International Journal of Applied Engineering Research ISSN 0973-4562 Volume 10, Number 15 (2015) pp 35239-35246

Model-Driven Design and Implementation of Scientific Data Management Information System

Pavel P. Oleynik, PhD

System Architect Software, Aston JSC Shakhty Institute (branch) of Platov South Russian State Polytechnic University (NPI) Rostov-on-Don, Russia
xsl@list.ru

Nikolay V. Kuznetsov, PhD

Advisor of CEO, Kaskad JSC, Director of the Center for Institutions of Innovation Economy Development, Financial University under the Government of the Russian Federation, Moscow, Russia,
nkuznetsov@outlook.com

Edward G. Galiaskarov and Natalia E. Borodina

Ivanovo State University of Chemistry and Technology Ivanovo, Russia
portugaled@yandex.ru and natashka0108@yandex.ru

Abstract
The article describes the process of model-driven design and implementation a new information system that is used for scientific data management and in international scientific-practical conference “Object systems” (objectsystems.ru). Reengineering of the existing system was made on the basis of analysis and object modeling of the domain. As a result, the existing errors have been corrected and new features have been added. All available data were saved and imported into the new system.

Keywords—UML; MDA; Object Design; Object Modeling; Database

I. INTRODUCTION

Publications in scientific journals and presentation at scientific conferences are one of the main ways of sharing of scientific knowledge, a way of demonstrating their scientific achievements. Participation in public scientific conferences is the most affordable, dynamic and effective way. For today a huge amount of scientific conferences exists. The conference organizers are faced with a serious problem of effective management of scientific publications submitted to the conference, and of a large amount of additional supportive organizational data: publication review, payment, collections sent to participants, etc. Solution of these problems requires the development and utilization of automated information systems.

This article focuses on the development of an information system for cataloging scientific publications based on reengineering the existing cataloging information system for conference "Object Systems" (objectsystems.ru). Section 1 contains in-depth analysis of existing publications on similar themes and of the results achieved. Performed analysis of the domain business processes is presented in Section 2. UML activity diagram is shown to demonstrate the results. The optimality criteria that have been identified for the new system under development are described in Section 3. Section 4 shows a reengineering process of existing system and describes the pitfalls therein. The import process into the new system of existing data accumulated over several years is presented in section 5. The implementation of the application business logic is described in section 6. In conclusion the issues of further development of this work are considered.

II. REVIEW OF EXISTING PUBLICATIONS

The idea of the article is not new, but the problem of cataloging scientific papers was considered by the other authors. The paper [1] is one of the first works devoted to similar subjects. The author describes the software used during the IEEE International Conference on Acoustics, Speech, and Signal Processing in 1988. At the time it was required to implement own database on the basis of DBASEIII in order to carry cataloging thousands of scientific papers. The text output to the screen is used to display information to the user. Graphical user interface which is used everywhere now was absent at the time. This work was performed in AT & T Bell Labs and has been used for a long time in the organization of scientific events.

In [2] the authors pay attention to the problem of preservation and transmission of scientific information and interaction of scientists at a distance from each other. This is done using a program developed in Java and called HS113, the essence of which is to send the image via various communication channels. Particular attention is paid to the hardware component of the developed complex. This system has been used by the authors in distance learning and conducting scientific conferences, when participants could not attend in person. There is also described in detail the functionality of

the different roles which scientists can execute during the scientific events.

The later work [3] is devoted to the development of a scientific works content registration system on the basis of a digital object identifier (DOI). After analyzing the hierarchical structure of DOI formation, the authors propose the architecture of distributed repositories for information, using different stores for different types of publications (journal articles, books, and other data). Then the authors examine in detail the information registration process that begins with the DOI parsing and creation of internal metadata, allowing to define a path and a physical store. With the help of the mechanism the authors implemented their own portal that enables to catalog the scientific publications and to search for information. Despite the well-conceived ideas and presence of a practically implemented system, there is the key shortcoming in this work. DOI is assigned to the article only after the publication in the finished collection of works. At the time of the conference only DOI for the entire collection can exist, and this value can be absent for a particular article.

The problem of designing the scientific works cataloging systems from positions of security storage and access to data is studied in [4]. The authors present their own architecture, implemented in the form of interface for calling access functions. Attention is also paid to the composition and structure of software modules and hardware environment for developed software application. The authors propose to pay close attention to the audit mechanism and implement the basic module that logged all system activity. Unfortunately, the authors do not pay enough attention to the problem of constructing the user interface and functions provided by the system. All this is important for conducting a scientific conference.

The scientific information management system "ISTINA – Nauka MGU" described in detail in [5] is the most developed implementation. This system is intended for cataloging different types of work and has developed Web-based interface that enables the easy retrieval of information. But despite the presence of a number of advantages, proposed system cannot be used in conducting its own conferences, because does not provide saving for additional information required in this case. For example, we need to keep the entire review contribution process and track receipt of funds for participation.

At the moment on the Internet, there are several large services that simplify the process of the conference and implemented in the form of Web-sites. These include the two largest easychair.org and yeedao.net. They allow the organizer to enter information about this conference and the author to add his/her article. At the same time the ability to view the status of their publications and make the required changes in accordance with the observations of the reviewers exists. However, these services do not provide the ability to take into account the participation fee. This functionality is implemented either on the official website of the conference or in person by members of the Organizing Committee during the conference. The second method is absolutely unacceptable during the correspondence conferences, which now hold a lot.

Furthermore, the described sites do not allow reviewers to make corrections immediately into files because articles are presented in a pdf-file. The experience of the conference "Object Systems" showed that the most convenient way to transfer articles are a MS Word file, because it allows to make all the reviews directly into it and thereby reduce the time to obtain the final version of the papers accepted for publication.

III. ANALYSIS OF THE DOMAIN BUSINESS PROCESSES

To perform high-quality development of information system, it is necessary to have a complete understanding of the problem domain. For this purpose it requires analysis of the domain business processes. In the beginning, there should define stakeholders interested in the system.

1. The conference organizer is the main actor and a system user. The following tasks are a part of his responsibilities:
   - register publications;
   - appoint the reviewer;
   - check corrections according to the reviewer's comments;
   - check payment;
   - impose collection;
   - send collections and certificates to the authors.
2. The author is the one who writes an article and sends it to the conference. In addition, the author's responsibilities include the revision of article in accordance with the comments that the reviewer determines during the article review and the payment a fee if it is necessary.
3. The reviewer is someone who checks an article and evaluates its quality. Review of the article is the main task of the reviewer. Reviewing is to write comments and tips for improving the article and to determine the review results: to take an article for publication, reject or send back for revision. The reviewer also determines the best article in one of the possible nominations.

Figure 1 shows the business process model that describes the whole cycle of work with the publication of each author from the date of its submission to the shipment to the author of the conference proceedings.

IV. OPTIMALITY CRITERIA OF SYSTEM UNDER DEVELOPMENT

Before proceeding to develop an information system it needs to define the optimality criteria which are the requirements for system functionality. For the described application the optimality criteria (CO) are [9]:

1. Implement the ability to store information about different types of editions given the fact that the edition may be inner (conducted independently) or external (which is held by a third party);
2. Provide the ability to save information about the authors indicating academic degrees and positions in different organizations;

3. Realize the possibility to store information about organization including their type and departments in which the authors work;
4. Provide the ability to save information about the review of the article and their results;
5. Implement the ability to save information about the release of various types of editions, as well as providing information about them to the user;
6. Implement a mechanism for storing information about the articles written by certain authors;
7. Provide the ability to save information about the committees and the authors who are committee participants;

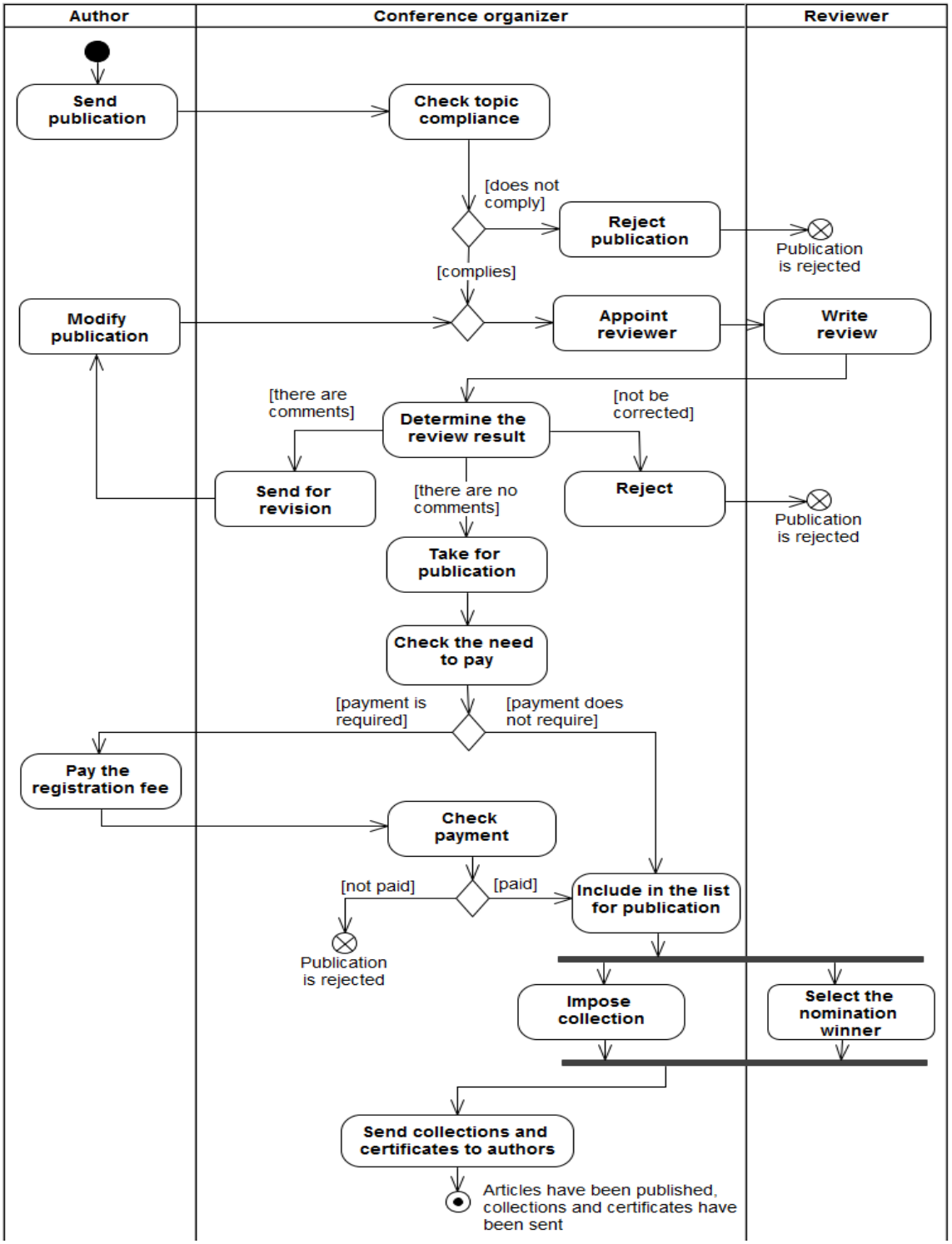

Fig. 1. **UML activity diagram of scientific data management**

8. Implement the ability to save information about the nominations of articles;
9. Provide the ability to track the recommendations of the publication of articles in other editions, such as magazines recommended by Higher Attestation Commission to publish the results of master's and doctoral thesis researches;
10. Implement analytics module that displays information about the articles that are in a certain state, about the packages with collections and certificates sent to authors and about the issues for committee participants.

The new information system has been developed on the basis of the optimality criteria.

V. SYSTEM REENGINEERING

The existing system was designed when the requirements were not fully clear. Design errors were discovered during the operation of the system, and hence the need arose for additional functionality. Problems encountered in the system designed are related to the characteristics of the selected programming language C# in which there is no multiple inheritance of implementations [12-13]. The result is that the program code has turned out quite cumbersome. Multiple inheritance of interface declarations are also not used.

Model of the existing system was obtained with reverse engineering of source code using Enterprise Architect which is a CASE platform for software design. After analyzing the model a number of defects have been identified namely:

1. The system maintains the information of only two types of editions: conference and journal which are subdivided in turn on inner (self-conducted) and external (conducted by an outside organization). Class hierarchy is organized quite irrationally. Essentially we have two kinds of editions, but they are represented in the four classes: an inner edition, an external edition, a journal article and a conference publication.
2. The presence of numerous implementations for the same interfaces that had to be used to emulate multiple inheritances.
3. Duplication of the same attributes in different base classes due to a lack of multiple inheritance.

Given the existence of these shortcomings, it was decided to implement a new version of the program, the core of which is the architecture of MDA (Model Driven Architecture). One of the standards used in the MDA technique is a graphical Unified Modeling Language (UML), which is basically positioned as independent of platforms and technologies. That is why it is used for the design of various aspects of information systems. We will use the class diagrams only because they allow to reflect the application structure. To implement an information system, the unified rapid development of enterprise information systems, SharpArchitect RAD Studio, is used [10-11].

The resulting model of the class hierarchy is divided into two parts and is shown in Fig. 2 and 3. Fig. 2 is a class diagram of the hierarchy of services provided with conference. Fig. 3 shows the basic class diagram.

As can be seen from Fig. 2 and 3 there are classes with postfix Object in the model. Classes Object are abstract base, non-persistent classes that are used to represent of attributes of different types. An abstract class in UML is denoted by writing the class name in italics. Because the class diagram was built in Microsoft Visual Studio 2012 and abstract classes are implemented as interfaces, then they appear regular font (not tilted). This implementation can be explained by the fact that multiple inheritance is used, but in C# multiple inheritance can be emulated through the interfaces only.

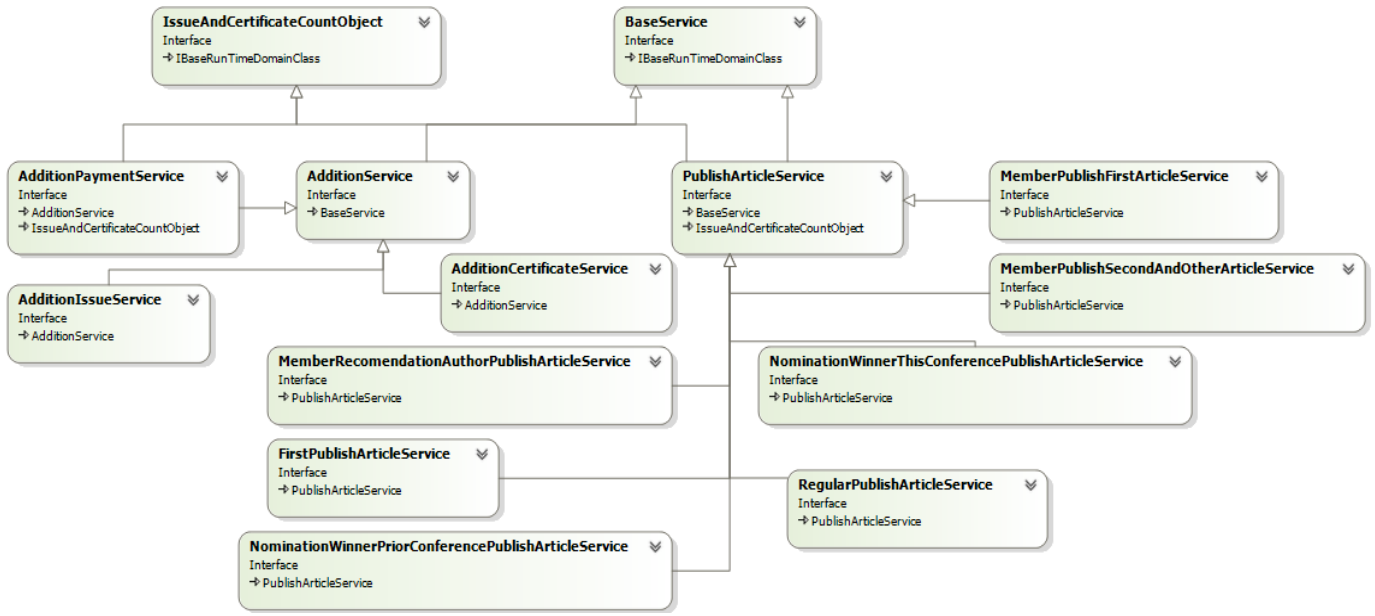

Fig. 2. **UML class diagram of the hierarchy of services scientific data management**

Fig. 3. **Basic UML class diagram**

There are 14 abstract classes in whole model. Each such class keeps information about a specific attribute. It is possible to inherit an Object class from another.

As properties and signs of attribute are defined in the abstract class, its use allows to reduce time and to facilitate implementation process of derivative classes. In case of this approach there is no duplicating of a code. From this it follows that if necessary to make any changes or in case of detection of an error, correction will happen in one certain place of the program (base class).

According to CO_1 and also on the revealed shortcomings of the existing system, the class Edition Kind that defines kind of edition was selected in this model. It allows storing the different kinds of edition (journal, conference, monograph, etc.) in system without creating new tables in database. All information on edition is stored in one class Base Edition. As editions can be inner or own (conducted independently) and external, there are two derivatives of base class: Own Edition and Foreign Edition respectively.

The above classes have a relation at Foreign Issue and Own Issue respectively. These classes represent the entities storing information about issues of editions (the number of journal, collections of conference), external and own issues are respectively presented. This implementation provides performance of CO_5.

It is known that information about articles printed in issues is also important for system users it is CO_6. For this purpose in model there are such classes as Foreign Article and Own Article.

The articles to conference are sent by the authors. Therefore according to CO_2 it is necessary to store certain information about them. The class Scientist is realized to provide this feature. Classes Degree presented an academic position and Science Rank presented an academic degree are organized for convenience of storage and use in system.

For a completeness of information on the author we need also to know his/her place of operation. The class Working Place modeling a workplace is realized for this purpose. This class is obtained as a result of the organization of the n-ary association. This class helps define the relationship between the author (Scientist), the organization (Organization), the department (Department) and the position (Post). It is necessary to note that the special class Organization Kind storing information about the organization kind has also been proposed.

This class is also implemented in system for convenience of operation with it, after all the organizational and legal form isn't unique, and every time to enter the same value tiresomely. This diagram satisfies CO_3.

Article reviews are executed by the committee members. This meets the requirements containing in CO_4 and CO_7. In order that the user could easily find information about the review and its author, in system there are the class Member presented the committee member and Review modeling the review. These classes are related to each other and thus relationship is organized between the review and the reviewer. There are organized enumerations, which are granted the possible variants of review results (Review Result), the committee types (Committee Type) and the committee positions (Committee Post).

For the published articles authors can be recognized as winners in certain nominations that meets the requirement of CO_8. The class Nomination is provided for storage of this information in the system.

In accordance with CO_9, which states that the system shall provide the ability to learn whether the article be recommended for publication in other editions, there is the class Recommendation In Support Edition. The specified class comprises information on result of the recommendation, its text and also date of saving result.

For users of system it is very important to know the summary analytical information concerning articles which are in certain statuses (on the review, rejected, recommended for publication), and also sent letters and issues intended to participants of committees. For satisfaction of CO_{10} in system the special Analytics module which provides to the user all above-stated information is organized.

Having considered the scheme of the system organization and having read its description, it is possible to draw a conclusion that all system optimality criteria are provided.

After development the new model was compared with the model of the existing system. It was necessary to check, whether all classes had been reflected in the new model. The review indicated that reengineering is executed correctly and in full.

As data which are used by the system, are stored in the relational database, the next step was to check the new database to find the equivalent tables corresponding to the old database tables and the presence in them of all required columns.

VI. IMPORTING DATA

The transfer of data from the old system to the new one becomes the next important and indispensable step after implementation and verification of the new system. This stage is called data import.

Direct copying of the data was not possible. This was caused by several reasons:

1. The current system used a value of type GUID as object identifier, but the new one uses a value of type INT.
2. The model of the new system implies different types of editions, dividing them only on external and internal. While the current system has a clear distinction between the types of editions: journal and conference, which in turn are divided into the own and outside ones.

Accordingly, the data must be imported so as not to lose the relationship between primary and foreign keys and matching between records.

The set of multitable complicated SQL queries were written to solve this problem. Filling the database tables are performed sequentially, starting with simple tables that do not refer to other tables and moving to more complex, involving relationships (links) between the tables.

VII. WRITING BUSINESS LOGIC AND VERIFICATION OF COMPUTABLE VALUES

After all SQL-queries were executed and the data were transferred from the existing system to the new one, it was necessary to check whether all the records had been successfully imported.

This task was not difficult for the tables, the values of which are stored in the database. This has been tested by comparing the number of records for the same tables in two different systems.

Implementation of computable attributes used to calculate the values resulted in the need to write business logic in C #. Correctness was verified by comparing the certain selected values of existing and new systems. If the values of selected records matched, it was considered that the implementation was done correctly.

Each class consists of a number of attributes. In addition, each of them has its own degree of significance. So the absence of a value of one attribute does not affect the system functionality, and on the contrary the absence of the other will make the system flawed.

Thus, to prevent the occurrence of wrong situations when the user interacts with the system the following business rules were identified:

1. Mandatory attribute. This rule is defined for attributes whose values are important for proper system operation. Hence the user cannot save the record, if the value of this attribute will be absent.
2. Unique value. This rule defines the uniqueness of the attribute value or combination of values. The system restricts saving the record if in certain fields the values that have already been saved previously to another record are present.
3. Value or set of values is checked by determining the predicate (logical expression).
4. The absence of rules means that the presence of the value of this attribute is possible, but is not mandatory. The absence of value will not affect the system operation.

All types of described business rules have been implemented in the new system.

CONCLUSION

This article describes the development of an information system for cataloging scientific papers based on reengineering existing information system for conference "Object Systems". Special attention was paid to the purpose and content of the basic steps and performed actions. Development of the system is to finalize the business rules and configuration graphical user interface forms. We plan to develop Web-based applications for the system in the future.

REFERENCES

[1] Ackenhusen John G. Database management system for conference program preparation // IEEE Transactions on Professional Communication, Vol. 32, No. 3, September 1989, 1989, 194-199 pp.

[2] Deponte J., Muller H., Pietrek, G., Schlosser S., Stoltefuss B. Design and Implementation of a System for Multimedia1 Distributed Teaching and Scientific Conferences // International Conference on Virtual Systems and MultiMedia, VSMIM'97, 10-12 Sep 1997, Geneva, Switzerland, 1997, 156-165 pp.

[3] Tu Yong, Peng Jie, Liu Runda, Zhao Hui. Developing a Registration System for the Scientific Content Using DOI // Fourth International Conference on Cooperation and Promotion of Information Resources in Science and Technology, 21–23 November 2009, Beijing, China, 2009, 65-70 pp.

[4] Gong Zhiping, Yang Xiaosong, Wang Shuo, Yang Yanhong. Research on security structure of scientific and technical information system // 2012 (ICSAI), 2012 International Conference on Systems and Informatics, ICSAI 2012, 19-20 May 2012, Yantai, China, 2012, 2660-2663 pp.

[5] Vasenin V. A., Golomazov D. D., Gankin G.M. Architecture, Methods and Tools of the Intellectual System of Thematic Analysis of Scientific Information "ISTINA — Science in MSU" Core / Journal "Software Engineering", № 9. 2014, 3-12 pp. (In Russian)

[6] Oleynik P.P. Using of Object-Relational Mapping Tools for Development of Scientific Conference Managment Information System // Informatsionnye tekhnologii v nauke, ekonomike i obrazovanii: materialy 5-y Vserossiyskoy nauchno-prakticheskoy konferentsii 2-3 sentyabrya 2010 goda / pod. red. O.B. Kudryashovoy; Alt. gos. tekhn. un-t., BTI. – Biysk: Izd-vo Alt. gos. tekhn. uni-ta, 2010, 102-105 pp. (In Russian)

[7] Oleynik P.P., Igumnov E.A., Svechkarev E.A. Designing Scientific Conference Managment Information System // Object Systems – 2010 (Winter session), The Second International Scientific-Practical Conference Proceedings, Russia, Rostov-on-Don, 10-12 November 2010. Edited by Pavel P. Oleynik, 48-51 pp. (In Russian), http://www.objectsystems.ru/files/Sertificates2010_2/Object_Systems_2010_Winter_Session_Proceedings.pdf

[8] Oleynik P.P., Igumnov E.A., Svechkarev E.A. Implementing of a Review Module in the Scientific Conference Managment Information System // Object Systems – 2011: Proceedings of the Third International Theoretical and Practical Conference. Rostov-on-Don, Russia, 10-12 May, 2011. Edited by Pavel P. Oleynik, 26-29 pp. (In Russian), http://objectsystems.ru/files/Object_Systems_2011_Proceedings.pdf

[9] Borodina N.E., Oleynik P.P., Galiaskarov E.G. Reengineering of Object Model by the Example of Information System for Cataloging Scientific Articles for International Conferences // Object Systems – 2014 (Winter session): Proceedings of IX

International Theoretical and Practical Conference (Rostov-on-Don, 10-12 December, 2014) / Edited by Pavel P. Oleynik. – Russia, Rostov-on-Don: SI (b) SRSPU (NPI), 2014, 17-23 pp. (In Russian), http://objectsystems.ru/files/2014WS/Object_Systems_2014_Winter_session_Proceedings.pdf

[10] Oleynik P.P., Computer program "The Unified Environment of Rapid Development of Corporate Information Systems SharpArchitect RAD Studio", the certificate on the state registration № 2013618212/ 04 september 2013. (In Russian).

[11] Pavel P. Oleynik, Olga I. Nikolenko, Svetlana Yu. Yuzefova. Information System for Fast Food Restaurants. Engineering and Technology. Vol. 2, No. 4, 2015, pp. 186-191., http://article.aascit.org/file/pdf/9020895.pdf

[12] Pavel P. Oleynik. Using Multiple Inheritance in Modern Frameworks. Engineering and Technology. Vol. 2, No. 4, 2015, pp. 202-206., http://article.aascit.org/file/pdf/8960766.pdf

[13] Pavel P. Oleynik. Metamodel-Driven Design of Database Applications. Journal of Computer Science Technology Updates, 2015, Vol.2, No. 1, pp. 15-24., dx.doi.org/10.15379/2410-2938.2015.02.01.03, http://www.cosmosscholars.com/images/JCSTU-v1n1/JCSTU-V2-N1/JCSTU-V2N1A3-Oleynik.pdf