# THE USE OF SITUATION REPRESENTATION WHEN SEARCHING FOR SOLUTIONS IN COMPUTER AIDED DESIGN SYSTEMS

**Alexander Kuzemin, Mikhail Sorochan, Igor Yanchevskiy, and Asanbek Torojev**

*Abstract: Projects solutions reuse methodology is offered for software development. The main idea consists in connection of the system objective with the situation using the entities which describe the condition of the system in the process of the objective statement. Every situation is associated with one or several design solutions, which can be used at the development. Based on this connection the situation representing language has been created, it lets to express a problem situation using a natural language describe. The similarity measure has been built to compare situations, it is based on the similarity coefficients with adding the absent part weight.*

## Introduction

Business software development for modern organizations is a severe and difficult process. The latest researches in software companies show that more than a half of all projects exceed the time and resource budgets [1]. One of the commonly occurring tasks in computer aided design systems is an assembling of the object being designed using complete components or similar ones after some modification [2]. Any engineering project including a software one, which solves new problems by existing solutions reuse, has great possibilities to reduce resource and time costs to develop a new project [3].

But in component reuse the problem of finding the proper project solution matching requirements established to a desirable component arises. Industrial level design methodologies (RUP, UML, SADT etc.) have no embedded guidance for solutions reuse, their main goal is the design models visualization.

## The Goal of the Work

The main goal of this research is to increase productivity in the computer aided software engineering systems by reuse of the solutions obtained before. Not only the completed program modules and components can be reused, but algorithms, diagrams and other project artifacts [4, 5]. Solutions reuse is effective only when the taken to search the existing solution than that spent for creating a new one. Therefore, when developing the reuse methodology the main attention should be paid to the algorithms of the project solutions search.

## Statement of the Problem

To reuse design solutions effectively one has to solve two main tasks: the current solution classification in the knowledge base and search for the most suitable solution for a particular task. This can be done by developing a unified mechanism of describing solutions' parameters to compare them with the current task being solved by designer. Such descriptions are offered to shape in the form of situations, which consist of the natural language constructions describing the system state and its environment.

## The Main Body

Modeling an organization problem domain a designer builds a system model which is a reflection in an observer mind of objects attributes and their relations in the research problem solution.

The $Sm$ system as a reflection of the target organization for which the software is developed can be presented by a designer as [6]:

$$Sm = < A, R, Z, P, G >, \quad (1)$$

where:

$A$ - elements (components) of the $Sm$ system,

$R$ - relations of the system elements,

$Z$ - the system goals,

$P$ - the system designer,

$G$ - the designer language (the chosen modeling methodology)

The goal of the software being developed to meet the $Z$ goals is satisfaction of system $Sm$. In the initial stages of the design process a designer $P$ using different language means $G$ carries out research on the target organization to clarify the relationship of $A$ elements connected in the form of $R$ to gain $Z$ goal.

Environment together with the system concerned in it can be modeled in different ways. Traditional methods which describe the object as a set of mathematical formulas do not give the idea on the object being described by the given equation.

The main problem in data gathering by analysts from the problem domain experts is associated with different language entities operating with and the need in a common communication language [7].

In [7] the author puts forward two hypotheses concerning the presentation of information on the objects. The first one says that all information about the object under control can be presented by the common natural language means. The second one says that any text which describes the object under control may be translated into a formal language. There is no doubt that there may exist objects for which the mentioned hypotheses fail but experience has proved their consistency.

Considering the solution of some problem or achieving some goal one may say that there exist some initial conditions of the task. This information describes the condition of the modeled object and its environment in the moment when the need appears the need to solve the set appears. Such a state can be effectively presented by the situations presented in the [8].

The situation consists of entities sets. To find a design solution appropriate to the current situation one needs to compare the current situation with a set of etalon situations. Before the situation building it is necessary to identify entities composing the situation. Conventional classification methods such as the cluster analysis do not match here because it is unknown beforehand what is entity and how its attributes are correlated with etalon entity attributes. That's why before the situation description it is required to develop and guide some additional operations on the entities identification.

To identify correctly the entities belonging to the current situation description a categorical classification algorithm can be used. This algorithm lets to correlate effectively unknown entity with the known class. It is implied that there exist categories groups of the entity classes. These categories are built as hierarchies with subsumption relations.

When modeling the organization problem domain the designer finds himself in a certain situation which is defined by the group of the interrelated entities presenting different organization features and describing the current situation. An interrelation with some entity means that the entity exists in the current situation and in some way takes part in the action, it is a rule or deliverable, it acts as an observer or tool, it has some special feature etc. That is why the organization presentation may be considered as a system consisting of a set of entities which are interrelated between each other and form integrity and organic unity.

Having found himself in one of such situations a designer may have a need to create one or some design diagrams, require some guidance or need some other project artifact. The similarity of such situations is confirmed by many researches [9]. Therefore, the designer encountering meeting such situations may operate common design solutions too. Every time meeting the situation known before and for which there exists a group of design solutions the designer may use the solutions associated with the situation.

The situation contains sets of entities each of them reflects its qualities in any characteristic category. Such categories can present processes and activities, resources and goods, organization structure, deliverable structure and requirements, cost and so on. An entity set separated from entities' interrelations can hardly describe a situation because those entities are related to each other, they take part in variety of processes, appear and disappear, create new entities and cooperate between each other.

It is offered to divide categories into common main groups reflecting the main aspects of structural and dynamic organization presentation. Each of those groups will consist of categories sets presenting entities of the group.

Category $C$ is presented as an hierarchy by the oriented graph of entities $C =< O, A >$ which is build by the subsumption indication where $O$ is a set of graph nodes presenting entities $\{e_1, e_2, ..., e_n\}$ and $A = \{G_{1,} G_{2,} ..., G_{n-1}\}$ is a set of graph arcs representing generalization relation between two entities directed from a successor to an ancestor. Entity $e_1$ is a primary entity of category $C$ which represent the most common entity in a hierarchy.

Every entity includes a pointer to the decision rule which defines a set of features distinguishing this entity from others.

The decision rule is presented as:

$$DR = \underset{i}{\wedge} F_i \, , \qquad (2)$$

where $F_i$ is a separate feature.

To calculate the value of the decision rule $DR$ each feature $F_i$ gets the truth value if this entity has the feature otherwise it gets a false value.

The decision rule defines a group of features making it possible to distinguish the given entity from other ones. A feature is something that characterizes the entity defining its likeness or difference in a variety of entities. Feature may present both presence values (positive feature) and absence values (negative feature).

In a variety of features one selects the most typical and distinctive for this kind of entities. Therefore in entities formation abstracting and idealization are required. As a rule it is called an entity essential features allocation. It should be remembered that allocation of exactly essential features is not always performed in practice and even in some cases is unrealizable. The essence of objects, processes, relations and events is defined by science. In this case it is enough to point to the features which distinguish the discussed entities from others to avoid confusion. Along with essential features it is possible to use some features sufficient for distinction; they let to define an entity sufficiently and unambiguously not to confuse the entity with others though these features may be inessential for this entity. In practice such features are used even widely. In this work attention is paid to the essential features use. It is considered that it is sufficient to point on features unambiguously distinguishing some entity from others.

Each entity must contain only one semantic value which concerns only one essence. This makes it possible to avoid ambiguity using this entity in situations.

An unknown entity is classified by the entity selection strategy, for the category is $C =< O, A >$ it is presented as :

$$S_C =< P, D, p_{0,} D_0 > , \qquad (3)$$

where $P = \{p_{1,} p_{2,} ..., p_n\}$ is a points set, $D = \{D_i\}, i = \overline{1, n}$ is a set of transitions to every point $p_i \in P$, references to primary point $p_0 \in P$ from which the identification begins, reference to the primary transition $D_0 \in D$ which assists to make decision whether the entity belongs to the category or not.

In transition $D_k = \{d_{j,k}\}$ parameters $j = \overline{1, n}, k = \overline{1, n}$ mean that the transition is directed from the point $p_j$ to the point $p_k$ or in other words from the entity class $e_j$ to the entity class $e_k$ .

The transition $D_k$ from some entity class to the entity class $e_k$ corresponds the decision rule $DR_k$ .

The entity identification strategy in a category is not a separate set but only a semantic union over entities hierarchy in the category and their decision rules. Such separate representation allocation makes it possible to consider the strategy in algorithms as a single element. The strategy is not a static set. It is build depending on the current structure of the category and entities it consist of.

Thus, it is obvious that the possibility of transition to some point is defined only by decision rule connected with corresponding entity. $D_k = \{d_{i,k}\}$ transitions direction is backwards to arcs directions from the set

$A = \{G_1, G_2, ..., G_n\}$ . Every $G_j(e_i, e_k)$ arc corresponds to the set of $D_i$ transitions for the pair of points $p_k$ and $p_i$ .

Every strategy begins with the primary transition $D_0$ . Passing through the transition means that the unknown entity may belong to this category. The primary transition leads to the primary point $p_0$ corresponding to the entity $e_0$ of the category $C$ .

The situation is formed as a set of triples – micrositutations. The situation $S$ is presented as an aggregate of $n$ microsituations $s_i$ [8]:

$$S = \{s_i\}, i = \overline{i, n}, s_i =< e_i, R_{e_i} >, \quad (4)$$

The part of the situation described by the pair $s =< e, R_e >$ is called a microsituation of this entity $e$ .

The entity $e$ is called the central entity of a microsituation $< e, R_e >$ , set $R_e$ is a context of the central entity $e$ . An entity is called the central one when it is a central semantic unit where this microsituation is based. Set $R_e = \{r_i\}, i = \overline{1, m}$ consists of entities expressing the relation of the central entity $e$ with other secondary entities participating in the microsituation.

The microsituation corresponds to the triple *subject – control action – object* a human mind is operating with [10]. The subject is the central entity, the control action is the context and object is the secondary entity. In a general case the sequence *subject – control action – subject – control action … object* [10] corresponds to the case when the secondary entity from one microsituation is the central entity in the other microsituation.

To compare microsituation let us define the similarity measure expressing the distance between situations. Its calculation is divided into calculation of the distances between microsituations and deeper between entities.

The general principle of calculation of the distance between two entities, microsituations and situations consists in separation of the common part of the objects being compared and calculation of how each of them differ from the common part. The idea of distance calculation is depicted on figure 1.

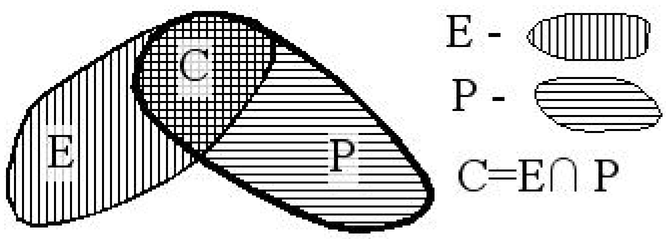


Fig. 1 – Correspondence between two sets

Comparison between two sets etalon $E$ and problematic $P$ is exemplified. It is also known that this sets have a common part $C = E \cap P$ . Let us consider that etalon set $E$ is linked with some solution $D$ and it is necessary to calculate the similarity between the sets $E$ and $P$ to decide whether the solution $D$ can be used with the set $P$ . The similarity degree $Sy$ is defined by a part of set $C$ in every of comparable sets. But here it is necessary to take into account that solution $D$ is connected with elements of the set $E$ and in the case of $C \subset E$ that solution is leaned on the absent part $E / C$ that is missing from the set $P$ . This decreases the applicability degree of the solution $D$ for problematic set. At the same time the set $C \subset P$ defines the part of problematic set $P$ for which this solution $D$ is applicable. Therefore, the similarity degree of the etalon and problematic sets

$Sy(E,P)$ in the common case is a part of the set $C$ in the union $E \cap P$. But here the additional weight of absent part $E/C$ weakening $Sy$ must be also considered.

Let define the set $A$ cardinal number as $\mu(A)$. Starting from Tversky's contrast model [11] and relying on the association coefficients (Zhakkar) and similarity (Roberts and Tanimoto), taking in account the remark on the absent part the similarity degree can be presented as:

$$Sy(E,P) = \frac{\mu(C)}{\mu(C) + \mu(P/C) + \frac{\mu(E)}{\mu(C)} \cdot \mu(E/C)}, \quad (5)$$

where $\mu(E)/\mu(C)$ is the absent part strengthening coefficient which is directly proportional to the part of the set $E/C$ in the set $E$.

Thus, calculated in such a way the similarity $Sy$ is dissymetric $\forall A \neq B, Sy(A,B) \neq Sy(B,A)$ and therefore cannot be a metric, but in the context of the considered problem it is an appropriate one.

The absent part strengthening coefficient is present in all further equations for distance calculation for situations, microsituations and entities. Similarity measure is expressed by the interval $[0,1]$. The same situations have the similarity equal to unity and for absolutely different situations the similarity is equal to zero.

Considering the situation $S$ as a set of $n$ microsituations $\{s_1, s_2, ..., s_n\}$, the distance between etalon $S_1$ and problematic situation $S_2$ will be presented as:

$$L(S_1, S_2) = \frac{1}{n} \begin{cases} \max\limits_{a_j} \sum\limits_{i=1}^{n} L(s_i, s_{d_i}), d_i \in a_j, a_j \subset T_m^n, npu\, n \leq m \\ \max\limits_{a_j} \sum\limits_{i=1}^{m} L(s_{d_i}, s_i), d_i \in a_j, a_j \subset T_n^m, npu\, m < n \end{cases}, \quad (6)$$

where $n$ and $m$ is a quantity of microsituations for the situations $S_1$ and $S_2$, respectively; $L(s_1, s_2)$ is a distance between microsituations $s_1 \in S_1$ and $s_2 \in S_2$; $T_n^m$ is a distribution of $A_n^m$ groups of microsituations' numbers of $n$ elements by $m$, $T_n^m = \{a_j\}, j = \overline{1, A_n^m}$; every element $a_j$ consists of the set of microsituations numbers $a_j = \{d_1, d_2, ..., d_m\}, d_k \in N$.

The distance between microsituations is presented in the form of the following product:

$$L(s_1, s_2) = L(e_1, e_2) \cdot L(r_1, r_2) \cdot \frac{1}{n} \cdot \begin{cases} \max\limits_{a_j} \sum\limits_{i=1}^{n} L(e_i, e_{d_i}), d_i \in a_j, a_j \subset T_m^n, npu\, n \leq m \\ \max\limits_{a_j} \sum\limits_{i=1}^{m} L(e_{d_i}, e_i), d_i \in a_j, a_j \subset T_n^m, npu\, m < n \end{cases}, \quad (7)$$

where $e_1$ and $e_2$ are the central entities of microsituations $s_1$ and $s_2$, respectively; $r_1$ and $r_2$ are the entities of microsituations $s_1$ and $s_2$ representing also features and relations; $e_i$ and $e_j$ are the secondary entities; $n$ and $m$ are amounts of entities taking part in relations $r_1$ and $r_2$; $T_n^m$ is a distribution of $A_n^m$ groups of entities numbers of $n$ elements by $m$, every element $a_j$ consists of the set of microsituations numbers $a_j = \{d_1, d_2, ..., d_m\}, d_k \in N$; $L(a,b)$ is a distance between entities $a$ and $b$.

The etalon entity is an entity situated in the concerned category. The problematic entity is an entity taking part in the current situation description. The distance between the comparable entities is defined as a degree of feature correspondence with the nearest paternal entity (NPE) and its remoteness from NPE. The distance between etalon $e$ and problematic $p$ entities is presented as follows:

$$L(e,p)=\frac{D_a}{D_a+(P_a-D_a)+\frac{E_a}{D_a}\cdot(E_a-D_a)}, \quad (8)$$

where $D_a$ is a quantity of NPE features, $E_a$ is a quantity of etalon entity features, $P_a$ is a quantity of problematic entity features.

Design solutions search consists of the following stages:

Current situation description as a narrative text.

Revealing of the entities in the available categories from the obtained description.

Construction of the description in the situation representing language.

Search for the etalon situation close in its structure to problematic one within the defined similarity interval

Narrowing of the search by including entity objects attributes

Situation description looks as follows:

*situation : microsituation-group* | *entity-declaration microsituation-group* |
*entity-declaration entity-attributes microsituation-group*

Components of *entity-declaration* and *entity-attributes* are not compulsory. They may absent if there is no need to give names to same kind entities and (or) indicate values of their attributes.

Situations' description presented in the situation description language obey definite rules which let to understand the situations structure with some fixed meaning.

**Axiom 1.** The following statement holds for two microsituations $s_1$ and $s_2$ of some situation $S$. If the microsituation $s_1$ is described before the microsituation $s_2$ in the situation $S$ description, then it can be said that a time interval $\Delta t \geq 0$ passed between the microsituations $s_1$ and $s_2$.

**Theorem 1.** If the entity $e_2$ inherits the entity $e_1$ - $G(e_2,e_1)$ in some entitiess $C$ category then the entity $e_2$ can substitute the entity $e_1$ in the microsituations without loss of meaning and self-descriptiveness of the given microsituation.

**Proof.** According to the category structure and entity identification strategy to identify the entity $e_2$ it is necessary to give an affirmative answer to the decision rules $p_1, p_{k_1}, \ldots, p_{k_n}, p_2$ corresponding to the enities $e_1, e_{k_1}, \ldots, e_{k_n}, e_2$. This means that having identified the problematic entity as $e_2$ the above mentioned decision rules were passed including $p_1$, which is related to $e_1$. Therefore, the entity $e_2$ can be interpreted as the entity $e_1$ offering its characteristic features.

## Conclusions

The presented similarity measure makes it possible to search for design solutions in the knowledge base to provide the basement for engineering solution synthesis tasks in the computer aided design systems. The obtained methodology ensures increase in the designer work efficiency at the cost of using both finished and similar to the current task solutions.

The further evolution of the developed methodology is an expansion of expressive means of the situations representation language for description of casual-effect relations between entities in a situation. In some cases it may be of interest to provide microsituations and entities degree expression importance.

## References


1. *Леффингуэлл Д.*, Уидриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. - М: Издательский дом "Вильямс", 2002. - 448 с.
2. *Петренко А.И.*, Семенков О.И. Основы построения систем автоматизированного проектирования. – Киев: Высшая школа. – 1984.
3. *Park JY*, Musen MA VM-in-Protege: A Study of Software Reuse // Proceedings of MEDINFO'98. - 1998. – pp. 112-115.
4. *Соммервил И.* Инженерия программного обеспечения. - М: Издательский дом "Вильямс", 2002. - 624 с. 5. *Гамма Э.*, Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. - СПб: Питер, 2001. - 368 с.
6. *Системный* анализ в экономике и организации производства. - Л: Политехника, 1991. - 398 с.
7. *Поспелов Д.А.* Ситуационное управление: теория и практика. - М: Наука, 1986. - 288 с.
8. *Гладун В.П.* Процессы формирования новых знаний. - София: Педагог 6, 1994. - 192 с.
9. *Beznosov Konstanin*. Architecture of Information Enterprises: Problems and Perspectives // Advanced Topics in Software Engineering - 1998. - p. 25
10. *Ерофеев А.А.*, Поляков А.О. Интеллектуальные системы управления. СПб: Издательство СПбГТУ, 1999. - 247с.
11. *Tversky A.*, Similarity, separability and the triangle inequality // Psychological Review. - 1982. - №89(4). - pp. 123-154.


## Authors' Information


**Alexander Ya. Kuzemin –** Prof. of Information Department, Kharkov National University of Radio Electronics

**Mikhail V. Sorochan –** Post-graduate student of Kharkov National University of Radio Electronics, msorc@ua.fm

**Igor V.Yanchevskiy –** General Director of the Company “Computing Technique and Automation Means” (Russia)

**Asanbek A.Torojev –** General Director of the Joint Russian-Kirghiz Venture “Computing Technique and Automation Means”


# MULTI-AGENT SYSTEMS IN THE HARVEST PROGNOSIS


## Laura Ciocoiu, Cristian Paraschiv, and Dragoş Barbu



Abstract: *The paper presents a case study of geo-monitoring a region consisting in the capturing and encoding of human expertise into a knowledge-based system. As soon as the maps have been processed, the data patterns are detected using knowledge-based agents for the harvest prognosis.*



Keywords: *data mining, topological maps, GIS, knowledge based agents, Model Based Reasoning*


## Introduction

The process of geo-monitoring a region needs to use knowledge-based systems as a resource for aiding the specialists and the people in achieving their objectives. The model design process represents, in fact, the transferring of human experience in monitoring into an *interactive model.* Knowledge about process of the geo-monitoring of a region (e.g. map interpretation, statistics methods, strategies, etc) is represented by models that refer to observable features and significance.