

D. E. DVUKHGLAVOV, A. V. DOLGORUKOV, S. R. SHOVKOPLIAS

DEVELOPMENT OF THE MODEL OF A DOMAIN REPRESENTATION IN THE DECISION SUPPORT SYSTEMS

The amount of information to be processed to make effective and informed decisions in the enterprises management in modern conditions of social development necessitates the use of decision support systems (DSS). Systems of this class provide storage and processing of large volumes of necessary information in a certain domain. At the same time, the high dynamism of the modern world leads to the fact that the domain represented in the system may change. This feature of modern conditions of DSS use determines the relevance of ensuring the adaptability of this systems class. This article presents the results of research on improving considered systems adaptability by providing the possibility of the modifying domain model represented in system being without making changes to the software architecture. The studies were aimed at resolving the discrepancy between the requirements of the systems adaptation efficiency to new functioning conditions and the actual duration of the existing adaptation procedures. It is shown that the traditional approach to the model representation, based on the mapping of the domain model in the relational database structure requires the involvement of specialists in the databases development and data access software. Standard procedures for the execution of such applications can take a long time. The approach which assumes that the domain model is stored in a special type of database in which the description of real objects, their attributes, attribute values and relations between them become records in the tables of such a database has been proposed as a solution for the discrepancy. This determines the possibility of the domain model modifying by changing of the tables contents without modifying structure of such database. To determine the such database structure, the domain representation model formalized description based on set theory has been made. The proposed approach is the basis for a software system design, using which will allow to modify the specific domain model directly by decision makers without involving correspond software developers. The primarily implementation area of the proposed solution is data-oriented DSS.

Keywords: decision support systems, domain model, description of the object by a set of characteristics, adaptability of decision support systems, modification of the domain model.

Д. Е. ДВУХГЛАВОВ, А. В. ДОЛГОРУКОВ, С. Р. ШОВКОПЛЯС

РОЗРОБКА МОДЕЛІ ПРЕДСТАВЛЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ В СИСТЕМАХ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

Обсяг інформації, що підлягає обробці для вироблення ефективних і обґрунтованих рішень при управлінні підприємствами в сучасних умовах розвитку суспільства, викликає необхідність використання систем підтримки прийняття рішень (СППР). Системи такого класу забезпечують зберігання і обробку великих обсягів необхідної інформації в певній предметній області. При цьому висока динамічність сучасного світу призводить до того, що модель предметної області, що представляється в системі, може змінюватися. Така особливість сучасних умов застосування СППР обумовлює актуальність систем даного класу. У даній статті представлені результати досліджень, присвячених підвищенню адаптивності розглянутих систем за рахунок забезпечення можливості модифікації моделі предметної області, що представляється у СППР, без внесення змін в архітектуру програмного забезпечення. Дослідження були спрямовані на розв'язання суперечності між вимогами оперативності адаптації систем до нових умов функціонування і реальною тривалістю існуючих процедур адаптації. Показується, що традиційний підхід до подання моделі, заснований на відображенні моделі предметної області в структурі реляційної бази даних, в разі необхідності зміни вимагає залучення фахівців з розробки баз даних і програмного забезпечення для доступу до даних. Стандартні процедури виконання таких модифікацій можуть розтягнутися на досить тривалий час. В якості рішення пропонується підхід, який передбачає зберігання моделі предметної області в базі даних спеціального виду, при якому опис реальних об'єктів, їх атрибутів, значень атрибутів і відносин між ними стають записами в таблицях такої бази даних. Це визначає можливість модифікації моделі предметної області шляхом зміни вмісту таблиць такої бази даних без модифікації структури. Для визначення структури такої бази даних виконано формалізований опис моделі представлення предметної області на основі теорії множин. Запропонований підхід є основою для розробки створення програмної системи, застосування якої дозволить модифікувати модель конкретної предметної області безпосередньо фахівцями щодо прийняття рішень без залучення представників розробника відповідної системи. Сферою впровадження запропонованого рішення є перш за все СППР, орієнтовані на дані.

Ключові слова: системи підтримки прийняття рішень, модель предметної області, опис об'єкту набором характеристик, адаптивність систем підтримки прийняття рішень, модифікація моделі предметної області.

Д. Э. ДВУХГЛАВОВ, А. В. ДОЛГОРУКОВ, С. Р. ШОВКОПЛЯС

РАЗРАБОТКА МОДЕЛИ ПРЕДСТАВЛЕНИЯ ПРЕДМЕТНОЙ ОБЛАСТИ В СИСТЕМАХ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ

Объем информации, подлежащей обработке для выработки эффективных и обоснованных решений при управлении предприятиями в современных условиях развития общества, вызывает необходимость использования систем поддержки принятия решений (СППР). Системы такого класса обеспечивают хранение и обработку больших объемов необходимой информации в некоторой предметной области. При этом высокая динамичность современного мира приводит к тому, что модель предметной области, представляемой в системе, может изменяться. Такая особенность современных условий применения СППР обуславливает актуальность обеспечения адаптивности систем данного класса. В данной статье представлены результаты исследований, посвященных повышению адаптивности рассматриваемых систем за счет обеспечения возможности модификации модели представляемой предметной области без внесения изменений в архитектуру программного обеспечения. Исследования были направлены на разрешение противоречия между требованиями оперативности адаптации систем к новым условиям функционирования и реальной продолжительностью существующих процедур адаптации. Показывается, что традиционный подход к представлению модели, основанный на отображении модели предметной области в структуре реляционной базы данных, в случае необходимости изменения требует привлечения специалистов по разработке баз данных и программного обеспечения для доступа к данным. Стандартные процедуры выполнения таких модификаций могут растянуться на достаточно длительное время. В качестве решения предлагается подход, который предполагает хранение модели предметной области в базе данных специального вида, при котором описание реальных объектов, их атрибутов, значений атрибутов и отношений между ними становятся записями в таблицах такой базы данных. Это определяет возможность модификации модели предметной области путем изменения содержимого таблиц такой базы данных без модификации структуры. Для определения структуры такой базы данных выполнено формализованное описание модели представления

© D. E. Dvukhglavov, A. V. Dolgorukov, S. R. Shovkoplias, 2018

предметной области на основе теории множеств. Предложенный подход является основой для разработки создания программной системы, применение которой позволит модифицировать модель конкретной предметной области непосредственно специалистами по принятию решений без привлечения представителей разработчика соответствующей системы. Сферой внедрения предложенного решения являются прежде всего СППР, ориентированные на данные.

Ключевые слова: системы поддержки принятия решений, модель предметной области, описание объектов набором характеристик, адаптивность систем поддержки принятия решений, модификация модели предметной области.

Introduction.

The effectiveness of activities in most industries is determined by the quality of decisions that are made. Decisions can be considered "good" if in the process of their preparation all known factors are analyzed, possible consequences of the decisions taken are determined, they are accepted by the deadline. Such an approach is important for the management of enterprises, for predicting the development of economic situations, in military affairs, in medicine, in power engineering, etc.

The amount of information to be processed to make effective and informed decisions in the enterprises management in modern conditions of social development necessitates the use of decision support systems (DSS). Systems of this class provide storage and processing of large volumes of necessary information in a certain domain.

At the same time, the high dynamism of the modern world leads to the fact that the domain area represented in the system may change. This feature of modern conditions of DSS application determines the relevance of research on improving the adaptability of this class systems.

Literature review and problem statement.

General decision making issues are presented in [1–3]. Questions on the purpose of DSS, their classification, their structure and principles of development can be found in [4–6]. In [6] presented in particular the definition of DSS, which is considered in the research. Decision Support System (DSS) is a computer-aided computer system whose goal is to provide assistance to decision makers in difficult conditions for a complete and objective analysis of the subject activity. DSS in most cases is an interactive automated system that helps decision makers use data and models to identify and solve problems and make decisions.

Requirements defined for the DSS, include one which considered that in such a class of systems it is necessary to ensure the possibility of adapting changes to requirements and external conditions for making decisions. In turn, the provision of such a "flexibility" contributes to improving the reliability of systems, which refers to the ability of the system to perform the required functions over a long period [7].

Adaptability of data-oriented DSS can be achieved by quickly changing the domain model presented in the DSS. Traditionally, the domain model in such systems is implemented in the tables structure of the DSS data management system. The most common is the variant of organizing data storage in relational databases [8, 9]. If the database structure changes, either the field (s) will be added to a specific table, or a new table will be added. Modern development practices suggest that a graphical user interface is being developed to fill tables with data. This can be either desktop forms or web-pages. Changes in the structure will necessarily require changes to the data

entry interface. In work [10] it is shown that at the present time to modify the DSS database and the interface for its content only specialists with special knowledge and skills can be involved. Taking into account the fact that these specialists and users are often far from each other, the time for modifying the database structure and software modules for data manipulation can take quite a long time, makes relevant the possibility of making changes by users of the system.

Thus, it can be argued that there is a contradiction between the requirements of the efficiency of adapting systems to new operating conditions and the actual duration of the existing adaptation procedures. The solution may be to modify the database structure without the participation of programmers and without refactoring software components.

An approach to solving this problem was proposed in [10]. But practical developments have shown that the presented approach does not take into account the existence of a connection between typical objects of the region.

The purpose of the article.

The purpose of the article is improving a method of structuring information about the domain, which used in decision making process, due to the representation in models of the relationship between the objects of the domain.

Representation domain method improvement.

The essence of an approach to structuring is based on a fact that a subject industry is a set of objects. Each object describes a set of characteristics whose values form a clear idea of it. Experts should define types of classes for the structuring of the subject industry. Such a division will allow organizing the characteristics in a bulk form and provide in a certain sequence according to user requirements. The principle of representing the types of objects is demonstrated in Fig. 1. Formally the type of TOb_i objects can be described as follows

$$TOb_i = \langle NTOb_i, RATO b^i \rangle, \quad (1)$$

where $NTOb_i$ – a name of a type;

$RATO b^i$ – a set of descriptions characteristics associated with this type

$$RATO b^i = \{RA_m^{ij}\}. \quad (2)$$

Each element of a $RATO b^i$ set corresponds to establishing a connection between an object type and the corresponding characteristic:

$$RA_m^{ij} \Leftrightarrow A_m \rightarrow TOb_j. \quad (3)$$

It is assumed that there is a set of characteristics $A = A_1, A_2, \dots, A_{m-1}, A_m, A_{m+1}, \dots, A_M$, which can be used to describe any type of an object.

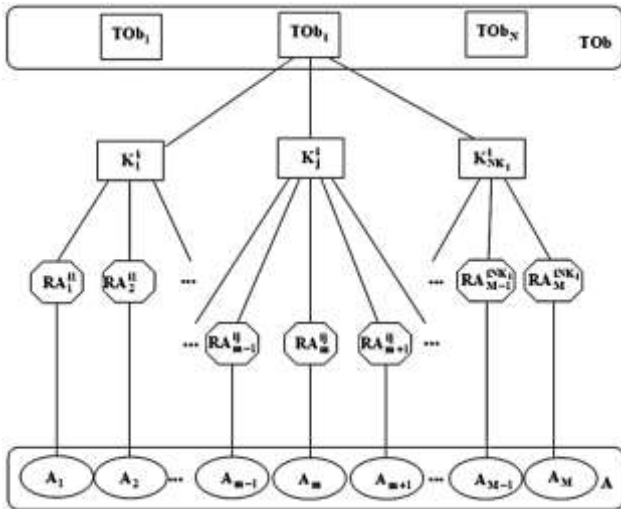


Fig. 1. Structure of representation of a typical object of the subject field

Also, when establishing a connection "object type-characteristic", a class K_{ij} ($K_j^i \in K^i$, K^i is a set of classes of the i -th type defined for the content structuring of the characteristics) is defined, to which a characteristic will be included, as well as restrictions are determined on the value of the characteristic RA_m^{ij} . So:

$$RA_m^{ij} = \langle A_m, K_{ij}, PA_m^{ij} \rangle. \quad (4)$$

Thus, a feature is included in only one class K_{ij} . The list of PA_m^{ij} limits depends on the type of value of the characteristic and differs when using the characteristic to describe objects of different types. The field of use of the method allows one to determine that the characteristic can store the value of one of six types: integer values (not a small part is allowed); numeric (the small part is allowed); symbolic; logical (true or false); date and time.

For each characteristic, restrictions are set for acceptable values. If a characteristic type is set to "listed", then a pre-defined set of possible values from which you can choose the desired option is pre-defined for it. For non-listed characteristics, values are arbitrary.

Depending on the type of values, one can set additional parameters. Thus, the method assumes that the logical value is an enumerated value with two possible values - "yes" and "no". For character non-listed values, one must set a maximum length of a value. For other types, if an arbitrary value is taken, maximum and minimum values that will determine the range of possible values should be set.

Part of the information on the domain is determined by the presence of different objects of relations of different types TL_r , $TL_r \in TL$. The ratio is binary and is set between the object types. Relationships of one type can be established between different types of objects. The set of relations taking place in the domain is a set $TLOB$:

$$TLOB = \{ TLOB_s \} =$$

$$= \{ \langle TOB_p, TOB_q, TL_r, RAL^{pqr} \rangle \}. \quad (5)$$

Relationships can also have attributes that are described in the same way with the attributes of object types, that is, they are also formed a description RAL^{pqr} , which defines the values range of attribute A_m , which is used to describe the type TL_r between objects types TOB_p and TOB_q . The difference is that the relationship characteristics are not structured. This is determined by the fact that the number of attributes describing the relationship does not usually exceed 5. Thus, we obtain:

$$RAL^{pqr} = \{ \langle RAL_m^{pqr} \rangle \} = \{ \langle A_m, PAL_m^{pqr} \rangle \}. \quad (6)$$

The structure of a typical description of the relationship between objects is illustrated in Fig. 2.

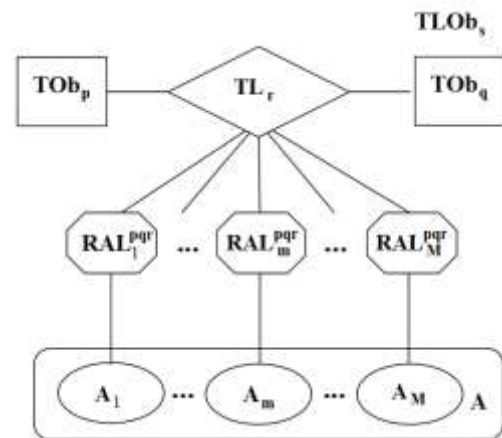


Fig. 2. The structure of the description of the typical relationship between objects

On the basis of certain types of objects, objects will be created next to certain objects of the domain:

$$Ob_n^i = Ob(TOb_i) = \langle NOB_n^i, ZA_n^i \rangle, \quad (7)$$

where NOB_n^i – a object name;

$ZA_n^i = \{ ZA_m^{in} \}$, $ZA_m^{in} \in PA_m^{ij}$ – a value set of n -th object i -th type.

This will characterize the transition from a general description of the domain structure to the detailed one. The defined structure of a particular Ob_n^i object in this case contains the values of corresponding characteristics ZA_m .

Fig. 3 illustrates how the information will be presented to the user.

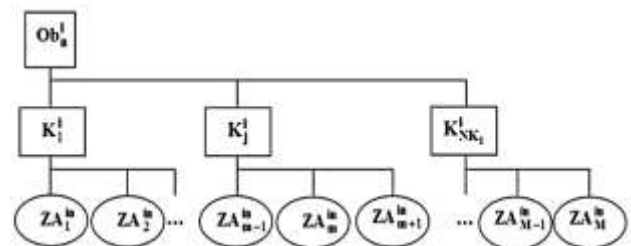


Fig. 3. The structure of i -th object instance

After creating certain objects between them, the relation of this form is established:

$$LOb_j^s = \{ LOb_s \} = \\ = \langle TOb_k^p, TOb_t^q, TL_r, ZAL_m^{ktr} \rangle. \quad (8)$$

Each of these relationships is created on the basis of a predefined relationship type $LOb_s = LOb(TLOB^s)$. TOb_k^p and TOb_t^q – these are instances of objects defined in the typical description of the relation $TLOB^s$; the relation characteristics take values from the corresponding values set defined for them when creating the type of relation:

$$ZAL_m^{ktr} \in PAL_m^{pqr}. \quad (9)$$

The structure of the relationship between certain objects is illustrated by Fig. 4.

Establishing relationships between types has the following features:

- relationships between different types can be established between two types of objects;
- a relationship between the types is characterized by a different set of characteristics;
- a list of types of relations coincides with the types of binary relations that are established between the entities of the database.

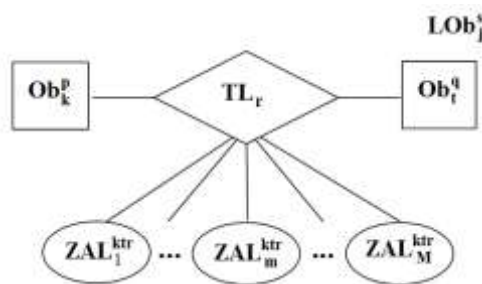


Fig. 4. The structure of the relationship between two objects instances

The principles for defining a set of characteristics for relations are similar to setting characteristics for an object type. Often, such a set is empty, because the importance of the presence of the relationship is important. Practical developments indicate that the number of relationship characteristics is not very large, so structuring on the classes is not used.

To create a domain model in accordance with the proposed principles, it is necessary to develop a structure for storing model components and a software system for working with them. When forming the requirements for the system architecture of the software system, the features of enterprise management should be taken into account. presents the requirements that the developed system should be placed on the server and provide a convenient interface to the user through the desktop client ("thick client" – for visual and interactive visualization), and through a browser ("thin client" – fewer features, but access is possible with any computer, without any preset). In addition, in the same work it is proposed to provide for the possibility of integration with existing accounting

systems of enterprises [11]. In modern conditions, a promising direction should be considered integration with SAP [12].

Conclusions.

Thus, the set of object types, their characteristics, relations between them allows us to represent the subject area of the decision-making task. Creating a domain model in accordance with the proposed approach has a similar sequence with the creation of the database table structure. But the proposed method provides the creation of an interface for inputting model components that describe an arbitrary subject area. The presence of such an interface will allow changing the create or change the domain model by decision makers in the case of expanding its boundaries. It should also be noted that the creation of such an interface will allow you to enter information about new objects immediately after creating the object type and determining (adding) characteristics and relations. This, in turn, increases the level of adaptability of DSS, first of all, of data-oriented DSS.

References

1. Черняховская Л. Р. *Поддержка принятия решений при стратегическом управлении предприятием на основе инженерий знаний*. Уфа: АН РБ, Гилем, 2010. 168 с.
2. Афоничкин А. И., Михаленко Д. Г. *Управленческие решения в экономических системах*. Санкт Петербург: Питер, 2009. 480 с.
3. Александров О. В., Двухглавов Д. Е., Павленко М. А., Романенко І. О., Тимочко О. І. *Теоретичні основи автоматизації процесів вироблення рішень в системах управління Повітряних Сил*. Харків: ХУ ПС, 2010. 172 с.
4. Герасимов Б. М., Дивизинюк М. М., Субач І. Ю. *Системы поддержки принятия решений: проектирование, применение, оценка эффективности*. Севастополь: Изд. центр СНИЯЭиП, 2004. 320 с.
5. Ситник В. Ф. *Системы підтримки прийняття рішень*. Київ: КНЕУ, 2004. 614 с.
6. Бідок П. І., Гожій О. П., Коренюк Л. О. *Комп'ютерні системи підтримки прийняття рішень*. Миколаїв: Вид-во ЧДУ ім. Петра Могили, 2012. 382 с.
7. Верескун В. Д., Цуриков А. Н. *Информационно-управляющие системы в научных исследованиях и производстве*. Ростов-на-Дону: ФГБОУ ВО РГУПС, 2016. 76 с.
8. Пасічник В. В., Реаніченко В. А. *Організація баз даних та знань*. Київ: BHV, 2006. 384 с.
9. Коннолли Т., Бегг К. *Базы данных. Проектирование, реализация и сопровождение. Теория и практика*. Москва: Вильямс, 2017. 1440 с.
10. Судаков Б. Н., Двухглавов Д. Е., Володіна І. М. *Метод структуризації предметної галузі в системах підтримки прийняття рішень*. *Вісник Нац. техн. ун-ту «ХПИ»: зб. наук. пр. Темат. вип.: Інформатика і моделювання*. Харків: НТУ «ХПИ». 2011. № 17. С. 157–161.
11. Скобелев П. О. *Интеллектуальные системы управления ресурсами в реальном времени на основе мультиагентных технологий*. URL: http://www.ssc.smr.ru/media/ipuss_conf/15/6_01.pdf (accessed 15.11.2018)
12. *Офіційна веб-сторінка компанії SAP*. URL: <https://www.sap.com/ukraine/index.html> (accessed 15.11.2018).

References (transliterated)

1. Chernjahovskaja L. R. *Podderzhka prinjatija reshenij pri strategicheskom upravlenii predpriyatijem na osnove inzhenerij znaniy* [Decision support in strategic enterprise management based on knowledge engineering]. Ufa, AN RB, Gilem Publ., 2010. 168 p.
2. Afonichkin A. I., Mihalenko D. G. *Upravlencheskie reshenija v jekonomicheskijh sistemah* [Management solutions in economic systems]. Saint Petersburg, Piter Publ., 2009. 480 p.

3. Aleksandrov O. V., Dvukhhlavov D. E., Pavlenko M. A., Romanenko I. O., Tymochko O. I. *Teoretychni osnovy avtomatyzatsiyi protsesiv vyroblennya rishen' v systemax upravlimyia Povitryanykh Syl* [Theoretical basis of automation of decision-making processes in Air Force control systems]. Kharkiv, KhU PS Publ., 2010. 172 p.
4. Gerasimov B. M., Divizinjuk M. M., Subach I. Ju. *Sistemy podderzhki prinyatija reshenij: proektirovanie, primenenie, ocenka jeffektivnosti* [Decision making support systems: design, application, performance evaluation]. Sevastopol', Izd.centr SNIJaJeIP Publ., 2004. 320 p.
5. Sytnyk V. F. *Sistemy pidtrymky pryynyattya rishen' [Decision making support systems]*. Kiev, KNEU Publ., 2004. 614 p.
6. Bidyuk P. I., Hozhiy O. P., Korenyuk L. O. *Komp'yuterni systemy pidtrymky pryynyattya rishen' [Computer decision making support systems]*. Mykolayiv, ChDU im. Petra Mohyly Publ., 2012. 382 p.
7. Vereskun V. D., Curikov A. N. *Informacionno-upravljajushhie sistemy v nauchnyh issledovaniyah i proizvodstve* [Information management systems in research and production]. Rostov-na-Donu, FGBOU VO RGUPS Publ., 2016. 76 p.
8. Pasichnyk V. V., Reanichenko V. A. *Orhanizatsiya baz danykh ta znan'* [Organization of bases of data and knowledge]. Kiev, BVH Publ., 2006. 384 p.
9. Konnolli T., Begg K. *Bazy dannyh. Proektirovanie, realizacija i soprovozhdenie. Teorija i praktika* [Databases. Design, implementation and maintenance. Theory and practice]. Moscow, Vil'jams Publ., 2017. 1440 p.
10. Sudakov B.N., Dvukhhlavov D.E., Volodina I.M. *Metod strukturyzatsiyi predmetnoyi haluzi v systemakh pidtrymky pryynyattya rishen'* [Method of structuring the subject area in decision making support systems]. *Visnyk Nats. tekhn. un-tu «KhPI»: zb. nauk. pr. Temat. vyp.: Informatyka i modelyuvannya*. [Bulletin of the National Technical University "KhPI": a collection of scientific papers. Thematic issue: Informatic and Modelling]. Kharkiv, NTU "KhPI" Publ., 2011. № 17. pp. 157–161.
11. Skobelev P.O. *Intellektual'nye sistemy upravlenija resursami v real'nom vremeni na osnove mul'tiagentnyh tehnologij* [Intelligent real-time resource management systems based on multi-agent technologies]. URL: http://www.ssc.smr.ru/media/ipuss_conf/15/6_01.pdf (accessed 15.11.2018).
12. *Ofitsiyna web-storinka kompaniyi SAP* [Official web-page of the company SAP]. URL: <https://www.sap.com/ukraine/index.html> (accessed 15.11.2018).

Received 20.11.2018

Відомості про авторів / Сведения об авторах / About the Authors

Двухглавов Дмитро Едуардович (Двухглавов Дмитрий Эдуардович, Dvukhglavov Dmytro Eduardovych) – кандидат технічних наук, доцент, Національний технічний університет «Харківський політехнічний інститут», доцент кафедри «Програмна інженерія та інформаційні технології управління»; ORCID: <https://orcid.org/0000-0002-3361-3212>; e-mail: ddimae72@gmail.com

Долгоруков Олексій Веніамінович (Долгоруков Алексей Вениаминович, Dolgorukov Alexey Veniaminovich) – кандидат технічних наук, Національний дослідницький технологічний університет "МІСТАС", доцент кафедри «Бізнес-інформатика та системи управління виробництвом»; ORCID: <https://orcid.org/0000-0002-9779-4550>; e-mail: DolgorukovAV@misis.ru

Шовкоплас Сергій Ростиславович (Шовкоплас Сергей Ростиславович, Shovkoplias Serhii Rostislavovich) – Національний технічний університет «Харківський політехнічний інститут», студент кафедри програмної інженерії та інформаційних технологій управління; ORCID: <https://orcid.org/0000-0003-1837-0213>; e-mail: sergey_shovkoplyas@outlook.com