Business Intelligence

SYSTEMOLOGICAL BUSINESS MODELLING FOR INFORMATION SYSTEMS DEVELOPMENT

Mikhail Bondarenko, Nikolay Slipchenko, Kateryna Solovyova, Olexiy Ukrayinets, Larysa Vinogradova

Abstract: One of the effective ways of the representation and analysis of the high level-purposes and business requirements to the information program system is modelling of a context of the system functioning – the organization which is automated by software creation and implementation. With this aim the domain specific language in the form of a UML profile for business modelling, where the basis is the systemological approach "Unit – Function – Object", is defined. The profile allows to create syntactically correct UML models of business and to provide simplicity of the description of the dependences between them and UML models of the program system. Profile implementation, instruments of verification of syntax a business model and its transformation into a software model is fulfilled in a CASE tool for UML modelling.

Keywords: Business Modelling, Unified Modelling Language - UML, Systemology, Systemological Approach "Unit – Function – Object" (UFO-approach), UML profile, Requirements, Model Driven Architecture - MDA.

ACM Classification Keywords: D.2 Software Engineering, D.2.1 Requirements/Specifications – Languages.

Conference: The paper is selected from Seventh International Conference on Information Research and Applications – i.Tech 2009, Varna, Bulgaria, June-July 2009

Introduction

The research done in software development companies shows that errors in defining requirements to program systems (PS) are a problem which is most often faced during software development, and the cost and complexity of their elimination is the highest and can considerably affect the budget and execution time of all the project [Leffingwell et al, 2000].

The majority of the existing requirements modelling methods do not allow considering high-level business requirements or software customers, who are not its users, but are interested in development and deployment of such a PS to solve business problems. To avoid this problem, it is necessary to start the engineering process with the description of the functional environment of the system, i.e. with business model development [Украинец, 2008 A].

Some modern methods allow to model business in the software engineering process (for example, SOMA [Graham, 2000], Rational UML profile for Business Modelling [Johnston, 2004]), however not always provide strict formal link with the system model or do not guarantee a syntactic correctness of the constructed model. Besides, frequently these methods and technologies frequently have no methodological basis that rises doubts in adequacy of the developed models to the existing business. There is no guarantee that business models reflect

the reality and give correct mechanisms of its analysis, and after all it is the main criterion of any model estimation.

«UML + UFO» – the Key to the Problem Solving

For business processes modelling, it is undoubtedly possible to apply great number of methods. As far as methods of object-oriented visual modelling which are united in unified modelling language UML, are applied virtually in all new projects program systems development, it is natural to try to use similar methods in the field of business processes modelling.

It will provide availability of the method to a wide range of non-experts, and in this case, it is very important as far as a business model should be clear for many people interested in the system and its users. One of the advantages of the given approach is the simplicity of the description of the dependences between business and program system models. It increases productivity of engineering process, reduces the number of system troubles, and also helps to make sure that the developed system solves real business requirements [Украинец, 2007 А].

The usual practice is the application of a method of Use Cases for business modelling, but this approach evokes a number of problems [Graham, 2000]. Use Cases, undoubtedly, are useful to define systems, but do not allow to describe the requirement completely. Use Cases link goals that represent roles of the users in the system with the system itself. On the other hand, requirements can be set up by people and organizations that will never appear close to the system. Ignoring of interaction with non-users can lead to loss of some important technical possibilities. That is, a special object-oriented business modelling method compatible with Use Cases models is necessary.

Thus, undoubtedly, for business modelling it is reasonable to use UML, but for this purpose standard language device is not enough. The creators of UML foresaw such a situation and have provided mechanisms to extend the language which allow defining new domain specific languages (DSL) on the basis of UML, including for business modelling. To provide strictness, entirety, the coordination of the device of modelling of business in a newly created DSL it is suggested to use systemology and systemological approach “Unit – Function – Object” (UFO-approach) [Бондаренко и др., 2004]. This theoretical device has shown its strength when solving different kinds of tasks, including business modelling. It possesses a very important advantage from the point of view of a considered problem – it is coordinated with object-oriented methodology [Украинец, 2007 В].

Functional systemology (systemological approach is the noospheric stage of the science development for complicated systems analysis) unlike other system approaches can be used to solve low-formalized ill-structured problems in the information and other areas. It allows forming informational resources of an organization; consolidating, systematizing, classifying information and knowledge; guarantees the “phase transfer of knowledge into power” [Бондаренко и др., 1998]. Obtaining of new knowledge on the basis of a new systemological method and criteria of natural classification allows to model most objectively deep conceptual knowledge in view of objects essential properties and to develop powerful ontology of ill-structured subject domains [Solovyova, 2000 etc.]. The perspective systemological technology of the organizational system modelling and analysis, material and information processes which proceed there (UFO-approach) allows consulting with challenges of putting in order in the working organization, designing and modelling of the new organization, and also forecasting of its development [Bondarenko et al, 2006, Бондаренко и др., 2004].

Systemological UML metamodel of Business

Definition a new DSL on the basis of UML that is UML profile development always assumes definition of structure of a language metamodel, metamodel unit structure – UML stereotypes and their metaclasses, and also relations between metamodel units. Three interconnected views are suggested to represent business model in language

UML (fig. 1). The business-instances model is the main starting point for business processes modelling (the functional structure of a business system). Business units, which provide performance of system functions, are represented on the subject domain model of business (organizational structure of a business system). The details of business processes in the form of separate business operations and performers responsible for their carrying out are represented on a business activity model [Украинец, 2008 B].

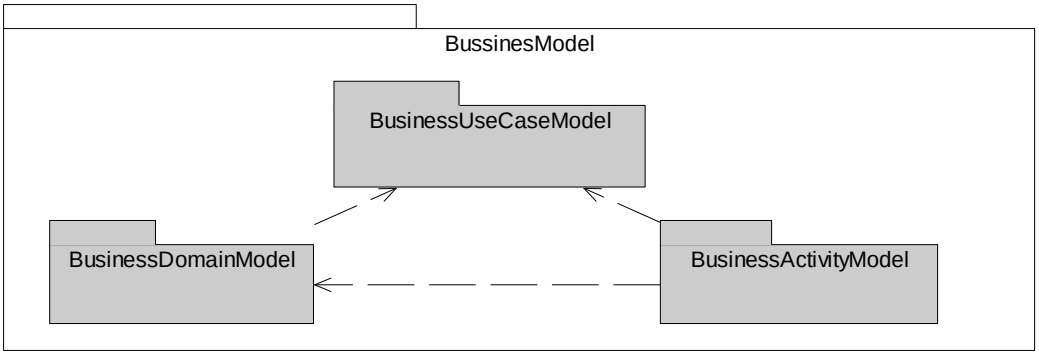

Fig. 1. Business model structure

The structure of each model's units (UML stereotypes) was defined according to the theoretical device of systemology and the UFO-approach to business modelling. To identify UML metaclasses corresponding to the new stereotypes the ontology of the common system modelling of Triune Continuum Paradigm was used [Naumenko, 2002]. Partly the results of this research [Украинец, 2008 B] are presented in tables 1, 2.

Table 1. Elements of Business Use Case Model

UFO Concept		UML Metaclass	UML Stereotype
UFO-Model		Model from Models	BusinessUseCaseModel
UFO-Element	Unit	1) Class from UseCases	1) BusinessSystem
		2) Class from UseCases	2) BusinessActor
	Function	1) UseCase from UseCases	1) BusinessUseCase
		2) Operation from Kernel, Interfaces	2) BusinessRequirement
	Object	1) Class from Kernel	1) BusinessWorker
		2) Class from Kernel	2) BusinessCustomer
Link		AssociationClass from AssociationClasses	BusinessEntity

Table 2. Elements of Business Activity Model

UFO Concept		UML Metaclass	UML Stereotype
UFO-Model		Model from Models	BusinessActivityModel
UFO-Element	Unit	Activity from BasicActivities, CompleteActivities, FundamentalActivities, StructuredActivities	BusinessActivity
	Function	Action from CompleteActivities, FundamentalActivities, StructuredActivities	BusinessAction
	Object	ActivityPartition from IntermediateActivities	BusinessActionWorker
Link		ObjectFlow from BasicActivities, CompleteActivities	BusinessEntityFlow

The relations between the business metamodel elements have been defined according to the theoretical device of systemology and the UFO-approach too. To provide strict correspondence of the UML metamodel and UFO-metamdel the mathematical apparatus of the theory of groups and the theory of graph was used [Украинец, 2008 A]. As a result a conceptual model of a UML profile has been obtained. The fragment of this model is presented on fig. 2.

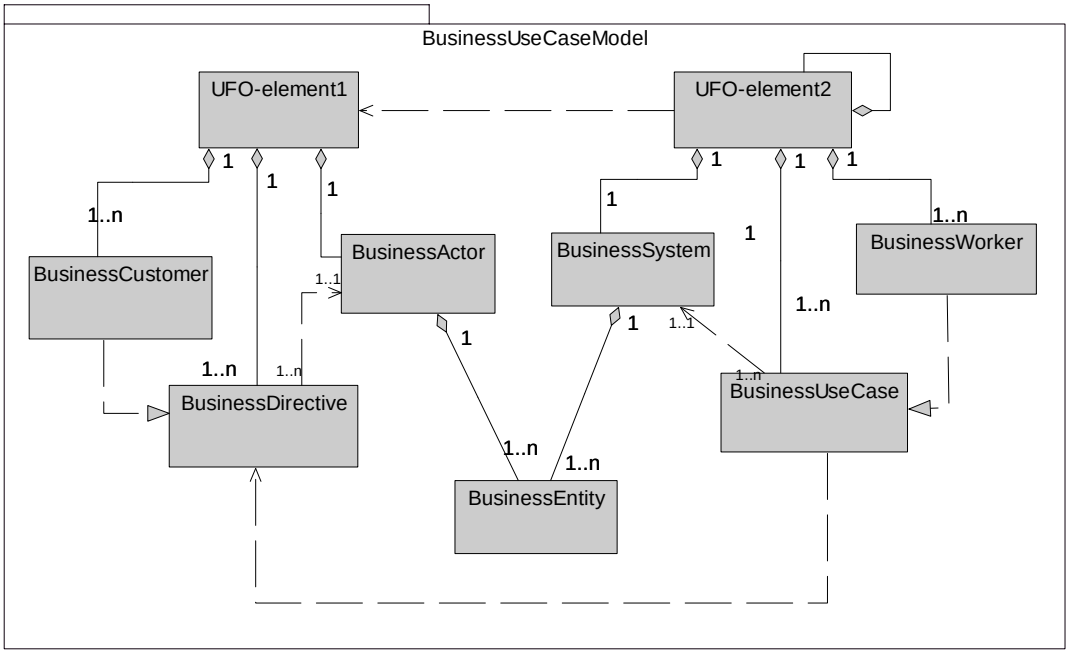

Fig. 2. Conceptual graph-analytic model of Business Use Cases

Going from Business Model to System Model

During business modelling not only it is important to obtain a qualitative model describing the purposes and the requirements of business to the program system, but also to provide usage of the modelling results in the analysis and system designing. For this reason it is necessary to describe formal link of a business model with a program system model. This solution will allow coordinating models and will give the chance to automate transition from a business model to a system model. It raises productivity of engineering process, reduces the system defects number, and also helps to make sure that the developed system solves real business requirements.

This idea is well stated in the MDA (Model Driven Architecture) international standard for which OMG (Object Management Group) is responsible [MDA, 2003]. There are three kinds of models described in MDA, which used during software development. The first one is called CIM (Computational Independent Model) – a business model. The second one is PIM (Platform Independent Model) that is a program system model. Meanwhile in MDA it is supposed that PIM can be partially obtained (generated) on the basis of CIM. The process of conversion of a business model into a system model from the considered issue's point of view according to MDA is shown on fig. 3. It is necessary to mention that nowadays the ideas of transformation of automated models and consideration of the business model as the key model in the software engineering process are widely spread.

For example, IBM Company widely advertises and actively extends such technologies as MDD (Model Driven Development) and BDD (Business Driven Development) [Swithinbank et al, 2004].

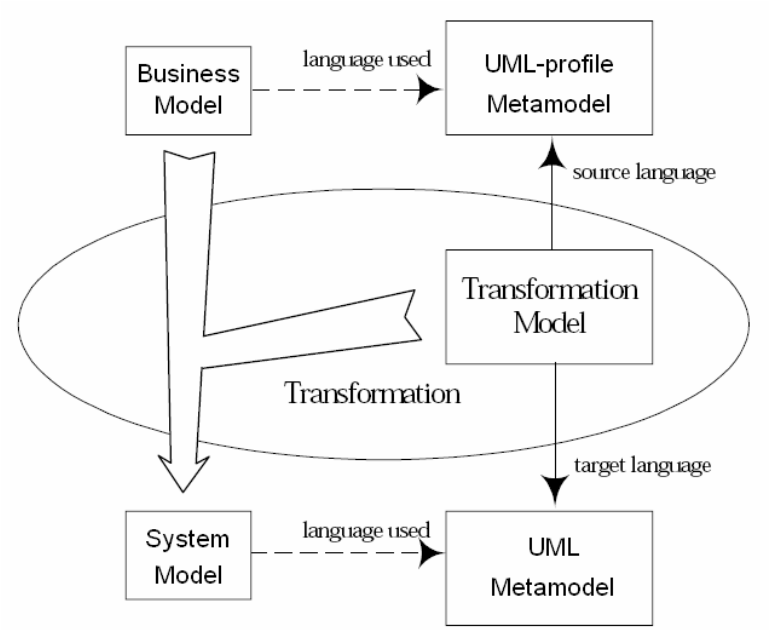

Fig. 3. Transformation of Business Model to System according to MDA

Before the conversion of the systemological business UML model in the system model some stages are supposed:

– creation of the business model

– syntax test of the business model on correctness, entirety and internal coordination that is expressed in the analysis of the admissible and necessary relations between the model elements;

– definition on the business model of those business processes and objects which will be automated at the expense of software development and deployment. They are designated by a stereotypic label "Automated".

The formal semantic link between the systemological business UML model and program system UML model has been defined using the ontology of the common system modelling of Triune Continuum Paradigm [Naumenko, 2002]. Partly he results of this research [Украинец, 2008 B] are presented in table 3.

Table 3. Semantic link between a program system model and a business model

System model	Business model
Actor from UseCases	BusinessActor from BusinessUseCase Model
	BusinessWorker from BusinessUseCase Model, BusinessDomain Model
Classifier from UseCases	BusinessSystem from BusinessUseCase Model
UseCase from UseCases	BusinessUseCase from BusinessUseCase Model
	BusinessAction from BusinessDomain Model
Activity from BasicActivities, CompleteActivities, FundamentalActivities, StructuredActivities	BusinessActivity from BusinessActivity Model
Action from CompleteActivities, FundamentalActivities, StructuredActivities	BusinessAction from BusinessActivity Model
ActivityPartition from IntermediateActivities	BusinessActionWorker from BusinessActivity Model
ObjectFlow from BasicActivities, CompleteActivities	BusinessEntityFlow from BusinessActivity Model

Package from Kernel	BusinessSystem from BusinessUseCaseModel
Class from Kernel	BusinessSystem from BusinessUseCase Model
	BusinessWorker from BusinessUseCase Model, BusinessDomain Model
	BusinessEntity from BusinessUseCase Model, BusinessActivity Model, BusinessDomain Model
Operation from Kernel, Interfaces	BusinessAction from BusinessActivity Model
	BusinessActionWorker from BusinessDomain Model
Relationships between classes	Relationships between business model elements

Design and Implementation of the Business Modelling System

The results obtained during the research have been brought to the level of practical implementation in the form of the program unit of the analysis and business modelling in Case tool of object-oriented modelling of IBM Rational Software Architect (RSA) [Swithinbank et al, 2004]. On fig. 4 the main functions of this program system and the users interested in it are presented. The description of the developed system model is given further.

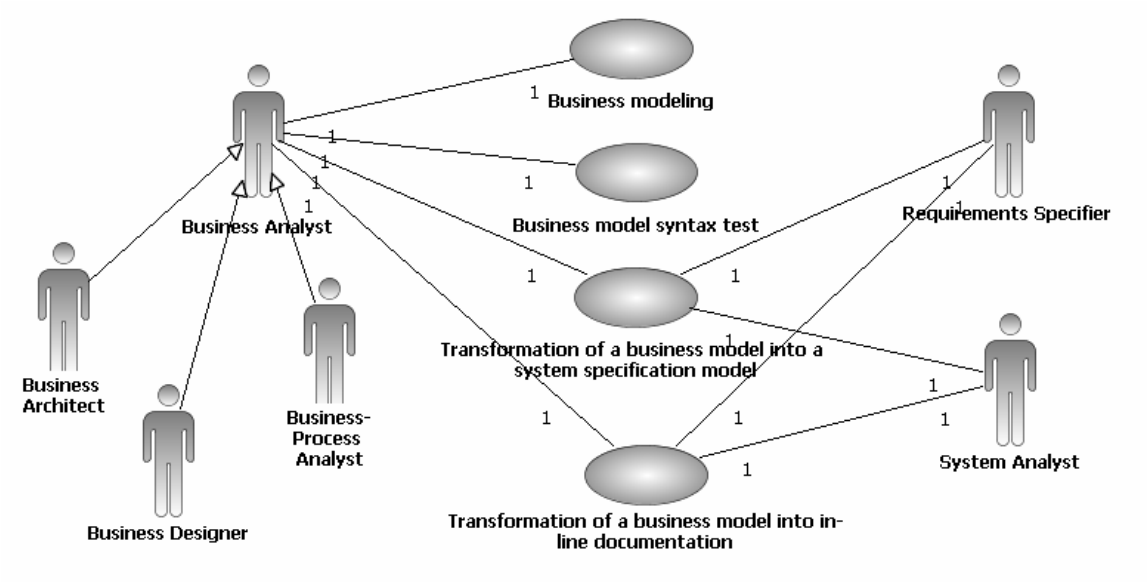

Fig. 4. Use case model of system of systemological UML modelling of business

Business analyst is the generalized role that includes common responsibilities for the business architect, business designer and business processes analyst. *Requirements Specifier* is the role specifies and supports the detailed requirements to the program system. *Systems analyst* is the role supervises and coordinates the process of requirements detection by definition of the fundamentals of the system functionality and boundaries.

Business modelling. This system use case allows a business analyst using a systemological UML profile to build business models which include a business Use Cases model, a business activity model and a business subject domain model. The business analyst can add, update or delete units of business model and the relations between them.

Business model syntax test. This system use case allows a business analyst to carry out verification on the syntax correctness of the constructed business UML model. During model verification the correctness of existing relations between units in a separate model, between units of different interconnected models, and correspondence of the specification of separate units of the model considered in different contexts in a business model is checked. Verification allows revealing the internal inconsistency of the business model, defining its

entirety, revealing duplicating and absent units of the model and the relations between them. As a result of the use case performance the business analyst obtains from system the report on containing errors in the model.

Transformation of a business model into a system specification model. This use case allows a business analyst, requirements qualifier and systems analyst to partially generate UML model of the program system specification from the systemological business UML model. As a result of the use case performance the use case model, activity model and class model of program system is partially generated.

Transformation of a business model into in-line documentation. This use case allows a business analyst, requirements qualifier and systems analyst to partially generate the in-line documentation from a systemological business UML model. As a result of the use case performance the use case model, a document which is the specification of a business system in the form of a structured text is partially generated.

The implementation of the business modelling system functions has been carried out by the various tools RSA presented in table 4.

Table 4. Implementation tools for business modelling system use cases in RSA

Use Case Name	Techniques in RSA
Business modelling	UML Profile Project
Business model syntax test	Constraints in OCL (Object Constraint Language) and built in RSA tools for analysis and constraints check
Transformation of a business model into a system specification model	Java Plug-in
Transformation of a business model into in-line documentation	Java Plug-in

Conclusion

Modern economic conditions, including the world financial crisis, dictate new rules of business behaviour in the area of information systems and technologies (IS / IT). They cause the necessity of strict control over return of investments and optimization of costs for software development and deployment in organizations along with other IS / IT. The developed language of business modelling, based on systemology and UML, is aimed at this problem solving. It allows connecting the purposes and business processes of an organisation with the software functions. Thus, it is possible to guarantee that the investments into software development and implementation are strictly aimed at the solution of business problems, perfection of its business processes and achievement of its competitive advantages.

While developing the language, its metamodel (a set of the model elements and the possible relations between them) and a strict formal connection between the business and system models were determined, the tools of modelling, business model syntax checking, as well as transformation of a business model into a system model and covering documentation were realized. The developed means of the language eliminate the defects of the analogues and allow avoiding errors in software requirements specifying and automating of its development. All this leads to decrease in effort, development and budget time especially in big software development projects.

Acknowledgements

The paper is partially financed by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA and the Consortium FOI Bulgaria (www.ithea.org, www.foibg.com).

Bibliography

[Leffingwell et al, 2000] D. Leffingwell, D. Widrig. Managing Software Requirements. A Unified Approach. AW, 2000.

[Украинец, 2008 A] Украинец, А.Г. Концептуальная модель UML профиля для моделирования бизнеса [Текст] / А.Г. Украинец // Системи управління, навігації та зв'язку. – К. : ЦНДІ НіУ, 2008. – Вип. 4(8). – С. 120 – 125.

[Graham, 2000] I. Graham. Object-oriented methods. Principles and Practice. Addison-Wesley Professional; 2000.

[Johnston, 2004] S. Johnston. Rational UML Profile for business modeling. IBM Rational Technical Library. [Online]. Available: http://www.ibm.com/developerworks/rational/library/5167.html

[Украинец, 2007 A] Украинец, А.Г. Системологический подход к моделированию бизнеса как основа бизнес-ориентированного подхода к информационным технологиям. [Текст] / А.Г. Украинец // Радиоэлектроника и информатика. - 2007. – №4. - С. 142 - 149. - ISSN 1563-0064.

[Бондаренко и др., 2004] Бондаренко, М.Ф. Моделирование и проектирование бизнес-систем: методы, стандарты, технологии. [Текст]: уч. пособ. / М.Ф. Бондаренко, С.И. Маторин, Е.А. Соловьева; – Х.: «СМИТ» - 2004. - 272с.

[Украинец, 2007 B] Украинец, А.Г. Разработка модели согласования объектно-ориентированной методологии и системологического подхода к моделированию бизнеса. [Текст] / А.Г. Украинец // Вісник Міжнародного Слов'янського університету. Серія «Технічні науки». - 2007. - Т.X, №2. - С.29-37 - 500 экз. - ISSN 1563-9961.

[Бондаренко и др., 1998] Бондаренко М.Ф., Соловьева Е.А., Маторин С.И. Основы системологии. Харьков: 1998. 118с.

[Solovyova, 2000] E Solovyova. Mathematical and Systemological Foundations of Natural Classification. Automatic Document and Mathematical Linguistics. New York: Allerton Press, Inc., 2000. V. 33, No. 4.

[Bondarenko et al, 2006] M. Bondarenko, K. Solovyova, S. Matorin and V. Matorin. "Embedding Conceptual Models of Knowledge in Technology of Systemological Analysis." Papers from the IFIP WG8.3 International Conference on Creativity and Innovation in Decision Making and Decision Support, vol. 2, p. 904-928, 2006, London, UK.

[Украинец, 2008 B] Украинец, А.Г. Структура и стереотипы UML-профиля для моделирования бизнеса[Текст] / А.Г. Украинец // Системи обробки інформації. – 2008. – вип. №6(73). - С. 142 - 149.

[Naumenko, 2002] A. Naumenko. Triune Continuum Paradigm: a paradigm for General System Modeling and its applications for UML and RM-ODP. Ph.D. thesis 2581, Swiss Federal Institute of Technology - Lausanne, June 2002. [Online]. Available: http://stella.epfl.ch/tcp/Naumenko-PhD-Thesis.pdf.

[MDA, 2003] MDA Guide Version 1.0.1. Object Management Group, Inc. [Online]. Available: http://www.omg.org/docs/omg/03-06-01.pdf

[Swithinbank et al, 2004] P. Swithinbank, M. Chessell, T. Gardner, C. Griffin, J. Man, H. Wylie, L. Yusuf. Patterns: Model-Driven Development Using IBM Rational Software Architect. IBM Corp, 2004. [Online]. Available: http://www.redbooks.ibm.com/redbooks/SG247105

Authors' Information

Mikhail Bondarenko - Rector, Kharkiv National University of Radio Electronics (KhNURE), Lenin Ave., 14, Kharkiv, 61166, Ukraine; e-mail: rector@kture.kharkov.ua

Nikolay Slipchenko - Chief of Scientific Department, Kharkiv National University of Radio Electronics, Lenin Ave., 14, Kharkiv, 61166, Ukraine; e-mail: slipchenko@kture.kharkov.ua

Kateryna Solovyova - Chief of Joint Scientific Educational Knowledge Management Center and Social Informatics Department, KhNURE, Lenin Ave., 14, Kharkiv, 61166, Ukraine; e-mail: si@kture.kharkov.ua

Olexiy Ukrayinets – Senior lecture of Social Informatics Department, Kharkiv National University of Radio Electronics, Lenin Ave., 14, Kharkiv, 61166, Ukraine; e-mail: agukr@inbox.ru

Larysa Vinogradova – Student of Social Informatics Department, Kharkiv National University of Radio Electronics, Lenin Ave., 14, Kharkiv, 61166, Ukraine; e-mail: si@kture.kharkov.ua