CSIT'2009

Proceedings of the 11th International Workshop on

Computer Science and Information Technologies

Crete, Greece
October 5–8, 2009

Volume 2

The International Workshop on Computer Science and Information Technologies

UDC 004.7

Proceedings of the Workshop on
Computer Science and Information Technologies (CSIT'2009),
Crete, Greece, October 5-8, 2009. Volume 2
Ufa State Aviation Technical University, 2009

ISBN 978-5-86911-965-0

The Workshop on Computer Science and Information Technologies (CSIT) is an annual high quality international conference providing a forum for exchange of scientific achievements between research communities of Eastern Europe and the rest of the world in the area of computer science and information technologies. *The 11th* International *Workshop on Computer Science and Information Technologies (CSIT'2009)* was aimed at bringing together researchers, practitioners, developers, and users to explore new concepts, tools and techniques for advanced information technologies. The symposium lasted three days, the program of each day being organized in the sessions.

The papers are included as they are submitted by the authors, without editing

IT-sphere constructive directions

S.G. Maslov
Theoretical foundations of computer science department
Udmurt state university
Izhevsk, Russia
e-mail: mshsci@yahoo.ru

A.P. Bel'tukov
Theoretical foundations of computer science department
Udmurt state university
Izhevsk, Russia
e-mail: belt@csdc.udsu.ru

O.A. Morozov
Department of university computer network
Udmurt state university
Izhevsk, Russia
e-mail: oam_oam@mail.ru

Abstract[1]

Problems and conditions connected with IT-sphere constructive directions are investigated

1. Introduction

In the contemporary society the IT-sphere becomes a system-forming factor of the high quality of life, a factor of operative and flexible response to the crisis phenomena, and also the strategic direction in forming and developing the vital activity of both the *separate person* and the society as whole. Besides, the IT-sphere should catalyze and concentrate knowledge of a highly moral harmonious sustainable development of the person, the society, and the nature. These processes are a prerogative of the creative approach when the thought should overtake carried out actions and ensure a sustainable development

In IT-sphere systems the phenomenon of complexity is manifested most vividly, since they personify in themselves attempts to construct systems at the boundaries of human knowledge and integration of our knowledge and understanding in the most complete and mass form both the surrounding reality and the person himself, and frequently in critical situations with an unstable behaviour. This complexity phenomenon is the basic "headache" of more than one generation mathematicians, programmers and computer scientists, but just overcoming it gives a key to the sustainable and break-through development.

Proceedings of the 11th international workshop on computer science and information technologies CSIT'2009, Crete, Greece, 2009

2. Constructive directions

Constructing programs or information systems is a realization of our creative and constructive abilities fulfilling sense of understanding a life, i.e. forming requirements, revealing problems and the purposes. Position of the creator (programmer, computer scientist, mathematician, designer, etc.) during overcoming the phenomenon of complexity always performs pendulum motions, between the essence and the phenomenon, the object and the process, changing the point of view, accents, criteria and the purposes. In more detail enormous creative way in the IT-sphere can be reflected as a stream of ideas and attempts of their realization:

- universal programming languages (PL/1, Algol-68, Ada);
- *universal technologies of programming* (top-down programming, assembly programming, R-technology, Σ-programming, UML-technology, etc.);
- multifunctional environment of programming (NUT, Delphi, Maple, etc.);
- *a variety of programming styles* (imperative, logic, functional, sentential, object-oriented, automata-based , event-based, constraint, tensor-based, genetic, associative, priority, etc.);
- *cross-platform systems* (CORBA, J2EE, NET, etc.).

It is actually, continuous experimentation with various forms, logics and semantics. It inevitably leads to the main question: what is the basis and what are mechanisms of constructing necessary objects and processes (effective reconstructing the system and/or creating something fundamentally new or innovations)? What is the interrelation between *a virtual reality* (or the imagined world, R_V) and *a physical reality* (or the real world, R_F)? If one understands a part of virtual reality (which can be materialized in physical reality) as *a constructive reality* (or artificial world, R_C), then what are the laws of its forming and evolution:

$$R_C = R_V \cap R_F, \quad R_F = \{A_i,\ i \in D\}.$$

Furthermore, in the process of constructing systems in the IT-sphere it is possible also to speak about other three physical realities:

$$R_F = R_{0F} \cup R_{1F} \cup R_{2F} \cup E$$

- *0-sort reality (R0F)* is the subject himself, "bio-computer", that who makes decisions, formulates the purposes, problems and criteria of an estimation;
- *1-sort reality (R_{1F})* is that which is observed and simulated, which is transformed and used;
- 2-sort reality *(R_{2F})* (a traditional, molecular or quantum computer), that is used for simulating and research, under the safe and scaled conditions (effective in some sense);
- *E* is the environment of the information propagation in which sensations *EP*, polysensory images *ES*, signs *EZ*, ideas *EI*, and thinking *EM* are developed:

$$E = E_P \cup E_S \cup E_Z$$

$$R_V = E_I \cup E_M \cup E_Z$$

i.e. the worlds of sensations, signs and polysensory images.

Simple interaction of physical realities 0^{th} and 1^{st} sort for contemporary human is already insufficient, therefore physical reality of 2^{nd} sort is appeared. The symbiosis of two calculators (0^{th} and the 2^{nd} sort of physical realities) is control of information, energy and material streams, achieved by means of a language and image form of their idea (abstract and concrete sign forms and operations with them).

Within the framework of the last three physical realities, if one stands on the system point of view [1], then the basic creative process (*PROGRAMMING*) can be defined as:

controlling information environment and information streams between physical realities of 0^{th}, 1^{st}, and the 2^{nd} sort, in process of which the information artifacts (images) of the abstract and real systems (objects or processes), which satisfying to needs of subject are created and used.

Information artifacts are constructed within the framework of the world of signs using *natural and/or artificial languages.*

The *natural language* is on form which focuses our attention on understanding, connectivity, analogousness and integrity of our representations and ideas. *Artificial languages* are the forms, which give the deeper and more exact representations of information and knowledge that are adequate to the solved problem, to the abilities of subject, and accentuate our attention in the computability, reliability and efficiency. These forms are connected with the processes of formalization and deformalization:

natural language → professional prose → terminological system → language of the applied scientific theory (system description layers)

It is important to make the whole chain of languages in general to have a coherent system of semantics, allowing shape representation and computing (in the particular case, calculations, inference, transformations, navigation, etc.) in a 2-type of environment. It is expedient from this point of view to refine the interrelation between the systems analysis [1,3] and language activity *(point of view appropriate to clarify).*

The *morphological layer* defines the set of the system parts; it corresponds to the language syntax. However, the syntax, pretending to build constructions correctly, frequently leads to "impracticable" constructions, whose imperfection is manifested in the behavior.

The *functional layer* defines dynamic behavior and transformation of systems. It corresponds to different semantics, for example [2, 4], denotational semantics (objects conversion calculus, the dynamics of objects aggregation and detailing), axiomatic semantics (axioms from a meaningful point of view are divided into constants (laws) and variables (condition), rules of inference), functional semantics, etc.

The *attributive layer* defines the qualitative and quantitative features of systems. It corresponds to logical semantics, which describes the features of the structure and the behavior of systems, in particular, their reliability, security, the complexity and ergonomic aspects. These features can be used in the process of constructing, selecting and executing the systems.

The *libernetic layer* has no obvious conformity in language activity. It concentrates attention in the invariant and variative, projection and integrated design features and behaviour of the system. The important special feature of this layer is passage from the base to the generalized ideas (coordinates) by means of control programs that are necessary and sufficient for achieving the purposes.

The *genetic layer*, finding out and overcoming obstacles and contradictions in the system construction, is a basis of revealing and concentration of experience in fixing of steady structures and behaviour of systems.

The genetic and libernetic layers are united in the *control* process of the system life cycle.

Today it is possible to establish that the concrete definition of conceptual layer into the mathematical and algorithmic layers frequently is produced in the implicit form. The reason lies in the fact that programmers, using one or other traditional programming style, come to the mathematical, computational or logical model specific in it. It strongly limits possibilities in constructing IT-systems, since it leads to the unnatural and illusory adequacy, to the technological breaks in resolving problems and using by a programmer one strategy or method in all cases of life. Actually, when the selection of more adequate mathematical form needs serious study, it is refuted, i.e. roughly speaking, the geometric (tensor), logical, statistical-probability and illegible formulations of the problems are equalized. This extremely important aspect of the constructing information artefact demands separate discussion.

This article for a while we will focus on programmer aspects. It is necessary to note incorrect equating concepts of *"paradigm"* or *"style"* of programming. It leads to the loss of concepts clarity, in particular, concept of constructiveness. There is no clearness of function distribution in the framework of the chosen three realities. If we recognize that information system is "bio-computer" and computer symbiosis in information processing, then it is necessary to generate in this respect the points of view on the attributes of programming, presented in the form of table (1), with a mark "*" at the interacting aspects.

Table 1. Aspects of programming

(*Sub* is the subject (the person, collective). *NR, AR, WS, SW, WSI* are acronyms for the natural reality, the artificial reality, the world of sensations, the sensory world, the world of signs accordingly)

Attribute of programming	*Sub*	*NR*	*AR*	*WS*	*SW*	*WSI*
Styles	*	*		*		*
Forms	*		*		*	*
Kinds (Sort)		*	*		*	*
Technique	*		*	*		*

From the point of view of the chosen realities it is expedient for the programming to isolate *styles, forms, types and techniques* as attributes of programming, which are concentrated at the uniform core, the world of signs.

It is natural that the basic problems and tasks appear in heads of people (persons or among subjects). Therefore it is required to organize in these heads clarity of ideas and good running of thoughts and reflections about their surrounding world and about themselves. It is required organize effectiveness and intellectual development of individual and collective thinking and activity in the process of observing, knowing, constructing and understanding reality. All it should be carried out by means of the ideal, abstract and concrete images presented in the world of signs, and also sensations perceived through the world, gestures and mimicry. The core here is the *style of thinking*, considering representative system, intelligence, a social type, psychotype, temperament and other factors. The *style* is constructed on the basis of system psychophysiology (JU.I. Alexander, V.B. Shvyrkov) and representations about levels of cognitive organization and regulation of behavior (B.M. Velichkovsky). In this case, it can be spoken both about the metaphorical, the motor, the soft and the rigid, the abstract and the concrete, and other styles and about the concept of the unity of consciousness and emotions, about the world of signs, which fixes the intermediate and end (eventual) results of thinking and activity.

The *form* is a part of the subject intention materialization result, which makes it possible to present the information artifact not only in the world of signs, but also in the sensory world. It is possible to speak about polysensory, bisensory (multimedia), linguistic, geometric, logical and other forms, which make it possible to represent clearly and tangibly abstract and concrete projections, and also integral the vision of information artifact and characteristics of their control.

The *kind* is connected with character of transforming and constructing forms, interaction between the natural and artificial realities, with balance optimization between objects and processes in the information artifacts representation. The following kinds are now well known: transformational, concretizing, assembling, synthesizing, conclusive (proving, evidence), verifying, fractal, genetic and/or evolutionary, and a lot of other ones.

The *technique* is connected with organizational aspects of the subject activity in the process of creating and using the information artifact. It is possible to mention as known techniques the following ones: designing, extreme or brigade programming, technique of the formalized tasks, and others.

In the process of programming and executing programs the essences and the phenomena are reflected in the form objects and processes or concepts and their transformations. To follow a certain lingual form (some language) the term, which designates concept, can have a constant value (invariant or constant) or variable one (variation or alternatives). Besides, there are points of view reflecting applicability of a term in a context. The understanding (interpretation) of the term can be expressed by the chain of concepts, which fixes a certain context of conversion or control:

term (object, process, property, etc.) → value → state → event → signal → priority→ action→ criterion → evaluation→ problem-solving.

For the completeness of idea it is necessary to introduce actualization of concepts in the process of computing (calculations, inference, transformations (conversions), navigation, etc).

Table 2. Matrix of computing strategies

(*Uncon, Con, Mix* are acronyms for unconditional, conditional, mixed processes accordingly. *Sy* and *Asy* are acronyms for synchronous, asynchronous processes accordingly)

Execution	*Uncon*		*Con*		*Mix*	
Consecutive						
	Syn	*Asy*	*Syn*	*Asy*	*Syn*	*Asy*
Parallel						
Concurrent						
Collaborative						
Antagonistic						

In this case various *strategies of computing* naturally turn out.

Any information artifact for the quantitative and qualitative estimation should be interpreted within the framework of LT-system (length-time measuring system). In this way it is possible to obtain analogs in the information area for the laws of energy and power conservation [3, 5].

Within the framework of the interpretation the estimation of structure and function complexity of the created information artifact are obtained. In the *dynamic complexity* the following combinations have a sense:

- L^1T^0 is the abstract memory size or computing field size for parallel computing (information access distance);
- L^3T^0 is a full memory size or volume of the processor field for unlimited parallelism;
- L^3T^1 is a rent of memory for consecutive computing or work amount for parallel computing;
- L^0T^1 is the time for consecutive computing or a delay for parallel computing.

In the *static complexity* it is possible to consider the following combinations:

- L^3T^0 is the algorithm volume (in structure elements);
- L^1T^0 is the diameter of algorithm (the size in 3D memory);
- L^2T^0 is a ratio of volume to diameter or a relative surface of the algorithm (the area of its projection or complexity degree (degree of entanglement)).

Questions about conservation laws of in computational complexity are much more complex, than the corresponding questions in physics. They correspond to the different trade-off problems of the complexity theory. For example, it is not clear, whether it is possible to reduce the volume of computing work by increasing the delay (which originally was essentially, but not catastrophically less, than this volume at a consecutive computing).

Complexity picture is capable to give relatively complete representation for forming constructive mechanisms in whose frameworks it is possible to allocate two basic directions, caused by symbiosis.

The first direction is a solution of creative problems on a basis of libernetic and genetic systems representations.

The second direction, together with the logical approach allows solving problems of parametric, structural and base level of system representations (see the figure 1 and the table 3).

Table 3. Levels of control and synthesis

Levels of control	Reducers of freedom degrees	Generators of freedom degrees	Note and refinement
Parametric level			
	$x_i\updownarrow \rightarrow const$	$const \rightarrow x_i\updownarrow$	a change in the number of freedom degrees, fixation, regulation
	$x_i\uparrow \rightarrow x_k\downarrow$, $x_i\downarrow \rightarrow x_k\uparrow$ - *negative* $x_i\downarrow \rightarrow x_k\downarrow$, $x_i\uparrow \rightarrow x_k\uparrow$ - *positive*		the feedback – reflexes and reactions
Structural level e_a – *elements*, r_v – *connections, relations*			
	e_i, $e_j \rightarrow e_k$ – "*gluing*"	$e_k \rightarrow e_i$, e_j - "*hewing*"	structure change
	$\pm (e_i)$		no uniqueness
	$+ (r_m)$	$- (r_n)$	structure change, synergy and invariants
Basic level - *static* - *dynamic*			
	$x_i \rightarrow x_k$ - a change in basis of variables; $x_i \in [a,b]_i \rightarrow x_i \in [c,d]_i$ - a change in intervals of existence to variables; $\mathbf{u}_l = \mathbf{T}_{ip}\overline{\mathbf{u}}_p$, $\overline{\mathbf{u}}_i \in \mathbf{Ox}_l$, $\mathbf{u}_p \in \mathbf{Az}_p$ $\mathbf{u}_l \xrightarrow{\mathbf{F}} \mathbf{u}_k \Rightarrow \mathbf{u}_l \xrightarrow{\mathbf{G}} \mathbf{u}_k$		a change in substance or point of view, a change in functional saturation (conversion, transformation)

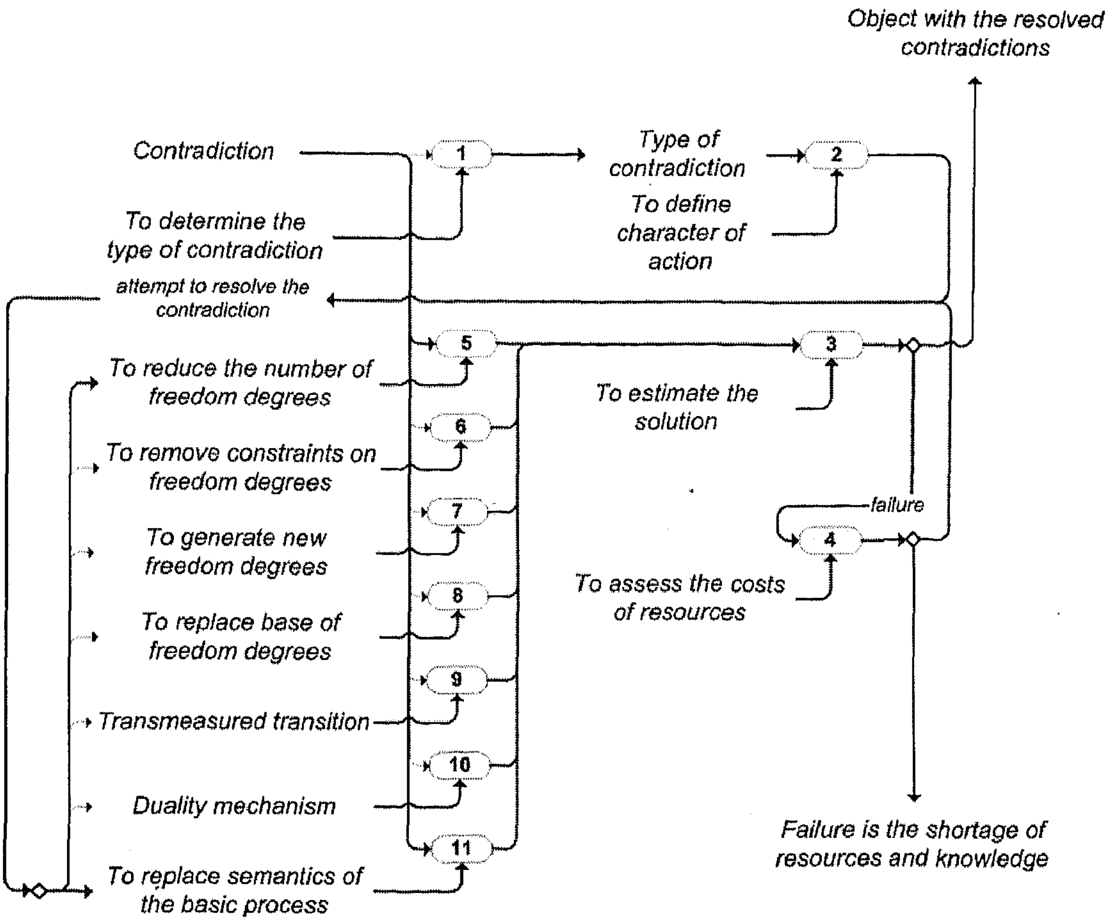

Fig. 1. Conflicts and contradictions resolution process by freedom degrees organizers

3. Conclusion

Thus, presented here the concept of descriptions of IT-sphere:

- *firstly,* integrates the well differentiated fields of knowledge and the chosen roles in the IT-sphere, in this case there appears the complete knowledge about the processes within the framework of the life cycles for human, a civilization and information systems;
- *secondly,* generates nonconventional researches in an estimation of IT Sphere;
- *thirdly,* is the powerful catalyst of a sustainable development of human, society and their surrounding nature.

Acknowledgments

This investigation is supported by the Grant 08-07-00460-a "Development of constructive activity web-sphere based on the libernetics approach (applied to biomechanics)" (2008-2010) of Russian Foundation For Basic Research.

References

1. Bel'tukov A.P., Maslov S.G., Morozov O.A. "Libernetics paradigm in IT-sphere". In: *Proc. of II-nd All-Russian scientific conference with international participation. Technologies of informatisation in professional activity (in science, education and industry),* Part I. "Bon Antsa" Publishing centre, Izhevsk, Russia, 2008, pp. 37-52.
2. Wolfengagen V.E. "Constructions in Programming Languages: Methods of Description". Center "JurInfo", Moscow, Russia, 2001.
3. Kuztetsov O.L., Bol'shakov B.E. "Sustainable development: Scientific foundations for design in the nature-society-human system". "Gumanistika", Saint-Petersburg, Moscow, Dubna, 2001.
4. Nepeivoda N.N., Skopin I.N. "Programming foundations". Institute of computer science, Moscow, Izhevsk, 2003.
5. Petrov A.E. "Dual networks tensorial method". "Wildlife management information technologies center", Moscow, Russia, 2003.