Experience-Oriented Conceptual Space for Designing the System with Software

P. Sosnin, *Member, IEEE*

***Abstract*—The design of any system with software is a behavioral process that encompasses a certain area of the physical space on the definite interval of time. The space of designing is useful for expressing in forms of the corresponding models. The paper presents a reflection of an operational space onto a conceptual space CS(t) that is experience-oriented. It can be achieved via question-answer interactions of designers with an accessible experience and it, in its turn, will facilitate increasing the success of designing.**

***Keywords* — Conceptual space, designing, question-answering, project task, software engineering.**

I. Introduction

In design theory and practice, the use of the term “space” has a long history, during which this term was applied to various interpretations for different purposes. At present, the role of the constructive use of design space and its models is steadily expanding. Constructive accounting of the design space is especially necessary for the development and application of Socio-Cyber-Physical Reality in which the interaction of a person with a computerized environment is not limited only to interaction with the physical world.

For the search of forms that are appropriate for constructive accounting such essence as the space in designing, it is suitable to investigate developing the software intensive systems (SISs) that are typical examples of computerized environments. It is especially important because the degree of success in designing the SISs has been extremely low for the last twenty years[1].

The main reason for this is a very high complexity of perceiving by designers of situations in important points of the project life cycle. That is caused by an unpredictable appearance of numerous situational factors that it is dangerous not considering in decision-making (for reference in the useful review [2] about 400 of such situational factors are mapped into 48 of variants distributed among 11 groups). Furthermore, the negative influences of human factors often accompany the arisen situational factors.

In our deep conviction, for reducing the complexity in reacting to indicated situations, it is necessary to provide the designers the use of interacting with an appropriate model of space [3], in which the situation arises.

Since the appearance of situations for decision-making is unpredictable, the appropriate model of the space is unique for any SIS-project, and designers should build this model in coordination and in parallel with creating the other artifacts of the design process.

In the paper, we suggest modeling the operational space OS(t) of designing (an environment of designers’ actions) in a form of an experience-oriented conceptual space CS(t), the structure and content of which are results of reflecting the OS(t) onto the semantic memory of the question-answer type.

It is expected, the designer uses the toolkit WIQA (Working In Questions and Answers) that is intended [4] for conceptual designing of the SISs.

II. Preliminary Bases

In this paper, we understand any space as a set of objects with relations among them that are involved in the activity of designers who fulfill a set of operations with objects and their group. Among such operations, one needs to mark Generating, Choosing, Combining, Activating, Investigating and Transforming the necessary objects. Obtained results can be explicit or implicit reasons for transitions to objects of another space, for example, “Reflecting on the related object” or “Activating the related object.” Interests of our research are aimed at reflections of essences of the OS(t) onto CS(t), objects of which are uploaded into the semantic memory.

In interactions with objects of the CS(t), designers can use operations that are similar to indicated above, but they are implemented in active use by designers of accessible experience and its models. Such works are based on understanding the questions as naturally-artificial phenomena that provide access to the experience [4]. In this sense, verbal expressions of questions and answers are ‘traces’ of the access to the experience. Models of such traces find their objectifying in the conceptual space located in the semantic memory of the WIQA-environment.

Our constructive work with the CS(t) began about fifteen years ago when we began to master the reflections of the OS(t) onto the semantic memory of the question-answer type (QA-memory) [4]. In the structure and content of the operational space, we oriented on developing the systems with the use of methods and means that have similarity with components of Rational Unified Process [5].

The general scheme of the investigated reflections is

Paper received April 25, 2018; revised August 8, 2018; accepted September 10, 2018. Date of publication December 25, 2018. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Miroslav Lutovac.

This paper is a revised and expanded version of the paper presented at the 25th Telecommunications Forum TELFOR 2017 [22].

This work was supported by the Russian Fund for Basic Research (RFBR), Grants #18- 07-00989a, #18-47-732012 р_мк and the State Contract №2.1534.2017/4.6.

P. Sosnin is with the Ulyanovsk state technical university, Russia (phone: 007-951-0954466; e-mail:sosnin@ulstu.ru.

presented in figure 1 (inherited from [4]), where OS(t) corresponds to the collaborative designing of systems with software. On the scheme, the reflection (***R^{QA}***) shows that all that is involved in the designing of the system ***S*** is found their expression as models in the QA-memory of the WIQA.

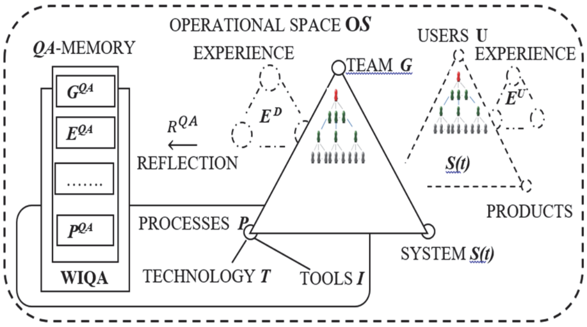


Fig. 1. Reflection of OS(t) on QA-memory.

Such reflection can be described by the following expression:

$$\boldsymbol{OS(P, K, E^D, E^U, S_i(t), \dots, X, t) \xrightarrow{R^{QA}} CS(K^{QA}(t), E^{QA}(t), P^{QA}(t), S_i^{QA}(t), \dots, X^{QA}(t)),} \quad \textbf{(1)}$$

where X indicates on any essence that is absent in the scheme, and other symbolic designations correspond to the names of essences in figure 1. Let us additionally note that the results of the reflection ***R^{QA}*** are dynamic objects that complete the conceptual space in its current state CS(t).

When a combination of verbally registered questions and answers describes an object in the certain space, we register such a combination as QA-object in the cell (or cells) of QA-memory. Any QA-object integrates verbal 'traces' of interactions with the experience that were used in perception of the corresponding object in the certain space.

In the described case, the main feature of QA-objects is their placement in the QA-memory.

Such objects can be simple if each of them is uploaded in one cell, for example, for representations of questions of different types (Q-nodes), including questions like a task (Z-nodes), and answers (A-nodes) to questions. When an interactive object is located in a set of the bound cells, it is a complicated QA-object. With each QA-object is associated not only its representation in the semantic memory but also the system of operations, including operations for visualizing the objects. Any QA-object or their comprehended combination we interpret as an example of QA-nets.

We began our study of QA-nets by considering the design process from the viewpoint of solving the project tasks that was presented by the tree of tasks. Furthermore, with any task, we bound a protocol, in which we registered question-answer reasoning used in the process of the task solution. The initial intention of the study was to apply the tree of tasks and subordinated QA-protocol for estimation the current state of the project for managing the collaborative work of designers. Later, such a type of artifacts was adopted for use in activities shown in figure 2. In this scheme, one can view a visual tree of nodes, any of which has X or Y-type but nodes can have not only hierarchical relations. The nodes can be bound by common additional attributes modeling useful semantic relations. So, in a general case, the structure can have the form of the net.

For example, in conceptual space CS(t), designers have the possibility to interact with the following types of nodes: Q – question and A – answer in QA-reasoning; G – team of designers and M – member of the group; R – executed role and C- competency; D – name of data andV – its value and so on. In the current state, if the CS(t), all its QA-nets are intertwined, forming the artificial environment, the use of which facilitates including some positive effects in the design process.

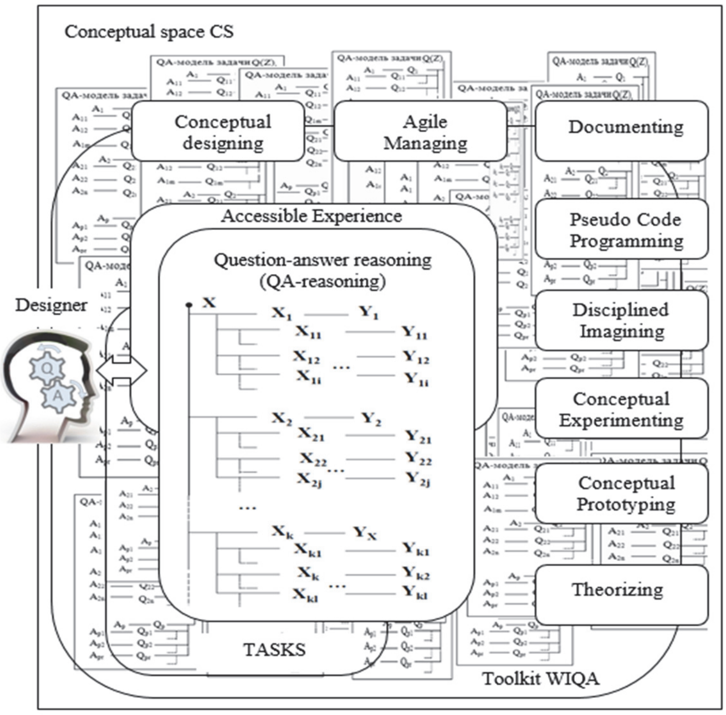


Fig. 2. Basic applications of QA-nets.

III. Related Works

Let us view some thematic directions of related works, the first of which is bound with understanding and objectifying of spaces in the domain of design. As told above, in the most general case, space is an essence or construction that consists of objects of different kinds and relations among them, including dynamic relations. The important feature of any space is having the regularities manifest themselves in actions in this space as surrounding of people or as surrounding of the certain system or systems. In this respect, any space has boundaries that limit the area of interest of a particular activity.

In practice of designing the system with software, it actively uses the phrase "design space" that is far from new. In the report [6], this phrase points out "*a multi-dimensional design space that classifies system architectures. Each dimension of a design space describes variation in one system characteristic or design choice.*" A very interesting interpretation of the design space is specified in the dissertation [7], where this artifact was disclosed from the viewpoint of "*the construction, exploration, and expansion of a conceptual space.*"

A similar position, considering the design space "*as a conceptual space, which encompasses the creativity constraints that govern what the outcome of the design process might (and might not) be*," is presented in the paper [8]. Authors of this paper interpret the design space as a dynamic artifact that is developed and changed by designers by accessible and mastered knowledge and experience.

Additionally, we mark the paper [9], where the design space is interpreted "*as a conceptual tool that can be used both for designing and understanding design processes*." Here, the author underlines that "from all work done during the design process, designers construct knowledge and experience of the design space" in the constructive form.

The second direction focuses on versions of existing or objectifying the design space that is useful for considering in the design process. Often enough, spaceis explicitly or implicitly used for referencing on physical spaces in which design activities are implemented, for example in design studios or labs [10].

In the theory and practice of design, it is widely used as a metaphorical landscape of design opportunities, for example in papers [11]. However, it is more popular interpretations of the design space as artifacts that belong to the class of conceptual spaces. This class of spaces includes their versions with some other applications. Disclosing these versions, we start with the paper [12] underlining the role of the CS(t) in creative activity "*someone who seeks to understand what creativity is, and how it is even possible, needs to consider the mental geography of conceptual spaces.*"

One type of geography focuses on "Conceptual space as a Geometry of Thought" proposed by P. Gärdenfors [13]. This type of spaces can be formalized with the use of metrics and useful ordering of their objects. Any object of the CS(t) is presented as the certain domain with characteristics of object quality. These features are used for logical views on the CS(t) [14], its theoretical descriptions [15] and thorough formalization [16]. We also mark the paper [17] where the design is understood and formalized with the use of three worlds (spaces) ─ the external world, the interpreted world, and the expected world, in which designers apply Function-Behavior-Structure framework (FBS-framework, FBS-model). In the paper [18], there is a suggestion for the answer to the question "How a function is transformed into behavior?"

All papers indicated in this section were used as sources of requirements in developing the set of instrumental means providing the proposed approach to modeling in our version of conceptual spaces.

IV. Specificity of QA-space

A. Features of becoming the conceptual space

In the proposed version of CS(t), any of such spaces is a unique construction because it is built for the certain project in parallel with other actions of designers. Therefore, creating the space must be ordered, and for that, we have embedded the mechanisms of automated design thinking in the WIQA. Such a kind of thinking and corresponding workflows are initiated by any new task that has unpredictably arisen in front of designers. The first of these tasks is usually the root task of the innovative project. It can be said, the becoming of the space CS(t) is managed by new tasks, for the work with which the designers apply automated design thinking. Conceptual space is a space for think [3].

Let us clarify some moments of becoming the CS(t). In the general case, designing the certain system begins with an innovative intention expressed (for example) by a list of essential keywords. At the first stage of design thinking (the Empathize stage), looking through keywords, designers create some discourses disclosing the essential expectations of the potential users. These discourses contain information about important requirements, QA-analysis of which helps the designer to formulate the initial statement St(Z*, t2) of the task Z* (The Define stage). Analyzing the text St(Z*, t2), the designer tries to invent the idea that can lead to the appropriate solution (the Ideate stage), for estimating of which the corresponding prototype should be built (the Prototype stage) and tested (the Test stage).

The described way of working has an iterative character with possible returns to previous steps. Moreover, the first solution is seldom effective, and the search for alternative solutions should be repeated for the future choice of one of them.

B. Example with the project "Base of Precedents"

The WIQA toolkit consists of subsystems, one of which is a subsystem "Base of Precedents." So there was a time when we started the project of this subsystem. On an example of the project, we clarify some elements of our version of design thinking. We begin clarifications with the following discourse:

D1. When operating the system with the built-in complex "Base of Precedents" (?) in the event of an unpredictable situation (?) for both the designers and users, there should be an opportunity for an intelligent reaction (?) aimed at handling (?) the situation for continued actions.

The reason of formulating this discourse is caused by the real practice of developing the modern systems with software that demonstrate unpredictable influences of numerous situational factors both as on the process of designing so on the process of exploiting.

Some points in the text D1include the sign "?" indicated on directions of question-answer analysis. Among units of the conducted analysis we mark the following of them:

Q1. What is meant by intellectual reactions of a designer or a user?

A1. With a new intellectual reaction, it is rational to bind the search of a new occupational action that corresponds to a way of solving a new task (?), conditions of which correspond to unpredictably arisen situation.

Answer A1 is better to interpret as the requirement that is habitual for the designer, but for the user, it is an intellectual function embedded into the system to be developed.

Q2.How to simulate the similar reactions?

A2. Since the reaction is recommended to build in the form of finding a solution to a new task, it is rational to represent it in the form of a precedent model (?) taking into account analogies (?) with intellectually processed (?) conditioned reflexes (with the conditioned reflexes that have the kind of human actions).

In this answer, we are oriented on the following definition "precedents are actions or decisions that have already happened in the past and which can be referred to and justified as an example that can be followed when the similar situation arises" [19].

The answer A2 has also related the question "What to do with the built precedent model?" For the user, there are the following versions of the reaction to the question:

- Transmitting the model of the precedent to specialists who use it for evolving the system functionality;
- Developing the precedent and embedding it into the system without intermediaries if the user has a sufficient qualification.

Other marked signs of questions in D1 and answers A1 an A2 indicate ways of continuing the question-answer analysis of the discourse D1. Among next question-answer pairs, we mark the following refinements:

- Creating the next precedent model, it needs to implement in parallel with solving the corresponding new task, beginning the processes with the use of automated design thinking;
- Applied design thinking must include the functionalities that can help the designer in conducting the thought experiments with the use of mental imagery.

In the project of the subsystem "Base of Precedent" there was a time when we have formulated the following root task:

Z(t): It needs to develop a subsystem "Base of Precedent", operative creating (?) and using (?) of which will facilitate (?) enhancing(?) the intellectual effectiveness (?) of the designer' activity in working (?) with unpredictable project tasks in designing the systems with software that will include mechanisms (?) for similar intellectual reacting (?) for users in unpredictable situations (?) arisen in exploiting of such systems.*

Here, questions marks also indicate ways for continuing the analysis in the frame of the proposed version of automated design thinking, outcomes of which were used in the development of "Base of Precedent."

C. Activity in conceptual space

When the described analysis is applied by designers in work with some project P, all obtained results, including the tested prototype in the form of pseudo-code,will define the initial version of the conceptual solution of the root task Z*(t) in its current state registered in the CS(t).Interactions of one of the designers with objects of artifacts Z*(t) and CS(t) via nodes of QA-nets are schematically shown in figure 3.

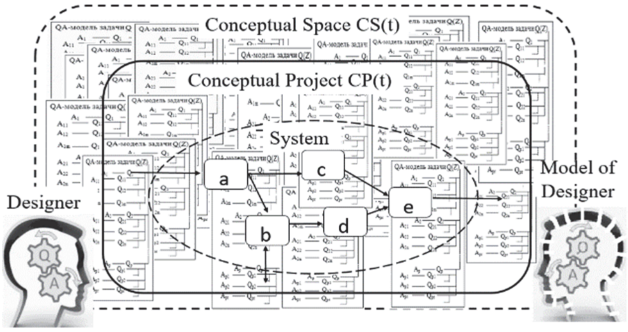


Fig. 3. Interactions of the designer with QA-nets and their nodes.

The scheme underlines, the conceptual project CP(t) are extracted from the CS(t), and components (for example, a, b, c, d, and e) of this extraction are also objectified in forms of QA-nets. By the system approach, components of the CP(t) can be interpreted as subsystems located in CS(t) and so on.

In our version of the artifact CS(t), the designer begins to create this space with the "empty state," starting with the moment of time t_0. As it is shown in figure 4, the designer is working in the CS-environment that is evolved on the course of design thinking [21]. Space is opened for interacting with an embedded object, their groups, and relations. Among these relations, there are cause-and-effect regularities of the space.

Thus, the CS(t) has a potential for checking the adequacy of verbal expressions and even for correctly generating them or for extracting the expressions from the current state of the CS(t). The use of the artifact helps the designer in achieving the sufficient degree of the personified conviction in estimating the adequacy of definitions and descriptions and in their understanding.

Building the conceptual solution of the root task Z* of any project of the SISs, we interpret as the first step in the development of the corresponding CS(t). Other steps of its developing are caused by conceptual solving the other project tasks, for the work with which designers should use the described mechanisms of (automated) design thinking.

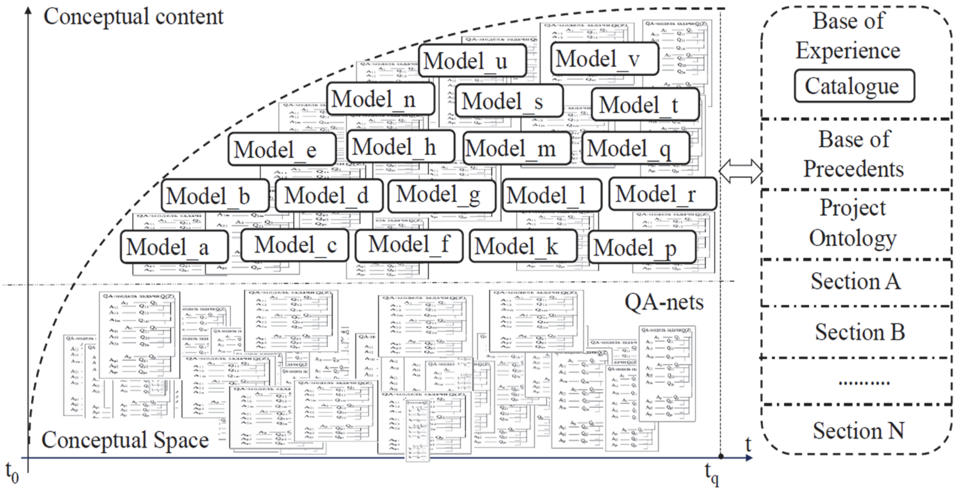


Fig. 4. Beginning the results of conceptual solving the tasks.

The structure and content of the scheme demonstrate the life cycle of the CS(t) in conditions when the designers apply QA-approach and precedent-oriented approach [4] in conceptual solving of the project tasks. In applying the precedent-oriented approach to the certain project task, the designer creates the corresponding model of the precedent, preparing the task solution for the reuse. Such models, we interpret as models of experience that are generated in the design process and accumulated in the Base of Precedents. Any of such models is a regularity in the developed CS(t).

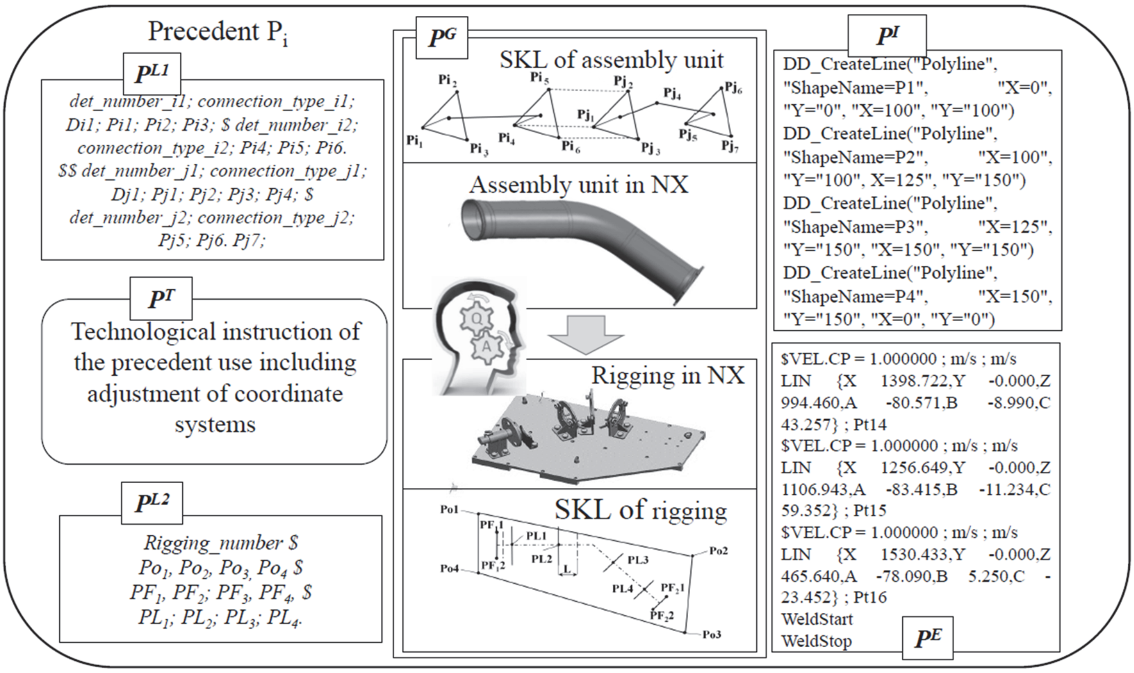
D. Geometric modeling in manufacturing the aircraft pipelines

In the general case, some parts of the artifact CS(t) can be realized as subordinated spaces. For example, in our current project of the technological preparation (TP) of producing the aircraft pipelines, we use precedent-oriented geometrical modeling of the assembly units. The basic feature of modeling is the use of the master geometry (MG) of the aircraft.

In the frame of the MG, for engineers, it is possible to simulate the pre-installation of aircraft subsystems and various pieces of aircraft equipment. During such work, step by step, the MG-space is filled by components of aircraft sub-systems.

The MG also helps technologists in developing the processes for manufacturing the aircraft components, and it also enables to assess how the maintenance operations can be performed.

In welding the parts of any assemble unit, it needs to provide the correct positioning of the welded parts of the metal pipelines in accordance with their positioning in the MG. For providing the necessary positioning, we have decided to use skeletal models (SKL) of parts, assembly units and appropriate rigging, P_i. For the reuse, corresponding models are combined in precedent forms and placed in the Base of Precedents as units one of which is shown in figure 5.


Precedent P_i
P^{L1}
det_number_i1; connection_type_i1; Di1; Pi1; Pi2; Pi3; $ det_number_i2; connection_type_i2; Pi4; Pi5; Pi6. $$ det_number_j1; connection_type_j1; Dj1; Pj1; Pj2; Pj3; Pj4; $ det_number_j2; connection_type_j2; Pj5; Pj6. Pj7;
P^T
P^{L2}
Rigging_number $
Po_1, Po_2, Po_3, Po_4 $
PF_1, PF_2; PF_3, PF_4, $
PL_1; PL_2; PL_3; PL_4.
P^G
SKL of assembly unit
Pi_1
Pi_2
Pi_3
Pi_4
Pi_5
Pi_6
Pj_1
Pj_2
Pj_3
Pj_4
Pj_5
Pj_6
Pj_7
Assembly unit in NX
Q
A
Rigging in NX
SKL of rigging
Po1
Po2
Po3
Po4
PF₁1
PF₁2
PF₂1
PF₂2
PL1
PL2
PL3
PL4
L
P^I
P^E

Technological instruction of the precedent use including adjustment of coordinate systems

```
DD_CreateLine("Polyline",
"ShapeName=P1",          "X=0",
"Y="0", "X=100", "Y="100")
DD_CreateLine("Polyline",
"ShapeName=P2",        "X=100",
"Y="100", X=125", "Y="150")
DD_CreateLine("Polyline",
"ShapeName=P3",        "X=125",
"Y="150", "X=150", "Y="150")
DD_CreateLine("Polyline",
"ShapeName=P4",        "X=150",
"Y="150", "X=0", "Y="0")
```

```
$VEL.CP = 1.000000 ; m/s ; m/s
LIN   {X   1398.722,Y   -0.000,Z
994.460,A    -80.571,B    -8.990,C
43.257} ; Pt14
$VEL.CP = 1.000000 ; m/s ; m/s
LIN   {X   1256.649,Y   -0.000,Z
1106.943,A   -83.415,B   -11.234,C
59.352} ; Pt15
$VEL.CP = 1.000000 ; m/s ; m/s
LIN   {X   1530.433,Y   -0.000,Z
465.640,A   -78.090,B   5.250,C   -
23.452} ; Pt16
WeldStart
WeldStop
```

Fig. 5. Model of precedent.

The shown precedent model integrates the textual instruction P^T, logical specifications P^{L1} andP^{L2}, graphical presentations for assembly unit and corresponding rigging and also pseudo code of SKL and CNC-program of robotic welding.

Any such precedent model registers the certain CS-regularity built with the use of prototyping (experimental actions). Moreover, these regularities are extracted from the CS during targeted work that includes conceptual experimenting and theorizing [21] in the environment of the WIQA toolkit where regularities are combined in the Base of Precedents. This kind of artifact was embedded in a number of SISs among which we mark the SIS for the collision avoidance of the sea vessels, and the cartographical model of the space was used in this SIS [4].

It is the principal feature of the proposed approach to creating and using the CS in designing the systems with software. That is why this kind of the CS we specify as "experience-oriented."

V.Conclusion

In this paper, we described the constructive version of objectifying the conceptual space oriented on designing the systems with software. The basic feature of the proposed version is the use of QA-nets simulating the basic essences involved in the design process.

Our work with QA-nets in conceptual designing of the SISs, including the development of the WIQA toolkit and its extensions, confirmed that such a kind of artifacts helps to build the CS that adequately reflects the operational space of designing.

Till an awareness that QA-nets embedded into a certain project can be combined in the CS, these nets have been applied in designing of a number of software intensive systems [4} in which conceptual space was used no more than metaphorically. Awareness came only at the beginning of this year, and the first constructive application of the CS is being objectified in the project mentioned in the previous subsection. Currently, the proposed version of CS is materialized in the WIQA toolkit.

When the state of the CS is applied for conceptual designing of a certain SIS, this toolkit provides the following possibilities:

1. To build conceptual solutions of the project tasks in the frame of the CS and in parallel with its becoming and controlled evolving.
2. To build and keep the alternative solutions (the use of the space as a source of alternatives).
3. To discover and define CS-regularities that are useful for the corresponding project and will be embedded into the system to be designed.
4. To create such basic components of the CS as the Base of Experience including the Base of Precedents;
5. To form the basis for creating the project theory (as applied theory of the CS) that leads to traditional positive effects from theorizing.
6. To apply the current state of the CS for preventing the semantic errors, their discovering and correcting, first of all, in controlled achieving of the necessary degree of understanding and registering it for the reuse form.

This space is opened for the combinatorial, investigational and transformational actions of designers, who creatively solve the necessary tasks. They can fulfill such work in the mode of behavioral programming.

References

[1] Chaos reports 1994-2016. Available at http://www.standishgroup.com,(accessed in 15 April 2018).

[2] P.Clarke and R.V.O'Connor, "The situational factors that affect the software development process: Towards a comprehensive reference framework,"*Journal of Information Software and Technology*, vol. 54, no. 5,pp. 433-447, 2012.

[3] R. M. Young, "Mental Space,"*Process Press*, 1994.

[4] P. Sosnin, "Experience-Based Human-Computer Interactions: Emerging Research and Opportunities", IGI-Global, 2017.

[5] IBM Rational Unified Process, Available online: http://www-01.ibm.com/software/rational/rup/ (accessed 20th February 2018).

[6] T. G. Lane, "A Design Space and Design Rules for User Interface Software Architecture," Technical Report CMU/SEI-90-TR-22 ESD-90-TR-223, Software Engineering Institute - Carnegie Mellon University, 1990.

[7] C. Heape, "The Design Space: The Design Process as the Construction, Exploration and Expansion of a Conceptual Space."Ph.D. diss. *The University of Southern Denmark, Sønderborg*, Denmark, 2007.

[8] E. Kang, E. Jackson, and W. Schulte, "An Approach for Effective Design Space Exploration." In: Calinescu R., Jackson E. (eds) Foundations of Computer Software. Modeling, Development, and Verification of Adaptive Systems. *Lecture Notes in Computer Science*, vol. 6662, pp.33-54, 2010.

[9] B. Westerlund, "Design Space Conceptual Tool – Grasping the Design Process,"in *Proc. of the Nordic Design Research Conference*, pp. , 2005.

[10] Lucero, A., Vaajakallio, K., and Dalsgaard, P. "The dialogue-labs method: Process, space and materials as structuring elements to spark dialogue in co-design events*." CoDesign*vol. 8, no. 1, 1-23, 2012.

[11] M. A. Boden. "Conceptual Spaces," P. Meusburger et al., Eds., Milieus of Creativity, Knowledge and Space 2, Springer Science + Business Media B.V., pp.235-243, 2009.

[12] W. Gaver,"Making spaces: how designworkbooks work,"*in Proc. of the SIGCHIConference on Human Factors in Computing Systems*(CHI '11) 1551-1560, 2011.

[13] P. Gärdenfors, "Conceptual Spaces −The Geometry of Thought,"*The MIT Press*, Cambridge, Massachusetts, 2000.

[14] P. Gärdenfors, "Semantics based on conceptual spaces. Logic and its applications," M. Banerjee and A. Seth (Eds.), Berlin, Heidelberg: Springer. *Lecture Notes in Computer Science*, 6521, pp. 1–11, 2011.

[15] J. T. Rickard,J. Aisbett, and G. Gibbon,"Reformulation of the theory of conceptual spaces," Inf. *Sci*., 177 (21), (2007), pp. 4539-4565.

[16] L.Bechberger and, K.-U. Kühnberger, "A Thorough Formalization of Conceptual Spaces," Advances in Artificial Intelligence, *Lecture Notes in Computer Science*, 10505, pp.58-71, 2017.

[17] J.S. Gero and N. Kannengiesser, "The Situated Function-Behavior-Structure Framework,"*Artificial Intelligence in Design*,vol. 25, no.4, pp. 373-391, 2004.

[18] S. Al-Fedaghi, "Function-Behavior-Structure Model of Design: An Alternative Approach,"*International Journal of Advanced Computer Science and Applications*, vol. 7, no. 7, pp. 133-139, 2016.

[19] Precedent, Available at http://dictionary. reference.com/ browse/precedent

[20] K.J. Stol, P. Ralph, and B. Fitzgera, "Grounded Theory in Software Engineering Research: A Critical Review and Guidelines,"in Proc.of the 38th International Conference on Software Engineering, pp. 120-131, 2016.

[21] P. Sosnin, "An Application of Design Thinking in a Creation of a Project Theory Supporting a Conceptual Design of a Software Intensive System,"in *Proc. of the 11th IEEE International Conference on Application of Information and Communication Technologies*, pp. 376-380, 2017.

[22] P. Sosnin, "Question-answer nets as a valuable source of information in designing the software intensive system," *2017 25th Telecommunication Forum (TELFOR)*, Belgrade, 2017,