ATLANTIS PRESS
Atlantis Highlights in Computer Sciences, volume 3
21st International Scientific Workshop on Computer Science and Information Technologies (CSIT 2019)

The Approach to Basic Abstraction Construction for the Architectural Schemes Design

Yury Rogozov
Institute of Computer Technologies and Information Security
Southern Federal University
Taganrog, Russia
yrogozov@sfedu.ru

Alexander Belikov
Institute of Computer Technologies and Information Security
Southern Federal University
Taganrog, Russia
anbelikov@sfedu.ru

Oksana Shevchenko
Institute of Computer Technologies and Information Security
Southern Federal University
Taganrog, Russia
ovshevchenko@sfedu.ru

***Abstract*—According to statistical studies, a large number of IT projects end in failure. This problem arises due to the inconsistency of the forms used by persons to represent the image (meaning) of the system. Existing decisions on the development of the system image or meaning relate to the field of architectural (conceptual) design. Analysis of existing approaches to architectural design shows that the basis of the approaches is the notion of "final result", in the form of an artifact (ready-made forms). On the one hand, representation forms of systems are used, that contain meaning (knowledge forms), on the other, meaning is fixed in forms (meaning forms). However, when converting the meaning represented by a particular form into forms that contain meaning (functions), the meaning is most often distorted, and sometimes lost. The main results presented in the paper are based on the proposed distinction between the concepts of knowledge and cognition as well as the forms of their presentation. Knowledge is currently represented as a form of action (function) or a form containing knowledge in itself. Cognition is proposed to represent by structure, mechanism of action, which is the process of knowledge forms design (functions). To test the results, by analogy of the Zakhman model, it was proposed to present the structure of actions of artifacts creation in the form of a table. With this approach, the architecture model is at the same time the meaning of the represented form of the table and the process of specific forms designing of actions containing the meaning (functions, forms of systems). Based on the results obtained, an effectiveness assessment is made using the obtained model to eliminate the semantic gap in the systems design.**

Keywords—architecture design, basic abstraction, information system, design process, form of meaning, form of knowledge

I. INTRODUCTION

According to statistical studies from The Standish Group International, a large number of IT projects end in failure: approximately 95% of projects have an overspending of various funds, on average, 60–160%; exceeding the terms, on average, 30–200%; more than 30% of projects cease, without completion [1]. This is observed against the background of the availability of documentation, modeling, design tools and frameworks of program code development, as well as when the field is saturated with good specialists.

One of the reasons is the complexity of transferring the originally inherent meaning into the system. It's known, in order to implement a project, it should be understood by the performer. According to researchers, the cause of the current situation is the inconsistency of the image (meaning) of the target system, which arises due to a misunderstanding between the persons involved in the development at different life cycle stages. This problem arises due to inconsistency of the forms (artifacts: project document, specification or model) used by the persons to represent the image (meaning) of the system.

According to the researchers, current situation is caused by the inconsistency of the image (meaning) of the target system, which arises due to a misunderstanding between the persons involved in the development at different life cycle stages (persons think about different things, at the design stage they think about design, at the development stage they think about programming, etc.) [2]. This problem arises because of inconsistency of forms (artifacts: project document, specification or model) used by persons to represent the image (meaning) of the system. Although the vast majority of modern research and practical developments aim to create various forms of the object and methods for constructing them [3], these forms are understandable only to a narrow circle of persons, and as a rule are at one life cycle stage of information systems development [4].

II. THEORETICAL PROBLEM ANALYSIS

To analyze the problem and find solutions, it is necessary to understand the ways of obtaining meaning, forms, as well as the process of transforming meaning into system forms.

Existing decisions on the system image or meaning development relate to the field of architectural (conceptual) design, since architecture means a certain sense of the system.

Analysis of approaches to the architectural systems design shows that the basis of existing approaches is the concept of "final result", in the form of artifact (ready-made forms). There are two types of forms:

- The form of meaning – is the meaning fixed in a certain form, which allows to display the subjective (cognitive process) component of the system design process.

An example of meaning forms is the Rich Picher approach [5]. This approach was developed by Peter Checkland to describe complex situations within the framework of the soft systems methodology. "Rich Picher" models are a collection of graphic primitives, text, sketches, etc. to describe the meaning of the system.

Another example of this form is the "Systemigrams" modeling method proposed by J. Boardman [6]. The main

idea of "Systemigrams" is the compatibility of structured text and its graphic display representing the image (meaning) of the system.

- The form of knowledge – is knowledge that is fixed in an objective form.

The form of knowledge is a form that without the help of person or without explanation to it can not be unambiguously understood. As a rule, these models have a strict structure and can be formalized. Examples of such forms are: mathematical models, program code, etc. This type of form is used for system design.

The transition from the meaning form to the system form occurs in the design process (Fig 1).

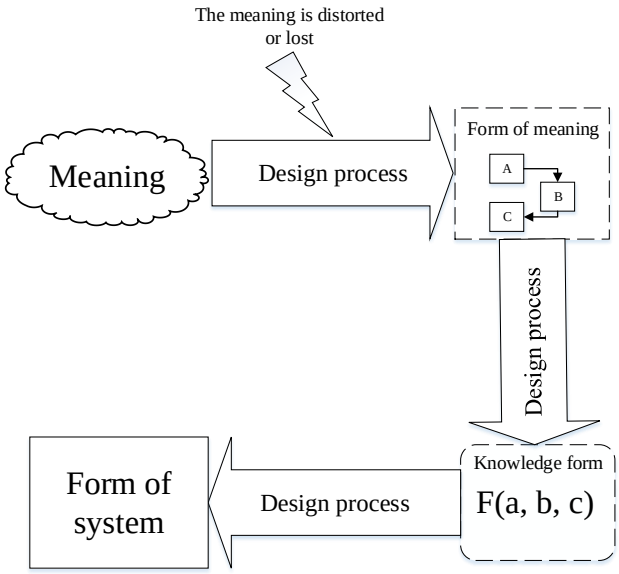

Fig. 1. The process of transforming the form of meaning into the resulting form

However, when transforming the form of the meaning (in the design process) into the form of knowledge the meaning is most often distorted and sometimes lost.

The second problem is that the process of system architecture creation is not formalized, it is an empirical process, which obviously leads to the first problem. But in order to formalize the process of architecture design, a special basic abstraction is needed (integrating concept), which, on the one hand, would settle the form of meaning instead of forms of knowledge, but, with the ability to be presented as a cognitive process for specific forms (forms of knowledge) design.

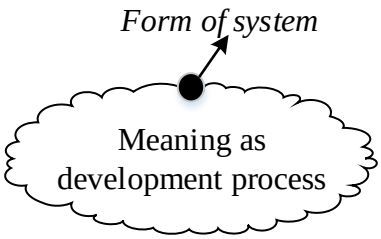

Fig. 2. Integrating concept, which generate the result forms

The form of meaning is the action structure, which is the process of forms of knowledge (functions) design [7].

The main property of the meaning form should be the possibility of its transformation into knowledge forms. Therefore, it is necessary to create special forms of meaning that can generate forms of knowledge.

To solve this problem, it is proposed to use a kind of integrating concept that combines the form of meaning and the design process (cognitive process) (Fig 2).

At the same time, this integrating concept should be a structure that allows to generate various forms of knowledge and then transform to system forms.

III. Problem Analysis from the Point of View of Information Systems Design

In order to understand the failures reasons in the information systems development process, it is necessary to consider the existing process of information systems (IS) development in details.

Conventionally, the process of information systems development can be divided into 3 main stages:

- Problem domain analysis;
- Information system design;
- Information system development (implementation).

Information system development always begins with setting goals by the customer. Next is the problem domain survey, which can be divided into several steps: collecting the information, business processes construction ("as is" model), and the last step of the domain analysis stage is business processes optimization ("to be" model).

After that, the information system design stage begins. This stage is divided into the following sequence of steps: requirements specification, modular architecture design, detailed design. Each of these steps can be divided in turn into smaller steps.

The next stage is the information system development. This stage is divided into two main steps: programming and testing.

At each stage of the life cycle, a certain set of artifacts (technical solutions, models and documents) is generated, and for each stage the documents and decisions made at the previous stage are initial.

However, this sequence of stages has some disadvantages. The meaning is contained in the artifact and only the designer of the form (model) knows this meaning. Experts of each design stage create different forms of supposedly the same meaning, but in practice the meaning contained in the forms is completely different.

Consequently, a problem arises, which consists in the disunity and the following differences in the goals and forms that represent meaning, among different groups of persons (problem domain experts, system analysts, developers, etc.) participating in the project and the emergence of so-called misunderstandings. There are always a lot of such "inconsistencies", and losses from their elimination and entire system transformation are expressed in significant amounts, as a result budget and the time frame of the project implementation are increased.

To combine the meaning forms of different groups of persons involved in information system development is applied architectural design. Architectural design is a type of activity aimed at obtaining the system meaning as a whole from a set of unrelated meanings of artifacts. Analysis of the existing approaches to the architectural design of information systems (the Zakhman model, the 4 + 1 model, the SAM architecture model, etc.) shows that all them are based on the concept of the "final result", in the form of an artifact [8-10].

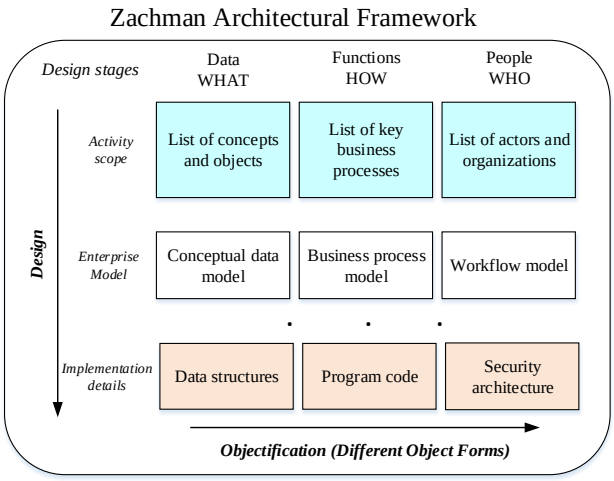

Fig. 3. Simplified Zachman model for information systems development

An artifact is a document or model. The appearance of the artifact depends on selected means, with the help of which it is possible to obtain this artifact (Fig. 3). For example, to obtain the "Business Process Model" artifact in the Zachman framework, IDEF and DFD methodologies are most often used.

In the course of the architectural representation development, a number of final forms (artifacts) are constructed, between which it is necessary to carry out numerous transitions including different subject knowledge (analyst - designer - developer). Transitions are carried out by coordinating the final artifacts of the object (business process - project documentation - source codes).

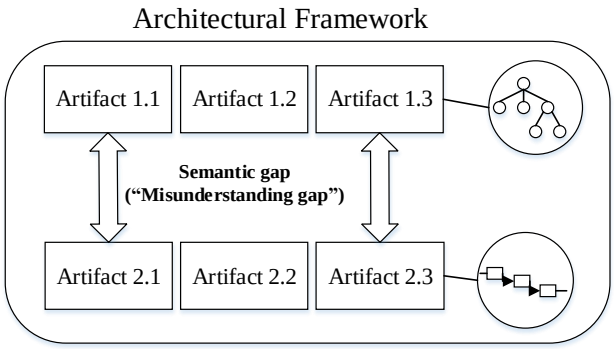

Fig. 4. Zahman model representation in the form of artifacts

Thus, the main problem of such architectural representations as the Zachman framework is the impossibility of an unambiguous transition from one artifact to another (Fig. 4). The main negative consequence of this problem is the unpredictability of the result, and, finally, risks of re-developing or re-performing of certain life cycle stages, which also do not provide a full guarantee of desired result obtaining. The transition, or rather the coordination of the final form of one design stage with the final form of the subsequent stage, is carried out. That almost always fails.

IV. Proposed Idea and Approach to Solving the Problem

The basic idea of indicated problems solving is based on the proposed distinction between the concepts of knowledge and cognition and the forms of their presentation. Knowledge is currently represented as a form of action (function) or a form containing knowledge in itself. According to H. Maturana, "cognition as knowledge is action, but the form of knowledge is the result of cognition" [11]. Therefore, cognition is proposed to represent by structure, mechanism of action, which is the process of knowledge forms design (functions). For example, the existing design process is essentially a structure for creating the specific knowledge forms or systems. As a category is a generic concept denoting a set of specific concepts, so the action structure is a category in relation to specific knowledge forms as actions forms represented by functions.

It is assumed that the theoretical system should correspond to the meaning, which indicates possible alternative solutions to the question of creating the specific forms of action (knowledge). The logic of constructing the specific forms of knowledge (actions) is reflected in action structure.

It is proposed, firstly, to change the design process itself, in which the one form design should take place, or rather its construction from components. Secondly, at each design stage it is proposed to represent each stage in the form of dynamic processes of obtaining or building forms instead of creating ready-made forms. And also to coordinate with each other dynamic processes of building forms instead of ready-made forms.

The currently existing methods and presentation forms (languages and notations) cannot solve this problem. It requires a transition to a qualitatively new level of understanding of the development process and the methods and forms of presentation. There is a need to develop new approaches and technologies (knowledge forms, semantic meaning of which is clear to all project participants), ensuring the harmonization of the knowledge used at different information systems development life cycle stages.

The idea is to use a system approach to solve the indicated problem. A system approach should be a process of transforming the knowledge forms corresponding to systems thinking (meaning) into forms used by system engineering (forms of meaning). It is necessary to develop such a new system notation, which will allow to move from the forms of systems thinking representation (from the meaning) to the forms of system engineering (to the process of obtaining meaning) and vice versa.

For these purposes, an integrating concept is needed, which will allow to describe the meaning and the process of its production (Fig. 5).

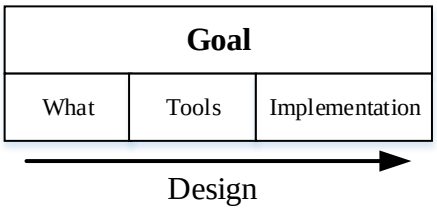

Fig. 5. Proposed basic abstraction

The proposed concept is a basic abstraction of a tool for solving the coordination problem, or rather the integration of different domain knowledge (forms), because it integrates two different points of view – the knowledge carrier of the problem domain ("What" characteristic) and the technical specialist ("Implementation" characteristic) [12]. Inclusion in its structure of an explicit specification of external environment and the of the action implementation purpose allows to fix a single point of view on the object and identify the available resources for the action implementation.

The components of the proposed concept (What, Tool and Implementation) are the design process stages, which are semantically inseparable and imply nesting into each other, which cannot be said of the existing approaches to the design processes construction (Fig 6).

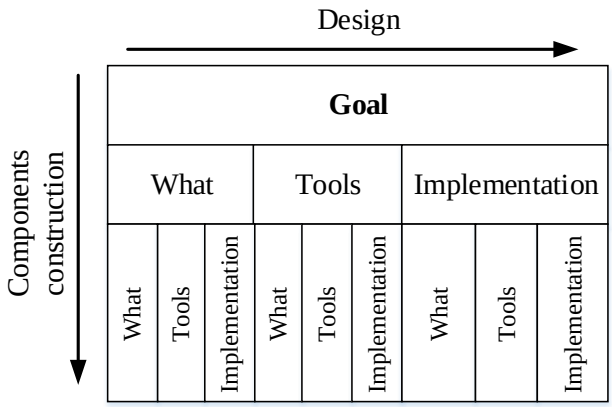

Fig. 6. Components of basic abstraction

The described basic system abstraction of the integrating concept does not depend on the level of application; it allows multiple hierarchical nesting of actions specification to obtain certain aspects.

The penetration of design stages allows us to say that at all design stages there is a construction of the same meaning form, which seems to be an integrative concept of unity.

Therefore, we can provide an algorithm for transforming the meaning form into the knowledge form:

1. Construct a meaning structure, which is an empty set of interpenetrating concepts;
2. Fill the integrative concept with content (What, Tool, implementation) describing the process of knowledge form construction;
3. To objectify the "What" concept, presenting it as an integrating concept;
4. To objectify the concept "Tool", presenting it as an integrating concept;
5. To objectify the concept "Implementation", presenting it as an integrating concept;
6. Repeat steps 3-5 until the final content is filled with domain data.

The transition from meaning forms to specific knowledge forms is carried out by filling them with specific content. The process of obtaining a form of concrete knowledge (concrete actions) about the system (presenting the target system meaning) is proposed to consider as the process of filling the possible action form with content.

V. Comparison of Existing and Proposed Approaches

Next, we compare the architectural framework of Zakhman with the proposed basic abstraction. To do this, we introduce the characteristic "Operations on Meanings". Using this operation, we estimate the number of constructed meanings in each approach.

There are 5 stages in the Zachman framework for information systems architecture design: planning, analysis, conceptual design, technological design and development. The stages in the Zachman table are arranged horizontally. Vertically located characteristics that describe a particular stage from different points of view: data, functions, etc. At the same time at each stage the meaning is formed. In the case of architecture consistent construction, the meanings are connected to each other sequentially according to stages. However, the meanings obtained at different stages are not interconnected, and the coordination of meanings occurs empirically (Fig 7).

Unlike existing approaches to system architecture construction, proposed approach uses basic abstraction, which allows constructing unified meaning and describing it. Design process consists of 2 stages (Fig 8):

1. Construction of the whole, formation of original meaning;
2. Designing the components of the original meaning.

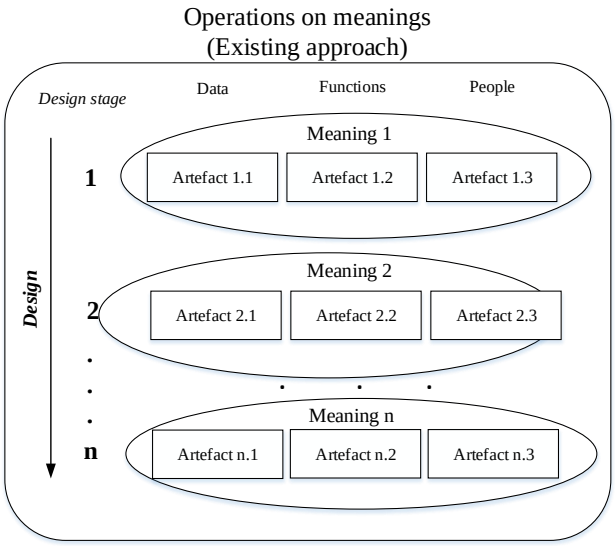

Fig. 7. Analysis of "Operations on the Meaning" in the existing approach

The greater the number of meanings obtained during the construction of system architecture, the greater the degree of inconsistency of the originally inherent meaning to the final (form of knowledge).

In the existing approach (considered earlier – Zachman framework), in the process of information system architecture creation, 5 different meanings (meaning forms) are created, this number is equal to the number of stages of architecture creation. When moving from one meaning (usually made by one person) to another meaning (made by another person), part of the meaning is lost or distorted. Since the number of meanings is 5, the described transitions between meanings will be equal to 4. It turns out that the construction of one architecture loses or distorts the meaning 4 times.

As noted earlier, a unified meaning is created in the proposed approach, so the need of transitions between different meanings eliminates.

Therefore, it can be concluded that in the proposed approach, the loss of meaning will be 4 times less than in the architectural Zachman framework.

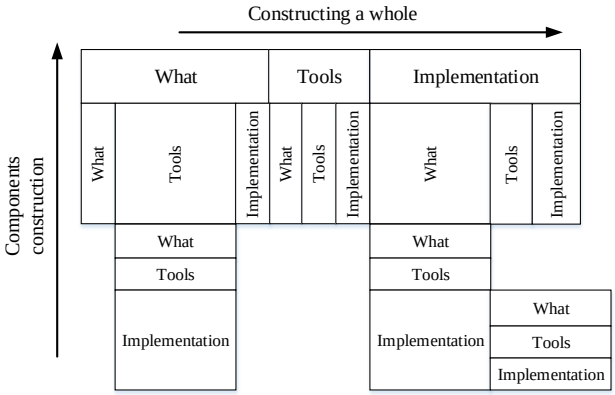

Fig. 8. Analysis of "Operations on the Meaning" in the proposed approach

VI. Conclusion

The analysis made in the paper made it possible to identify the problem of inconsistency of meanings and their forms, and to suggest possible solutions. Based on the analysis, a qualitatively new approach and an integrating concept for the description of information systems development process was proposed.

The architectural representation of the design tool proposed in the paper has the following advantages:

- the use of the integrating concept of unity ensures the absence of semantic gap between the design stages, since each stage objectifies the content of the whole component;

- coordination of multidisciplinary knowledge due to the specification of two different points of view – the knowledge carrier of the problem domain ("What" characteristic) and the technical specialist ("Implementation" characteristic);

- reduction of risks of non-compliance of the developed software with the original goals.

The ideas proposed in the paper should solve the existing problems in the information systems development and present a new look at the information systems design and development in general.

Acknowledgment

The reported study was funded by RFBR according to the research project № 17-07-00098 A.

References

[1] Standish Group 2015 Chaos Report – Q&A with Jennifer Lynch, URL: https://www.infoq.com/articles/standish-chaos-2015

[2] Sally W. Aboelela, Elaine Larson, Suzanne Bakken, Olveen Carrasquillo, Allan Formicola, Sherry A. Glied, Janet Haas, Kristine M. Gebbie Defining Interdisciplinary Research: Conclusions from a Critical Review of the Literature // Health Serv Res. 2007 Feb; 42(1 Pt 1): 329–346.

[3] Herre H. General Formal Ontology (GFO): A Foundational Ontology for Conceptual Modelling. In: Poli R., Healy M., Kameas A. (eds) // Theory and Applications of Ontology: Computer Applications. Springer, Dordrecht (2010).

[4] Schooley, B. L., Feldman, S. S., & Alnosayan, N. S. (2011). Development of a Disability Employment Information System: An Information Systems Design Theory Approach // System Sciences (HICSS), 2011 44th Hawaii International Conference on, 1-10.

[5] Checkland, Peter. "Soft Systems Methodology." Systems Research and Behavorial Science, 2000: pp.11-58.

[6] Boardman, J. T. and B. J. Saucer, Systems Thinking: Coping with 21 st Century Problems. Boca Raton, Taylor & Francis, 2008.

[7] Rogozov Yu., Bermúdez Soto J.G., Sviridov A., Lipko Yu., Belikova S., Belikov A., Kucherov S., Shevchenko O., Borisova E., Egorov A. Texts segmentation and semantic comparison: method and results of its application / Conference of Open Innovation Association, FRUCT. 2018. № 23. С. 321-328.

[8] Krzysztof Kluza, Piotr Wi´sniewski, Antoni Ligeza, Anna Suchenia,and Joanna Wyrobek Knowledge Representation in Model Driven Approach in Terms of the Zachman Framework // ICAISC 2018, LNAI 10842, pp. 689–699, 2018. https://doi.org/10.1007/978-3-319-91262-2_60

[9] Kruchten, P.B.: The 4+ 1 view model of architecture. IEEE softw. 12(6), 42–50 (1995)

[10] Mrdalj, S., Jovanovic, V.: Mapping the UML to the Zachman framework. In: AMCIS 2005 Proceedings, p. 315 (2005)

[11] R. Maturana, Humberto & J. Varela, Francisco. (2019). El árbol del conocimiento : las bases biológicas del entendimiento humano / Humberto Maturana R., Francisco Varela G.. SERBIULA (sistema Librum 2.0).

[12] Belikova S.A., Rogozov Y.I., Sviridov A.S. Method of user interface design based on semantic approach. Advances in Intelligent Systems and Computing. 2019. V. 763. pp. 311-318.