



2020 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: Eleventh Annual Meeting of the BICA Society

Prototype Mechanisms for Supporting the Network of Links to Parameterized Data Objects

Larisa Ismailova^a, Sergey Kosikov^b, Viacheslav Wolfengagen^{a,*}

^aNational Research Nuclear University “Moscow Engineering Physics Institute”, Kashirskoye Shosse, 31, Moscow 115409, Russia

^bNAO “JurInfoR”, M. Pirogovskaya str., 5, Moscow 119435, Russia

Abstract

The paper considers methods to organize a network of links to parameterized data objects. A data object is an object identified by standard methods, including the network identifiers URI, URL and URN. The data objects can be either atomic or have a complex structure, including links to other objects, and this ensures the linking of objects into a network, which can be considered as an information graph. It turns out to be possible to formulate a number of criteria for supporting the integrity of such a structure; the paper proposes language tools to formulate such criteria.

An essential feature of this network is the dependence of the objects' information content on a number of parameters, first of all, on time. Depending on the parameter, the information content of the object can change both with the preservation of the integrity of the structure, and with its damage. The paper considers methods for a parameterized description of objects while maintaining integrity based on the formalism of category theory. The formalization used allows defining two relations between objects and their names (identifiers): access and links. The paper studies the basic properties of such relations and reveals the differences between them.

Category theoretical means of describing objects require appropriate support tools for practical application. The paper proposes a configuration of support tools for a network of links to parameterized data objects based on a combination of conceptual modeling tools and applicative computing tools. The proposed tools provide basic methods of access to the network of information graphs. An essential feature of the proposed tools is the ability to determine the subjective way of considering the information graph (based on the category of correlations), as well as harmonizing an objective and subjective way of considering. The proposed tools were tested in the development of applied information systems in the field of implementation of the best available technologies (BAT).

© 2021 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 2020 Annual International Conference on Brain-Inspired Cognitive Architectures for Artificial Intelligence: Eleventh Annual Meeting of the BICA Society

Keywords: data objects, object identification, conceptual constructions, informational graph, semantics, category theory, model parametrization

* Corresponding author. Tel.: +7 (495) 778-87-26.

E-mail address: lyu.ismailova@gmail.com, kosikov.s.v@gmail.com, jir.vew@gmail.com

1. Introduction

The explosive development of Web technologies forwards in the agenda of the day the development of theoretically grounded mechanisms for the presentation and processing of information in Web systems [18, 16, 17]. The development of the concept of an object keeps staying one of the most essential tasks in the application of computer science methods in the framework of Web technologies. The use of the concept of an object stands in one-to-one correspondence with the cognitive concepts of modeling a subject area as it is assumed that a person operates with objects when describing subject areas [19, 6, 7, 13].

The concept of an object provides the possibility of joint consideration of several objects. At the same time the objects can be both atomic, that is, indivisible within the framework of the problem being solved, and composite, composed of simpler objects [15, 14]. The possibility of considering objects as atomic or composite is relative and in practice depends on the problem being solved. So, for example, the text displayed in the browser can be considered as an atomic object in the system of the site content management, or as an object composed of units of lower order – words, when solving the problem of placing this text with alignment.

A widely used way of organizing a set of objects is to establish links from one object to another. The resulting set of interconnected objects can be considered as a system of objects [8]. For example, such a system can be formed by a set of resources organized in the form of a site. To consider a system of objects it is convenient to take a semantic approach, in which it is assumed that objects are described in a certain language, and interaction with the system of objects is a procedure for calculating (obtaining values) expressions of this language. For example, displaying objects in a browser can be considered as a special case of such a calculation.

The procedure of obtaining values can be partially determined. In other words, the situation is possible when some expressions have no meaning. Such expressions may be considered as meaningless. The absence of meaningless expressions when defining a system of objects is hereinafter called the semantic integrity of the system, and the preservation of this property when working with the system is the preservation of semantic integrity. As a rule, it is possible to express requirements to the system that ensure the preservation of semantic integrity in some language, which is an extension of the object definition language.

When operating the objects, the first arising problem is their identification. The complexity of the identification problem in the general case is determined by the fact that objects can have a limited lifetime, that is, they can appear and disappear in a time less than the time of using the corresponding information system. An additional complication is introduced by the fact that objects can merge with each other or split into parts, which in turn are objects.

Practically the Web systems use URI (Uniform Resource Identifier), URL (Uniform Resource Locator), and URN (Uniform Resource Name) mechanisms to identify objects. It is suggested that these mechanisms are able to uniquely identify an object, and this, generally speaking, is not achieved in a natural language. However, in practice, these mechanisms can indicate an inaccessible object, which leads to a violation of the semantic integrity of the object system.

This paper aims to study the possibilities of preserving the semantic integrity when interacting with a system of mutually referencing objects. The paper consists of 6 chapters. Chapter 2 makes brief review of the approaches to the problem of preserving the semantic integrity. Chapter 3 contains the statement of the problem of maintaining a network of links to parameterized data objects. Chapter 4 contains a description of the basic means of determining parameterized data objects based on category theory. Chapter 5 describes the prototype support tools. Chapter 6 briefly summarizes the results.

2. Related work

The problem of semantically correct support of a network of references to parameterized data objects was previously considered in various ways. One of the earliest statements of this problem arose in practical programming due to the need to use various methods of memory management. As soon as programming languages that allow link management emerged, two related problems were identified: the problem of hanging links and the problem of garbage collection.

The problem with hanging links is that the phases of the object's lifecycle do not coincide with the phases of the link's lifecycle. This may result in a situation where the object is actually deleted from memory, but the reference to

it remains. Working with such objects leads to system instability. In modern programming languages (like C++), the problem is solved by using "smart pointers", to which information about the phase of the object's life cycle is attached. This solution makes it possible to synchronize life cycles, but requires additional overhead for its maintenance, which reduces the effectiveness of the resulting programs.

The problem with garbage collection is that external references to existing objects can be destroyed. In this case, individual objects or their collections become inaccessible from the external environment, which leads to both inefficient memory use and the possibility of semantic violations. The problem is solved by creating "garbage collectors" that identify areas of memory that are no longer accessible and make them available for use again. This approach ensures re-use of memory for inaccessible objects, but leads to additional costs in the running time of the program, which can be critical, for example, in real-time systems [4].

A semantically more interesting approach to maintaining the correctness of a network of object references arises in relational database theory when considering the problem of maintaining database integrity [5, 3]. In a relational database, objects are represented by identifiers, and the identifier can be defined in one (primary) relationship, and used in one or more other (derived) relationships. Removing an identifier from the main relationship generally does not cause it to be removed from derived relationships, which leads to the possibility of meaningless identifiers.

The problem is solved by defining the interaction with the database in terms of transactions. Several basic operations of working with relationships can be collected in a single transaction, which in principle allows you to consistently manipulate identifiers and ensure the semantic correctness of the database. However, there is currently no General approach to building semantically secure transactions. The database administrator is forced to re-design the transaction system for each individual case, which leads to the possibility of difficult-to-diagnose errors.

The problem of correct support of the link network in Web-oriented systems is particularly acute. It is assumed that the URI, URL, and URN mechanisms proposed for this purpose provide reliable access to the identified item. However, the URI can also refer to an inaccessible thing. There are two different relationships between names and things: access and reference. A link is inherently ambiguous in any language. Thus, any attempts by the Web architecture to make a link on the Internet completely unambiguous fail.

It is also necessary to note approaches to solving the problem from logical positions. Thus, S. Kripke [10] offers a logical approach that provides unambiguous identification, but the set of contexts in which such identification is possible is quite narrow [11, 12]. There are also approaches to identification based on the philosophy of knowledge [2]. However, in Web systems, links use descriptions, and therefore there is no unambiguous resolution of links. The Internet requires not only redirection, but also a uniform and logically consistent procedure for matching descriptions with URIs, which is currently not available [17].

3. Statement of problem

The objects, composing the description of practically interesting subject areas, usually contain links to each other. This makes it possible to consider the system of objects composing the description of the subject area as an information graph, the vertices of which correspond to the objects, and the arcs correspond to the connections of objects. The description of the subject area in the form of an information graph makes it possible to abstract from the details of the description and consider the models, obtained in different ways, from a single point of view.

The method of obtaining a model depends on the goals of the corresponding information system and the means of its development. For example, within the framework of the object-oriented programming paradigm, the models can be obtained in the form of systems of objects stored in RAM. Developing the database systems provides for obtaining the models for specific databases. Web programming provides for obtaining models in the form of sites or portals containing information about the subject area.

The use of an information graph can be interpreted as calculating the value (attribution) of an information graph or a part of it. Such a calculation is generally performed relatively to some context, where the context is understood as a data structure containing the values of subexpressions nested in the information graph, necessary to obtain the value of a given fragment of the information graph. Further on the context will also be referred to as a computing environment, which needs to be distinguished from a supporting programming environment that provides the contexts representations.

Inside of the information graph, two different relationships are supported between object references and the objects themselves – access and link. In this paper a link is understood as a relation ensuring the identification of an object, as well as the possibility of transferring an indication of an object to another context. The transfer of the indication does not actually affect the object to which the indication is made.

The links in different models can be represented in different ways. In database systems (relational), links can be implemented both in the form of key attribute values for objects stored in a certain relation, and in the form of higher-level reference objects – views and pointcuts of database. In object-oriented programming systems the ability to use links is evident as the ability to define polymorphic data types, which make it possible to process without specifying the structure of nested types. In Web systems the links are specified as URL that enable the transfer.

A feature of a link to an object is the possibility of a situation in which the object being referenced does not actually exist or is in a state that does not allow its calculation. This situation in general case leads to the loss of the semantic integrity of the information graph. For example, in Web systems this situation corresponds to a “dead link” in which the information integrity of the network is violated.

This paper interprets an access relation as a relation that provides an indication of an object, as well as the possibility of using an object in ways that do not violate the semantic integrity of the object. An access relation deals with the actual information content of an object. Thus, for example, in database systems an access relation ensures performing the integrity constraints, in particular, the executing the triggers that restore integrity. In programming languages the access relation is provided, in particular, by encapsulation, which makes it possible to process an object by methods that do not violate its semantic integrity. In Web systems, this relationship makes it possible to connect specific processing methods with individual objects (for example, listening to an audio file, watching a video file, etc.).

Like a reference relationship, access relationship to the object can also be violated. The violation is an attempt to process an object by such operation that is not applicable to the specified object. For example, a file converter from one format to another may not accept files of a certain format. Then an attempt to convert them will be an access error.

The principal complexity of the object representation of the practical subject areas description lies in the dependence of data objects on a number of parameters. Time is one of the most important parameters. As time goes the objects can be lost (cease to exist) or change their behavior in an undesirable way.

Let us consider a (hypothetical) example. Let the network have a file of a certain type and a programme that allows working with it. Suppose that the company-producer at some later moment makes the use of the programme paid. Then, when the existing information graph is saved, its calculation is significantly modified, since now it must include a payment procedure, an access authorization check, etc.

Other significant parameters may encounter the type of data presentation, including, for example, the format of the files used, the aspect of considering the subject area, the class of users to whom the fragment of the information graph is oriented, etc. The objects used appear at timely undefined moments, exist, inhabiting the global environment, and then may cease to exist, becoming inaccessible for search mechanisms. The problem of supporting the semantic integrity of the information graph in such conditions has been studied rather poorly and continues to be relevant.

The study of potential structural features of an information graph and methods of its parameterization allows to set up the problem of supporting a network of links to parameterized data objects as a task to develop the following:

- theoretical methods of parameterized data objects description with possibility to insert specific features of the objects' existence in the network, such as encapsulation and polymorphism;
- a set of tools that supports the representation of a network of links, taking into account the possibilities of parameterization.

To develop a link network support system, the possibility appears to be essential to create a supportive computing environment that provides link manipulation in a sufficient variety of contexts. One of the desirable characteristics of the environment is referential transparency, which is understood as the possibility to fix a fragment of a conceptual graph together with the environment providing its computation. This property can be achieved, in particular, in an environment of applicative type.

4. Categorical model

The theoretical methods of parameterized data objects description may be developed based on the category theory. The given paper proposes to use the internal theory of categories. Category C is taken as the initial one permitting the pullback. Then inside of this category the internal category is constructed and with the help of its tools the parameterized objects are described.

The construction is performed as follows [1, 9]. In category C , select object c_0 (object of objects) and c_1 (object of arrows). Morphisms are selected: $d_0 : c_1 \rightarrow c_0$ (morphism of the beginning of the arrow); $d_1 : c_1 \rightarrow c_0$ (arrow end morphism); $i : c_0 \rightarrow c_1$ (unit arrow morphism); $m : c_1 \times_{c_0} c_1 \rightarrow c_1$ (composition morphism). The selected morphisms must satisfy the following axioms:

$$\begin{aligned}d_0 \circ i &= d_1 \circ i = 1 \\d_0 \circ m &= d_0 \circ \pi_1 \\d_1 \circ m &= d_1 \circ \pi_2 \\m \circ (1 \times m) &= m \circ (m \times 1) \\m \circ (1 \circ i) &= m \circ (i \circ 1) = 1\end{aligned}$$

Here, π_1 and π_2 are projections.

The specified data defines an internal category. The internal category is used to construct a domain-oriented category of parameterized objects within a given category C (in the simplest case, the category of sets).

The internal functor $f : C \rightarrow D$ consists of a pair of morphisms $f_0 : C_0 \rightarrow D_0, f_1 : C_1 \rightarrow D_1$ commuting with d_0, d_1, i and m . Note that internal categories and internal functors form a category. Internal functors are used to represent changes to objects, in particular, to describe the transition from the object naming system to the object information graph. The correctness of such a transition can be expressed by the requirement that the mapping of the corresponding internal categories is an (internal) functor.

The internal diagram F on C is given as follows. Set: the object $\gamma_0 : F_0 \rightarrow C_0$, a morphism $e : F_1 = F_0 \times_{C_0} C_1 \rightarrow F_0$, satisfying the following axioms:

$$\begin{aligned}\gamma_0 \circ e &= d_1 \circ \pi_2, \\e \circ (1 \circ i) &= 1, \\e \circ (e \circ 1) &= e \circ (e \circ m)\end{aligned}$$

Internal diagrams allow us to consider not only functors from one internal category to another, but also functors to the C category itself.

The use of the category theory in the way indicated above ensures the following characteristics of the category model:

- possibility to select objects in accordance with the types of their dependencies on parameters;
- possibility to aggregate objects in accordance with the selected interactions based on ordered pairs;
- possibility to construct the means of related manipulation of objects based on pullbacks (fibered products);
- possibility of considering functional objects as objects of the first class based on exponentials;
- possibility to build hierarchies of objects based on mono arrows and their classification;
- possibility to develop support tools based on a categorical abstract machine.

5. Development tools

Prototypical tools for maintaining a network of links to parameterized data objects represent a prototypical system of knowledge representation, which allows supporting a parameter-dependent network of links. The model foresaw

for both the calculation of fragments of the information graph, taking into account the given parameters, and the generation of expressions representing possible fragments of the information graph, taking into account the imposed restrictions.

While making the computations, it was possible to calculate various fragments of the information graph with different values of the parameter. The implemented computational capabilities ensure modeling of imposed constraints for the considered subject area. The restrictions include quantitative constraints, constant substitution, and combined constraints as well. It is possible to set nested constraints.

The objects of subject area description are stored in a file. They are implemented both as the XML format representation and the internal system format representation that allows a more compact description. When loading, both syntactic and semantic integrity of information is checked. The experiments were carried out on the modular description of objects, providing the representation of common objects of various descriptions in special files.

The performed computations include the computation of the attributions of simple objects, as well as the recursive computation of the attributions of complex objects based on simple ones. The supported context contains the values of the syntax elements of the object description, as well as the values of the required parameters. In the whole, the computational component can be considered as a specialized categorical abstract machine for parameterized computations.

The user interface with the system ensures the study of the network of objects describing the subject area, which is considered as a directed acyclic graph. During visualization, the selection and processing of common subgraphs of the graph is provided. The editing of values assigned to simple objects of the system is also performed.

The supporting system is developed in the Java language. The test sample for developing and testing the domain-specific engines contains about 200 concepts.

The tools were tested while developing the applied models for representation of knowledge in the field of supporting the implementation of BAT. The constructed information graph represented a description of the subject area of training students in the legal framework for the implementation of BAT. There were also demonstrated the models of training materials and models of system integrity check.

6. Conclusion

The paper proposes an approach to the construction of mechanisms to support a network of links to parameterized data objects. Two relationships are defined between an object and the way it is identified: relationship of a link and an access. The similarities and differences of these relations, as well as options for their representation in practical information systems, have been studied.

The paper proposes an approach to describing the parametrization of a network of links based on category theory. The development of the basic category and the internal category by means of the basic one is considered, it, in aggregate, provides a description of parameterized data objects. The proposed approach provides as follows:

- construction of parameterized data objects of various types, managed by the choice of representing objects of the base category;
- providing tools for manipulating objects with support for integrity constraints by introducing structural constructions of the basic category;
- possibility to develop support tools by constructing a specialized categorical abstract machine.

The constructions of the proposed approach were put as the basis for the development of a prototype link network support tool. The prototype was tested when developing applied information systems in the field of BAT implementation. The testing has demonstrated the principal applicability of the proposed constructions.

Acknowledgement

The paper is supported by the grants 20-07-00149-a, 19-07-00326-a, 19-07-00420-a of the Russian Foundation for Basic Research.

References

- [1] Awodey, S., 2011. *Foundational Theories of Classical and Constructive Mathematics*. Springer Netherlands, Dordrecht. chapter From Sets to Types, to Categories, to Sets. pp. 113–125. URL: https://doi.org/10.1007/978-94-007-0431-2_5.
- [2] Barendregt, H., 1997. The impact of the lambda calculus in logic and computer science. *Bulletin of Symbolic Logic* 3, 181–215. URL: <https://doi.org/10.2307/421013>, doi:10.2307/421013.
- [3] Brodie, M.L. (Ed.), 2019a. *Making Databases Work: The Pragmatic Wisdom of Michael Stonebraker*. Association for Computing Machinery and Morgan & Claypool, New York, NY, USA.
- [4] Brodie, M.L., 2019b. *On Developing Data Science*. Springer International Publishing, Cham. pp. 131–160. URL: https://doi.org/10.1007/978-3-030-11821-1_9, doi:10.1007/978-3-030-11821-1_9.
- [5] Brodie, M.L., 2019c. *What Is Data Science?*. Springer International Publishing, Cham. pp. 101–130. URL: https://doi.org/10.1007/978-3-030-11821-1_8, doi:10.1007/978-3-030-11821-1_8.
- [6] Ismailova, L., Wolfengagen, V., Kosikov, S., Dohrn, M., 2020a. Capturing push-processing using enriched semantic mesh equipped with functionals-and-hops model. *Procedia Computer Science* 169, 590 – 596. URL: <http://www.sciencedirect.com/science/article/pii/S1877050920303288>, doi:<https://doi.org/10.1016/j.procs.2020.02.205>. postproceedings of the 10th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2019 (Tenth Annual Meeting of the BICA Society), held August 15-19, 2019 in Seattle, Washington, USA.
- [7] Ismailova, L., Wolfengagen, V., Maslov, M., Dohrn, J., 2020b. Semantic models to indicate post-truth with fake news channels. *Procedia Computer Science* 169, 297 – 303. URL: <http://www.sciencedirect.com/science/article/pii/S1877050920303057>, doi:<https://doi.org/10.1016/j.procs.2020.02.182>. postproceedings of the 10th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2019 (Tenth Annual Meeting of the BICA Society), held August 15-19, 2019 in Seattle, Washington, USA.
- [8] Kosikov, S., Ismailova, L., Wolfengagen, V., Marenkov, A., Ermak, M., Slietsov, I., Zaytsev, V., Shedko, A., 2020. Indexical structures to enable knowledge mining tasks. *Procedia Computer Science* 169, 284 – 290. URL: <http://www.sciencedirect.com/science/article/pii/S1877050920303033>, doi:<https://doi.org/10.1016/j.procs.2020.02.180>. postproceedings of the 10th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2019 (Tenth Annual Meeting of the BICA Society), held August 15-19, 2019 in Seattle, Washington, USA.
- [9] Mac Lane, S., 2001. *Logic, Meaning and Computation: Essays in Memory of Alonzo Church*. Springer Netherlands, Dordrecht. chapter The Lambda Calculus and Adjoint Functors. pp. 181–184. URL: https://doi.org/10.1007/978-94-010-0526-5_7.
- [10] Scott, D., 1970. Advice on modal logic, in: Lambert, K. (Ed.), *Philosophical Problems in Logic: Some Recent Developments*. Springer Netherlands, Dordrecht, pp. 143–173. URL: https://doi.org/10.1007/978-94-010-3272-8_7, doi:10.1007/978-94-010-3272-8_7.
- [11] Scott, D., 1980. Relating theories of the λ -calculus, in: Hindley, J., Seldin, J. (Eds.), *To H.B. Curry: Essays on Combinatory Logic, Lambda-Calculus and Formalism*. Academic Press, Berlin, pp. 403–450.
- [12] Scott, D.S., 1986. Capturing concepts with data structures., in: Meersman, R., Sernadas, A. (Eds.), *DS-2*, North-Holland. pp. 341–354. URL: <http://dblp.uni-trier.de/db/conf/ds/ds86.html#Scott86>.
- [13] Wolfengagen, V., Dohrn, M., 2020. On capturing the variability in the modeling of individual behavior, in: Samsonovich, A.V. (Ed.), *Biologically Inspired Cognitive Architectures 2019*, Springer International Publishing, Cham. pp. 574–580.
- [14] Wolfengagen, V., Ismailova, L., Kosikov, S., 2021a. Applicative model to bring-in conceptual envelope for computational thinking with information processes, in: Samsonovich, A.V., Gudwin, R.R., Simões, A.d.S. (Eds.), *Brain-Inspired Cognitive Architectures for Artificial Intelligence: BICA*AI 2020*, Springer International Publishing, Cham. pp. 572–577.
- [15] Wolfengagen, V., Ismailova, L., Kosikov, S., 2021b. Combinator-as-a-process for representing the information structure of deep computing, in: Samsonovich, A.V., Gudwin, R.R., Simões, A.d.S. (Eds.), *Brain-Inspired Cognitive Architectures for Artificial Intelligence: BICA*AI 2020*, Springer International Publishing, Cham. pp. 565–571.
- [16] Wolfengagen, V., Ismailova, L., Kosikov, S., 2021c. Computational model for capturing the interdependencies on information processes, in: Samsonovich, A.V., Gudwin, R.R., Simões, A.d.S. (Eds.), *Brain-Inspired Cognitive Architectures for Artificial Intelligence: BICA*AI 2020*, Springer International Publishing, Cham. pp. 578–584.
- [17] Wolfengagen, V., Ismailova, L., Kosikov, S., 2021d. Imposing and superposing the information processes over variable concepts, in: Samsonovich, A.V., Gudwin, R.R., Simões, A.d.S. (Eds.), *Brain-Inspired Cognitive Architectures for Artificial Intelligence: BICA*AI 2020*, Springer International Publishing, Cham. pp. 585–590.
- [18] Wolfengagen, V., Ismailova, L., Kosikov, S., Babushkin, D., 2021e. Modeling spread, interlace and interchange of information processes with variable domains. *Cognitive Systems Research* 66, 21 – 29. URL: <http://www.sciencedirect.com/science/article/pii/S1389041720300905>, doi:<https://doi.org/10.1016/j.cogsys.2020.10.016>.
- [19] Wolfengagen, V., Ismailova, L., Kosikov, S., Dohrn, J., 2020. Superimposing semantic mesh to prevent information processes entanglement. *Procedia Computer Science* 169, 645 – 651. URL: <http://www.sciencedirect.com/science/article/pii/S1877050920303203>, doi:<https://doi.org/10.1016/j.procs.2020.02.197>. postproceedings of the 10th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2019 (Tenth Annual Meeting of the BICA Society), held August 15-19, 2019 in Seattle, Washington, USA.