interest, the data structure was chosen to represent the evolving programme. A set of interface elements providing navigation through the evolutionary structure was also proposed.

The testing in the whole demonstrated a significant expansion of the capabilities of applied systems, arising from the introduction of evolutionary objects into them, and at the same time, the insufficient degree of methods development for describing and manipulating such objects, making it difficult to develop the means of visualization, navigation and transformation of evolutionary objects. As expected, a number of problems can be overcome by considering the basic categories with a richer internal structure.

6. Conclusion

The paper proposes the approach to constructing evolutionary concepts models as to an essential part of methods for solving the problem of supporting the semantic stability of information system objects. The approach is based on the integration of methods of category theory and the theory of applicative computing systems and provides for as follows:

- description of the basic case of the subject domain dynamics;
- description of the mechanisms of semantically stable operating mode of IS;
- model equational description of the mechanism for the information system returning to the mode of stable semantic functioning.

Elements of the proposed approach have been tested when developing tools for editing information object representations that provide for tracing the evolutionary trajectory of the edited object. The pointed tools were applied to develop the institutional basis for the implementation of the best available technologies in the Russian Federation; and they demonstrated the possibility of taking into account the evolution of objects in determining their behaviour, which defines the practical significance of the proposed approach.

References

[1] Atzeni, P. et al. 2013.The relational model is dead, SQL is dead, and I don’t feel so good

myself. SIGMOD Rec. 42, 1 (Jul. 2013), 64–68.
[2] Chernyshov A., Balandina A., Kostkina A., Klimov V. Intelligence Search Engine and Automatic Integration System for Web-Services and Cloud-Based Data Providers Based on Semantics, Procedia Computer Science, 10.1016/j.procs.2016.07.449
[3] Conceptual Modeling. Proceedings of 35th International Conference by ed. Comyn-Wattiau, I. et al, ER 2016, Gifu, Japan, November 14-17, 2016, Lecture Notes in Computer Science, Volume 9974 2016, 2016, Springer
[4] Gabriella Castro Barbosa Costa. Using data provenance to improve software process enactment, monitoring and analysis. In Proceedings of the 38th International Conference on Software Engineering Companion, ICSE ‘16, pages 875-878, New York, NY, USA, 2016. ACM.
[5] Larisa Ismailova. Criteria for computational thinking in information and computational technologies. *Life Science Journal*, 11(9s):415-420, November 2014.
[6] Montague, R., 1968, “Pragmatics”, in R. Klibansky (ed.), Contemporary Philosophy. A Survey, Florence: La Nuova Italia Editrice, 102–122. Reprinted in Thomason (ed.) 1974, pp. 95–118.
[7] Montague, R., 1970b, “Pragmatics and intensional logic”, Synthese, 22: 68–94. Reprinted in Thomason (ed.) 1974, pp. 119–147.
[8] Population Modeling Working Group. Population modeling by examples (wip). In Proceedings of the Symposium on Modeling and Simulation in Medicine, MSM ’15, pages 61–66, San Diego, CA, USA, 2015. Society for Computer Simulation International.
[9] Scott, D. "Advice in modal logic", in Philosophical Problems in Logic, ed. Karel Lambert. Reidel, 1970
[10] V. Wolfengagen et al. Migration of the Individuals Procedia Computer Science, Volume 88, 2016, pages 359–364, http://dx.doi.org/10.1016/j.procs.2016.07.449
[11] Wolfengagen,V.E., Ismailova,L.Y., Sergey V. Kosikov Computational Model of the Tangled Web Procedia Computer Science, 10.1016/j.procs.2016.07.440
[12] Wolfengagen,V.E. Ismailova,L.Y. et al Evolutionary Domains for Varying Individuals Procedia Computer Science, 10.1016/j.procs.2016.07.447