Journal of Physics: Conference Series

PAPER • OPEN ACCESS

# Formalization of system-object method of knowledge representation by calculation of systems as functional objects

# Formalization of system-object method of knowledge representation by calculation of systems as functional objects

**A G Zhikharev [1], S I Matorin [2], O A Tinyakov [3], N V Shcherbinina [2] and L V Migal [2]**

[1] Belgorod State Technological University named after V.G. Shukhov, 46, Kostyukova St., Belgorod, 308012, Russia
[2] Belgorod State University, 85, Pobedy St., Belgorod, 308015, Russia
[3] Kursk State University, 5, Kirov St., 305000, Russia

E-mail: zhikharev@bsu.edu.ru

**Abstract**. The paper considers some elements of the calculation systems as functional objects. The formal foundations of calculus of systems proposed by the authors were preceded by research on the development of a mathematical apparatus that allows formalizing the procedures for developing system-object simulation models of processes and systems. In the work, the previously developed formal apparatus is supplemented by the context operator, and some theorems related to the structural and functional characteristics of the modeled objects are formulated and proved. In particular, using the context operator, the statement is proved that the connection of a nodal object with the external environment generates the same connections of other nodal objects whose functions are realized due to the functions of the first nodal objects. It is shown that this statement is true for both incoming and outgoing connections with respect to a nodal object. Intrasystem connections are considered that are also capable of generating connections for their contextual node objects. In addition, the paper proposes a new formal record of the function of a nodal object for a situation when it is implemented due to the functions of other nodal objects. The proposed formalisms are considered on the example of a system-object model of an abstract system. In the future, based on the calculus of systems, optimization algorithms for system-object simulation models will be formulated according to various optimization criteria.

## 1. Introduction

To solve the problems of informational and analytical support of the activities of organizational systems, the authors developed a system-object method of knowledge representation (SOMKR), which has a number of significant advantages. Its advantages include the possibility of a graphical representation of knowledge, the possibility of formalizing these graphical representations, and the ability to transform a graphical representation into a simulation model. The formalization of the MPSE leads to the need for more formalization of the system-object approach “Node-Function-Object” (NFO-method), on which this method of knowledge representation is based.

Here are some research results in this direction, obtained with the use of the algebraic apparatus for calculating the objects of Abadi-Kardeli [1].

## 2. Materials and methods
Let us consider some elements of the calculus of systems as functional objects using the example of an abstract system, the graphic-analytical representation of which is shown in Picture 1. It should be noted that the notation of input and output stream objects, as well as other elements of the system-object model, are used in the work in accordance with previously developed formal foundations calculus systems described in [2,3].

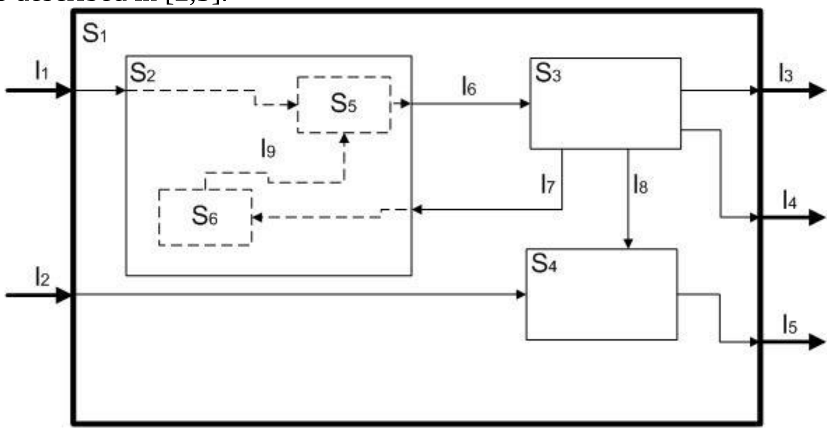


**Figure 1.** System-object model of abstract system.

In accordance with the calculus of systems as functional objects [3], we consider the formal representation of the model in the following form:

$$M=(L,S,C), \quad (1)$$

where:

L - is the set of stream objects of the model (hierarchy of system connections) [3]. Each stream object is represented by a set of named fields containing the values of the stream object [3].

For the considered example, the set has the form:

$$L=\{l_1,l_2,l_3,l_4,l_5,l_6,l_7,l_8,l_9\} \quad (2)$$

S – is the set of nodal objects of the model (subsystems) [7]. A nodal object in terms of calculation systems as functional objects:

$$s = [(L?, L!); f(L?)L!; (O?, O!, Of)] \quad (3)$$

For the considered example, the set has the following form:

$$S=\{s_1,s_2,s_3,s_4,s_5,s_6\} \quad (4)$$

C is the set of connections of the system, and the element of the set consists of three components:

$$c=\{s_{out},s_{in},l\}: s_{out}\epsilon S,\ s_{in}\epsilon S,\ l\epsilon L \quad (5)$$

Thus, an element of the set C is a connection of two subsystems. However, any system-object model has modeling boundaries, i.e. context. As can be seen from Picture 1, the node object $s_1$ is actually not connected to any other node object. To account for such situations, it is proposed to introduce a permanent nodal object scontext, which is a black box from the point of view of the model and representing the external environment of the simulated system. Then the set of nodal objects of the model will take the following form:

$$S=\{s_1,s_2,s_3,s_4,s_5,s_6,s_{context}\} \quad (6)$$

In view of the foregoing, the set of relationships of the considered model can be written in the following abbreviated form:

$$C=\{(s_{context},s_5,l_1),\ (s_{context},s_4,l_2),\ (s_3,\ s_{context},l_3),\ (s_3,\ s_{context},l_4),\ (s_4,\ s_{context},l_5),\ (s_5,s_3,l_6),\ (s_3,s_6,l_7),\ (s_3,s_4,l_8),\ (s_6,s_5,l_9)\ \} \quad (7)$$

Thus, the sets (2,6,7) fully take into account the structural characteristics of the system-object model under consideration. However, it can be noted that in the model shown in Figure 1, the functions of the nodal objects $s_1$ and $s_2$ are realized by the nodal objects of the lower levels. Let's consider in more detail these nodal objects: $s_1$ = [(L?, L!); f(L?)L!; (O?, O!, Of)].

The function of a nodal object, in fact, is realized due to the functions of nodal objects of the lower tier. For the formal designation of "nesting" functions we will use the following format:

$$s_1.f = \begin{cases} s_2.f \\ s_3.f \\ s_4.f \end{cases} \quad (8)$$

Moreover, the function of the node $s_2$ is also realized due to the functions of the nodal objects of the lower tier

$$s_1.f = \begin{cases} s_2.f = \begin{cases} s_5.f \\ s_6.f \end{cases} \\ s_3.f \\ s_4.f \end{cases} \quad (9)$$

Let's consider the functions of the presented nodal objects taking into account the stream objects-parameters of the functions and stream-objects - the results of the functions, for example, for the node object $s_1$ the function has the form:

$$s_1.f = f(l_1, l_2)l_3, l_4, l_5 \quad (10)$$

As can be seen from formula 10, one and two stream objects act as stream objects-parameters of the function of the node object $s_1$, and objects three, four, and five act as stream objects-results.

Then the complete record of the function of the nodal object $s_1$ can be represented as follows:

$$s_1.f(l_1, l_2)l_3, l_4, l_5 = \begin{cases} s_2.f(l_1, l_7)l_6 = \begin{cases} s_5.f(l_1, l_9)l_6 \\ s_6.f(l_7)l_9 \end{cases} \\ s_3.f(l_6)l_3, l_4, l_7, l_8 \\ s_4.f(l_2, l_8)l_5 \end{cases} \quad (11)$$

## 3. Results and discussion

Let's consider the node $s_5$ in more detail, in terms of calculus of systems, this node object is presented in the form:

$$s_5 = [(L?=\{l_1, l_9\}, L!=\{l_6\}); f(L?)L!; (O?, O!, Of)] \quad (12)$$

To determine the context of a nodal object, we introduce into the calculus of systems a special operator that defines the context of the node, in other words, the connection of the system with the outside world. One of the nodes of the model will act as a parameter of such an operator. Further, such an operator will be denoted as «kontext(s)», for example, for the node $s_1$, the functional of this operator can be represented in the format of the following expression:

$$context(s_1) = \begin{cases} [s_{context}, s_1, l_1] \\ [s_{context}, s_1, l_2] \\ [s_1, s_{context}, l_3] \\ [s_1, s_{context}, l_4] \\ [s_1, s_{context}, l_5] \end{cases} \quad (13)$$

That is, the context operator returns a list of relations between the node object and external node objects, in general, the operation of the context operator can be described as the following expression:

$$context(s) = \{[s_{out}, s_{in}, l] \in C | s_{out} \in s.U \vee s_{in} \in s.U\} \quad (14)$$

Applying the context operator to node $s_5$, we obtain the following expression:

$$context(s_5) = \begin{cases} [s_{context}, s_5, l_1] \\ [s_6, s_5, l_9] \\ [s_6, s_3, l_6] \end{cases} \quad (15)$$

It can be seen from formula 15 that the stream object $l_1$ is an incoming connection in relation to node $s_5$, while the source of the stream object $l_1$ is the external environment $s_{context}$. It's not difficult to notice that the right statement will be:

*Statement 1.*

Any connection with = {$s_{context}$, $s_{in}$, l} that makes up the context of the $s_{in}$ node (incoming connection in relation to the sin node) is also an incoming connection with the node s * if the function of the node s * .f is implemented including node functions $s_{in}$.f.

That is, if there is such a connection that is incoming in relation to a certain node and the source of communication is the external environment $s_{context}$, then this connection will also be incoming for nodes whose functions are implemented using the function of the first node directly or indirectly (through the functions of third node objects )

Formally, this statement can be written as follows:

$$\forall c = \{s_{context}, s_{in}, l\} | c \in context(s_{in}) \rightarrow c \in context(s^* | s^*.f = \begin{cases} s_{in}.f \\ \dots \end{cases}) \quad (16)$$

Evidence. To prove relation 16, we consider an arbitrary node s such that:

context(s)={$s_{context}$,s,l}, (17)

and also the node s * such that:

$$s^*.f = \begin{cases} s.f \\ \dots \end{cases} \quad (18)$$

By the terms we know that s.f=f(l), then, the right formula will be:

$$s^*.f = \begin{cases} s.f(l) \\ \dots \end{cases} \rightarrow s^*.f(l) = \begin{cases} s.f(l) \\ \dots \end{cases} \rightarrow \{s_{context}, s, l\} \in context(s^*) \quad (19)$$

From formula 19 we see, that the existence of the connection {$s_{context}$,s,l} guarantees the existence of the connection {$s_{context}$,s*,l} when the condition 18 fulfilled. By this means, returning to example of an abstract system, which its graphoanalytic representation is represented on picture 1, we can conclude, that the connection {$s_{context}$,$s_5$,$l_1$} guarantees the existence of other two incoming connections of nodal high level objects {$s_{context}$,$s_1$,$l_1$} and {$s_{context}$,$s_2$,$l_1$}.

It should be noted, that the formula 16 also can be right for outcoming connections of the system, which have a successor in the form of a special nodal object $s_{context}$, i.e.:

$$\forall c = \{s_{out}, s_{context}, l\} | c \in context(s_{out}) \rightarrow c \in context(s^* | s^*.f = \begin{cases} s_{out}.f \\ \dots \end{cases}) \quad (20)$$

Evidence. To prove relation 20, we consider an arbitrary node s such that:

context(s)={s,scontext,l}, (21)

and also the node s * such that:

$$s^*.f = \begin{cases} s.f \\ \dots \end{cases} \quad (22)$$

By hypothesis, we know that s.f = f (...) l, that is, the result of the operation of node s is a stream object l, then the formula:

$$s^*.f = \begin{cases} s.f(\dots)l \\ \dots \end{cases} \rightarrow s^*.f(\dots)l = \begin{cases} s.f(\dots)l \\ \dots \end{cases} \rightarrow \{s, s_{context}, l\} \in context(s^*) \quad (23)$$

It can be seen from formula 23 that the existence of the connection {s, scontext, l}, as in the first case in the incoming connection, guarantees the existence of the connection {s *, $s_{context}$, l} under condition 22. Considering the example of the system model shown in Picture 1, we can conclude that a connection, for example, {$s_4$, $s_{context}$, $l_5$} guarantees the presence of a connection of a top-level node object {$s_1$, $s_{context}$, $l_5$}.

By analogy with connections in which the external environment acts as a source or receiver, we consider the internal connections of the system. Those that connect two nodal objects located at different levels of the hierarchy. For example, the $l_7$ connection (see Picture 1) connects two nodal objects $s_3$ and $s_6$. Obviously, the following holds.

Statement 2.

Any connection c={$s_{out}$,$s_{in}$,l}, constituting the context of the node sin (incoming connection with respect to the node sin), is also an incoming connection with respect to the node s * if the function of the node s * .f is implemented including using the function of the node sin.f and has parameters of the above functions place the stream object l, i.e. $s_{in}$.f = f (l, ...) and s * .f = f (l, ...). Formally, this statement can be represented as:

$$\forall c = \{s_{out}, s_{in}, l\} | c \in context(s_{in}) \rightarrow c \in context(s^* | s^*.f(l, \dots) = \begin{cases} s_{in}.f(l, \dots) \\ \dots \end{cases}) \quad (24)$$

As can be seen from the formula, the direction of communication in this case does not play any role. Thus, the connection ($s_3$, $s_6$, $l_7$) of the system, the model of which is shown in Picture 1, guarantees the presence of connection ($s_3$, $s_2$, $l_7$) since the function of the node $s_2$ is realized, including due to the function of the node $s_6$.

## 4. Conclusion

In the work, the possibility of formalizing the system-object method "Node-Function-Object" and the system-object-based method for representing organizational knowledge based on it is investigated. The expediency of using for calculating systems as functional objects of calculating objects of Abadi-Kardeli [1] is shown.

In terms of the proposed calculus, the aggregation of systems as elements of the "Node-Function-Object" is ensured and their decomposition is facilitated.

The results show the feasibility of constructing a formalized theory of system-object analysis and modeling by expanding and improving the calculus of special objects as systems in the framework of the NFO method.

## 5. Acknowledgments

Researchers are supported by grants of the Russian Foundation for Basic Research: project № 18-07-00355, 19-07-00290 and 19-07-00111.

## References

[1] Abadi Martin and Luca Cardelli 1996 *A Theory of Objects* Springer-Verlag 397
[2] Matorin S I, Zhikharev A G 2018 *Calculation of the function objects as the systems formal theory basis* Advances in Intelligent Systems and Computing **679** 182-191
[3] Matorin S I, Zhikharev A G 2019 *Accounting for system-wide regularities in the system-object modeling of organizational knowledge* Scientific and Technical Information Processing **46(6)** 1-9