Role-Based Access Control Model as Applied to Object-Oriented Applications

Pavel P. Oleynik
PhD, System Architect Software, Aston
OJSC, Associate Professor, Platov South Russian State Polytechnic University (NPI), Rostov-on-Don, Russia,
xsl@list.ru

Abstract—**The article presents an overview of modern approaches to setting up of security and allocation of user access rights in applications of different architecture and the author's approach to allocation of rights to classes, class attributes and objects, complying with specific criteria. It is achieved with the use of hierarchy of classes, the composition and structure of which are described in detail herein. Concluding part of the article contains description of already developed applications, in which the author employed the model.**

Keywords: UML, OOP, OOD, Security Model, DB

I. Introduction

Currently the major part of new applications is developed with the use of object-oriented approach. This paradigm is based on inheritance and it allows for multiple reuse of previously developed items, implemented as classes. Consequently, the cost and duration of development of the whole information system are reduced, which is a key benefit for development of complex software products. Generally, these are multi-user systems and each user category requires only a part of the available information. Therefore, the other part of the information must be inaccessible to a user. This creates a problem of allocation of access rights for multi-user applications. The article presents a model of allocation of access rights for object-oriented applications. The model was developed by the author and employed in many complex applications.

The article has the following structure. Section 1 provides detailed overview of existing works, dedicated to the similar topic. Section 2 contains description of the model of allocation of access rights, used by the author. Section 3 contains real examples of implementation of the described model and allocated user roles. Concluding part of the article sums up the results and outlines the plan of follow-up research.

II. Review of Existing works

Allocation of access rights is one of the main issues, which arise after development of required functional options. For this reason, there is a number of works, representing different approaches. In [1] the authors suggest an approach, called Business-Driven Development, in which the key role is given to setting of security in the application. From perspective of implementation, the authors use a Model-Driven Architecture, MDA, and at the model level, they introduce concepts of business processes and models. Then security policies and templates, establishing particular rules, are defined for such processes and models. The work contains description of principles of allocation of rights at the level of Platform-independent model, and subsequent transformation into a Platform-specific model. As a result, the authors provide a set of templates for allocation of access rights, which may be adjusted if required. This solution was tested with the help of Service-oriented architecture (SOA). For enhancement of efficiency of description of life cycle of a software product and appropriate allocation of access rights, the authors made several suggestions concerning introduction of changes to such software development languages as BPEL and UML. A high number of diagrams, illustrating suggested solutions, as well as numerous fragments of source code in XML language, may be considered strengths of the article.

Paper [2] is more practical and focused. It describes a model of adaptive security for multi-agent information systems, which was used by the authors in a medical information system, called HealthAgents. The authors start with a description of a classic model of rights management, based on roles (Role-based access control, RBAC) and extend it for employment in multi-agent systems. In the article, the authors by means of a UML class diagram present a metamodel, which allows to manage access rights. For interaction with security roles, a basic Subject class was introduced, to which different rights apply. Derived classes represent users, organizations and agents. The analysis of the work shows that a domain-specific approach was used for the description of access rights. For description of the process of application of security policies, the authors depicted an interaction diagram and in XML language gave an example of test definition of access rights for particular users, stored in the system.

Article [3] describes the possibility of modelling of multi-level security, integrated into service-oriented applications. Security is of crucial importance in service-oriented architecture (SOA), which allows to develop different Web-applications. WS-Security web-service, which is controlled by means of SOAP-messages, is responsible for security matters. These messages may be attacked both by anonymous and trusted clients. Moreover, other types of attacks are possible, for example, a so-called denial of service (DoS), which results in depletion of computer resources, which in its turn leads to unavailability of Web-service. In the aforementioned article, security model is viewed as consisting of three levels and close attention is paid to each level. Obtained multi-level architecture of security is shown as a diagram, which illustrates different security domains, as well as the composition and structure of software, installed on each of them. Then different types of attacks, carried out on each of the links, are reviewed. For description of different types of attacks, UML class diagrams are used, which allows to analyze the results obtained by the authors and derive security models from them.

Article [4] presents a framework for description of security model of service-oriented applications (SOA). The authors draw upon the process of modelling of business processes and use BPEL notation for such purpose. In such a case, security is designed along with a model of business processes. The authors state that differences in approaches to work of a Business Analyst and an Expert in security matters result in particular mistakes, which in the long run endanger user data. The authors developed a number of annotations, allowing the Experts in security matters to determine particular security model. Suggested approach is demonstrated through the example of business processes of a service-oriented information system, providing details about academic progress of students. The article contains description of a possible implementation of the framework, its main modules and rules of interaction of experts with the system.

The article [5] is devoted to model-oriented templates (patterns) of application security, which were developed by the authors on the basis of analysis of phases of application development. The authors deal with applications, operating through the Internet. Developed security templates contain descriptions of solutions of typical security issues. Selection of particular pattern depends not only on the situation, but also on the other templates, which were employed earlier. I.e. the relationship among the patterns is taken into consideration. The authors provide the analysis of such relations for the first time ever. The authors suggest a general technology of adjustment of security templates on the basis of the model of rules transformation according to the previously employed patterns. This technology makes it possible to avoid inappropriate use of security templates. Two levels of abstraction are pointed out in the article: 1) analysis phase; 2) design phase. During each of the phases, one particular software module is executed; the structure and functions of the modules are characterized in detail by the authors. In conclusion, the authors present syntax of the language, used for description of rules of transformation of different patterns, which are similar to such languages as SQL, OCL, LINQ. For demonstration of obtained results, the authors describe a test information system, containing information about patients of a hospital. Use case diagram illustrates different categories of users and types of employed security patterns. Then the structure of the template and domain class diagram after employment of this solution are shown as a UML class diagram. This approach was applied to all established templates and workload of manual and automated application was estimated.

The article [6] is devoted to model-oriented approach to description of security, employed in the information system of electronic voting. Reviewed security requirements were presented as functional requirements at the stage of formalization of requirements in the form of UML use case diagram. Then the authors reviewed step-by-step algorithm of detection and implementation of security requirements and gave detailed description of each key element. The article contains description of application architecture and main computational nodes (computers), as well as the role imposed upon them. It allowed to identify possible vulnerabilities and types of attacks, against which it is necessary to defend. With the use of UML sequence diagram, it was possible to represent the authors' approach employed in the course of implementation of security model in the information system of electronic voting.

III. The Permissions Model

Currently classic model of rights management, based on roles (Role-based access control, RBAC) is prevalent. It emerged in the operating systems and has the following structure, shown in Fig. 1.

This system is popular because of its clear architecture, which has the following functions. A large number of roles, represented by a Role class, is created within the system. Particular access rights (permissions), represented by Permission class, are assigned to each role. Permissions are assigned to different objects of the system, which are represented by an Object class. User, described as a User class, is attached to at least one role. The roles themselves may be inherited and that allows to simplify the process of assignment of permissions for the objects.

This scheme is optimal in case of allocation of rights for one type of objects. For example, for allocation of access rights to the objects of file system (files, directories) in the operating system this exact approach is used.

In case of allocation of access rights within applications, written in object-oriented programming languages, a different system is required, because there are several objects, to which the rights may be allocated. Consequently, for design of an optimal system the following optimality criteria (OC) were established; these criteria are functional capabilities, that need to be implemented in a custom solution.

1) allocation of rights to data types, represented in the object-oriented paradigm in the form of classes;
2) allocation of rights to class properties;
3) allocation of rights to objects (instances) of classes.

Fig. 2 shows the structure of an optimal model of allocation of access rights for object-oriented applications.

Let us take a closer look at the figure. A well-developed metamodel of an object system is used for description of objects, to which access rights may be allocated. In our case, it is enough to have information about the classes and attributes (properties) of the classes. For compliance with the selected OC1 a TypePermission class was designed, which allows to allocate access rights to classes. For allocation of rights to class attributes (see OC2) a MemberPermission class was created. ObjectPermission class is used for allocation of access rights to class instances, which complies with the requirements of OC3.

After characterization of the structure and concept of implementation, let us proceed to review of the finished system. Fig. 3 is a class diagram illustrating the model of allocation of access rights for object-oriented applications, implemented by the authors.

Let us take a closer look at Fig. 3. All base classes that exercise key functional capabilities are assigned names ending with a suffix Base. Specifically, SecuritySystemRoleBase and SecuritySystemUserBase classes are considered root classes for representation of security role and system user respectively.

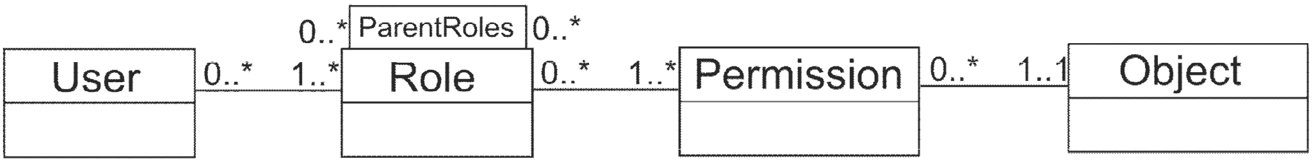

Fig. 1. Classic model of rights management, based on roles (Role-based access control, RBAC)

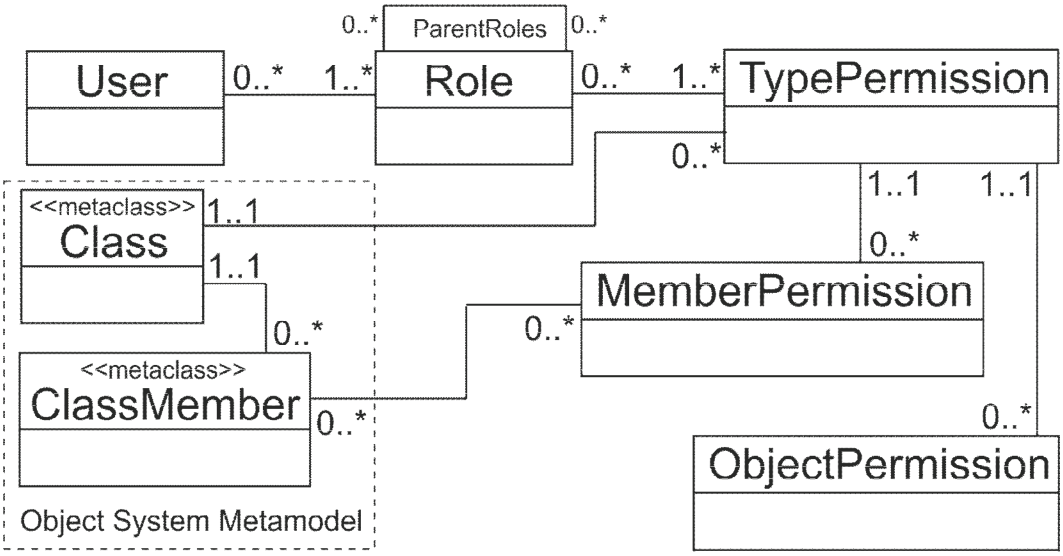

Fig. 2. Structure of an optimal model of allocation of access rights for object-oriented applications

Introduced TypePermissionMatrixItem class is used for indication of data type (class name), for which the allocation of access rights is required. The following types of permissions are provided for classes:

1) AllowCreate – allows a user to create objects (class instances);
2) AllowDelete - allows a user to delete objects (class instances);
3) AllowNavigate – allows displaying of menu item to a user for browsing of class instances;
4) AllowRead - allows a user to view objects of the class;
5) AllowWrite - allows a user to replace class instances with another objects.

SecuritySystemMemberPermissionsObject class allows to define rights for certain properties and to implement a complex security policy, which forbids the user from reading certain class attributes.

SecuritySystemObjectPermissionsObject class is used for allocation of rights for individual objects of a specific class, which comply with particular predicate. This condition is stored in the Criteria property.

UML diagram shows association relationships, which allow to understand interrelations between classes. In conclusion, it is worthwhile to say that presence of a variety of classes allows to unrestrictedly define different types of access rights, and that complies with the previously specified optimality criteria.

IV. Examples of Employment of the Model of Allocation of Access Rights

For implementation of the described model of allocation of access rights, presence of meta-information of the object system is very important. The model itself is physically stored in a relational database according to principles, specified in [7]. In the course of designing of a metamodel, the key concern was to develop the hierarchy of metaclasses, allowing to store information about literal types and about different classes of entities of the subject area [8-10]. Establishment of a well-developed metamodel allows to implement domain-specific approach in the course of design of database applications for different fields of work [11-13]. Articles [14-16] contain detailed description of use of the developed metamodel in the course designing of information systems.

In [17] the previously described security model was used for allocation of access rights within information system designed for arrangement of scientific conferences, which was many times used for holding of the International Research and Practice Conference referred to as “Object systems”(objectsystems.ru). Special attention was given to security matters at the design stage.

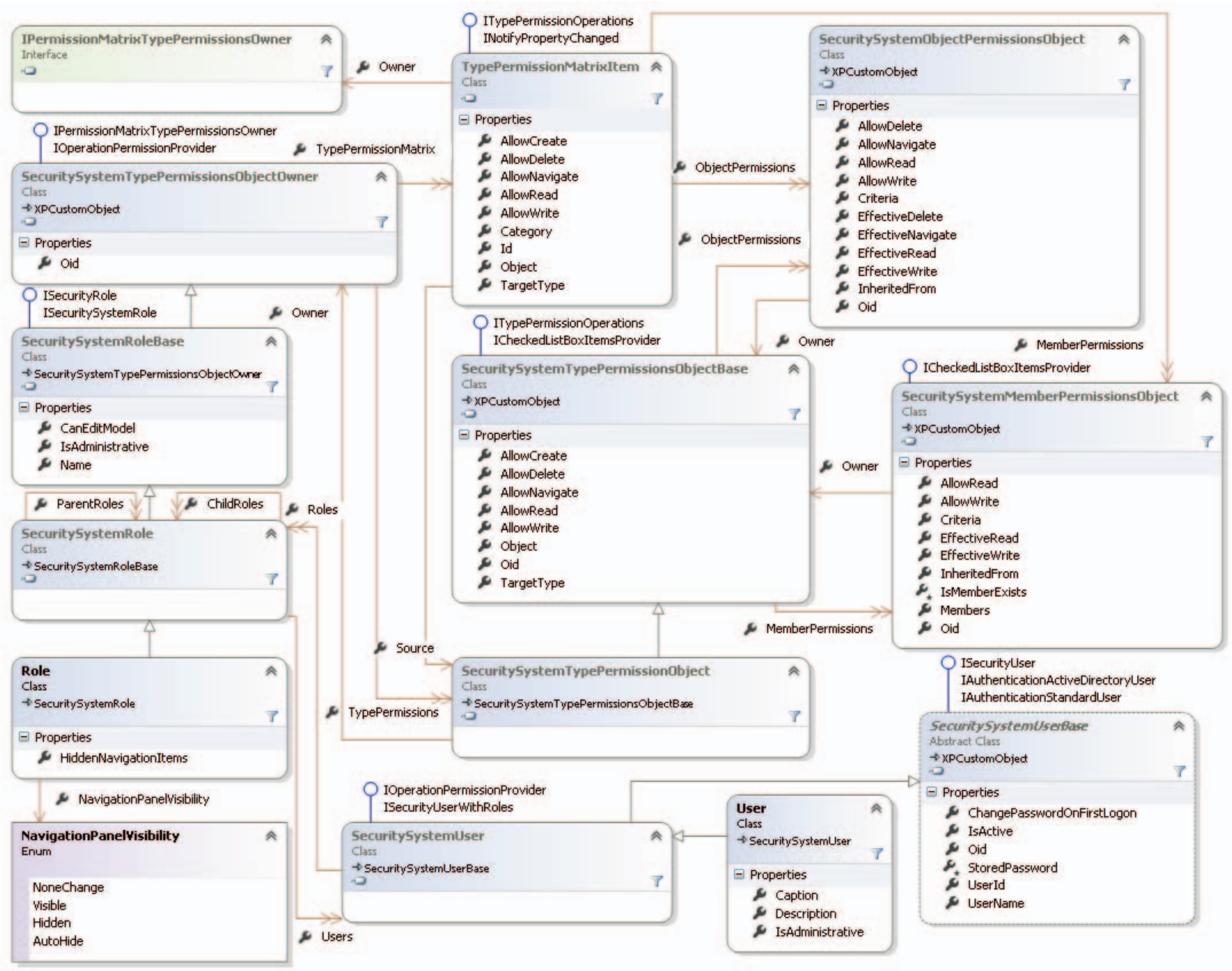

Fig. 3. UML class diagram of the implemented model of allocation of access rights

For this purpose, in the first place the following roles of system users were defined:

1. **Conference organizer** is considered the main actor and system user. His duties are as follows:

1) registration of publications;
2) assignment of reviewer;
3) verification of authors' corrections in accordance with the reviewer's comments;
4) verification of payment;
5) formatting and making up of a conference proceedings;
6) delivery of the conference proceedings and certificates to the authors of articles.

2. **Author.** Conference is not possible without authors as they write articles and submit them to the conference. Authors are responsible for correction of articles in accordance with reviewer's comments on the article, and for payment of the conference fee where necessary.

3. **Reviewer.** Checks the author's article and evaluates its quality. Reviewing includes writing of review of the article with specification of comments and suggestions on improvement thereof; summarizing the results of the review – whether to accept the article for printing, reject it or return for correction. During the makeup of the conference proceedings, reviewers award prizes to the best articles, submitted to the conference. Generally, there are several reviewers.

Based on this information, classes were established and types of access required for different roles were determined. Then class instances, shown in Fig. 3 were created.

Paper [18] contains description of information system of a beauty salon. In the course of research of business logic of the subject area it became clear, that it is necessary to implement within the system a lot of different calculations related to financial activities, expenditures and profitability of the salon. This information may be available only to the owner of the salon. In these circumstances, the following roles were established:

1. **Specialist of the salon.** His main duty is to provide services to a client. Therefore each specialist may only view (read) main reference objects of the system, such as: Working hours, Booking / Visiting, Schedule of visiting, Client, Vacation / Sick leave / Compensatory leave / Unauthorized absence from work, Service, Product, Certificate, Price, Percent, Specialist, Categories of the Specialist, Category of the service, Products in stock, Duty schedule, Operation time;

2. **Manager of the salon.** His main duty is to control activities of the salon, i.e. a manager registers clients and watches over the progress of execution of works by the specialist of the salon. Within the system, they are given authority to add/edit/delete data from reference objects: Schedule of visiting, Client, Specialist, Working hours, Booking / Visiting, Vacation / Sick leave / Compensatory leave / Unauthorized absence from work, Service, Product, Certificate, Discount, receipt of the product, Inventory control, Price, Stock, Percent, Category of the specialist, Category of the client, Category of the service, Document, movement of products, Products in stock, Sales, Salon, Duty schedule, Operation time;

3. **Owner of the salon.** Has the same rights as the Manager of the salon. In addition, he may view information from consolidated forms, such as: Salary, Profit, Profitability. Besides the owner of the salon may create users in the system and add them only to existing roles.

Articles [19-24] are devoted to architecture of the information system of fast-food restaurants. Key feature of applications of this kind is that they are used in complex queue systems with a large number of customers. For such software products, service time is critical, therefore graphic interface must be ergonomic. Frequently all-in-one computers with a touchscreen are used as a hardware platform. That is why in such applications attention is given both to development of graphic user interface and to principles of setting up of application security. The following roles were established:

1) **Waiter.** His main duty is to create orders, add products, bought by the customers to the orders, and complete payments;
2) **Cashier.** Cashier may not create new orders, but may delete wrong orders, view all orders for the current and previous shift, and complete payment of orders.
3) **Manager.** His main duty is to generate consolidated reports in relation to work for a shift, and to add new waiters and cashiers to the system;
4) **Stock manager.** His main duty is to add new dishes to the system.

Also in the course of design of each aforementioned application a separate role of system administrator was defined, which (the former) was granted powers to adjust rights for existing roles and create new roles. In fact, this role matches the powers of a system administrator of a domain of Windows operating system.

V. Conclusions and follow-up studies

Presented description shows, that the developed model of allocation of access rights may be successfully implemented for software products of different application areas, i.e. it is orthogonal in relation to them. Currently several applications, for which security is the main concern, are being designed and developed. This will allow full-scale testing of the developed model and refining it in accordance with identified faults.

References

[1] Nagaratnam N., Nadalin A., Hondo M., McIntosh M., Austel P. Business-driven application security: from modeling to managing secure applications. *IBM Systems Journal*, Volume 44 Issue 4, 2005, 847-867 pp.

[2] Xiao L., Peet A., Lewis P., Dashmapatra S., Saez C., Croitoru M., Vicente J., Gonzalez-Velez H., Lluch i Ariet M. An Adaptive Security Model for Multi-agent Systems and Application to a Clinical Trials Environment. *31st Annual International Computer Software and Applications Conference, COMPSAC 2007*, 24-27 July 2007, Beijing, China, 2007, 261-268 pp.

[3] Fengyu Zhao, Xin Peng, Wenyun Zhao. Multi-Tier Security Feature Modeling for Service-Oriented Application Integration. *Eighth IEEE/ACIS International Conference on Computer and Information Science, ICIS 2009*, 1-3 June 2009, Shanghai, China, 2009, 1178-1183 pp.

[4] Saleem M.Q., Jaafar J., Hassan M.F. Model Driven Security Framework for Definition of Security Requirements for SOA Based Applications. *2010 International Conference on Computer Applications and Industrial Electronics (ICCAIE)*, 5-8 Dec. 2010, Kuala Lumpur, 2010, 266-270 pp.

[5] Shiroma Y., Washizaki H., Fukazawa Y., Kubo A., Yoshioka N. Model-Driven Security Patterns Application Based on Dependences among Patterns. *ARES '10 International Conference on Availability, Reliability, and Security*, 15-18 Feb. 2010, Krakow, Poland, 2010, 555-559 pp.

[6] Salini P., Kanmani S. Application of Model Oriented Security Requirements Engineering Framework for Secure E-Voting. *2012 CSI Sixth International Conference on Software Engineering (CONSEG)*, 5-7 Sept. 2012, Indore, 2012, 1-6 pp.

[7] Oleynik P.P. Predstavlenie metamodeli ob"ektnoy sistemy v relyatsionnoy baze dannykh. *Izvestiya vysshikh uchebnykh zavedeniy. Severo-Kavkazskiy region. Spetsvypusk «Matematicheskoe modelirovanie i komp'yuternye tekhnologii»,* 2005. - S. 3-8.

[8] Oleynik P.P. Organizatsiya ierarkhii atomarnykh literal'nykh tipov v ob"ektnoy sisteme, postroennoy na osnove RSUBD. *Programmirovanie*, 2009, № 4. - S. 73-80

[9] Oleynik P.P. Implementation of the Hierarchy of Atomic Literal Types in an Object System Based of RDBMS. *Programming and Computer Software*, 2009, Vol. 35, No.4, pp. 235-240.

[10] Oleynik P.P. Class Hierarchy of Object System Metamodel. *Object Systems – 2012: Proceedings of the Sixth International Theoretical and Practical Conference*. Rostov-on-Don, Russia, 10-12 May, 2012. Edited by Pavel P. Oleynik. 37-40 pp. (In Russian), http://objectsystems.ru/files/2012/Object_Systems_2012_Proceedings.pdf

[11] Oleynik P.P. Domain-driven design of the database structure in terms of object system metamodel. *Object Systems – 2014: Proceedings of the Eighth International Theoretical and Practical Conference* (Rostov-on-Don, 10-12 May, 2014) / Edited by Pavel P. Oleynik. – Russia, Rostov-onDon: SI (b) SRSPU (NPI), 2014. - pp. 41-46. (In Russian), http://objectsystems.ru/files/2014/Object_Systems_2014_Proceedings.pdf

[12] Oleynik P.P. Using metamodel of object system for domain-driven design the database structure. *Proceedings of 12th IEEE East-West Design & Test Symposium (EWDTS'2014)*, Kiev, Ukraine, September 26 – 29, 2014, DOI: 10.1109/EWDTS.2014.7027052

[13] Oleynik P.P. Unified Metamodel of Object System. *Object Systems – 2015: Proceedings of X International Theoretical and Practical Conference* (Rostov-on-Don, 10-12 May, 2015) / Edited by Pavel P. Oleynik. – Russia, Rostov-on-Don: SI (b) SRSPU (NPI), 2015., http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf

[14] Oleynik P.P. The Elements of Development Environment for Information Systems Based on Metamodel of Object System. *Business Informatics*. 2013. №4(26). – pp. 69-76. (In

Russian), http://bijournal.hse.ru/data/2014/01/16/1326593606/1BI%204(26)%202013.pdf

[15] Oleynik P.P., Kurakov Yu.I. The Concept Creation Service Corporate Information Systems of Economic Industrial Energy Cluster. *Applied Informatics*. 2014. №6. 5-23 pp. (In Russian)

[16] Kurakov Yu.I., Oleynik P.P. Implementation method a unified information system of economic production and energy cluster in coal industry. *Mining informational and analytical bulletin*. №5/2015. 260-273 pp.

[17] 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

[18] Kozlova K.O., Borodina N.E., Galiaskarov E.G., Oleynik P.P. Domain-Driven Design of Information System of a Beauty Salon in Terms of Unified Metamodel of Object System. *Object Systems – 2015: Proceedings of X International Theoretical and Practical Conference* (Rostov-on-Don, 10-12 May, 2015) / Edited by Pavel P. Oleynik. – Russia, Rostov-on-Don: SI (b) SRSPU (NPI), 2015. (In Russian), http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf

[19] Oleynik P.P, Yuzefova S.Yu., Nikolenko O.I. Experience in Designing an Information System for Fast Food Restaurants. *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. – pp. 12-16. (In Russian), http://objectsystems.ru/files/2014WS/Object_Systems_2014_Winter_session_Proceedings.pdf

[20] Nikolenko O.I., Oleynik P.P, Yuzefova S.Yu. Prototyping and Implementation of Graphical Order Form for the Information System of Fast Food Restaurants. *Object Systems – 2015: Proceedings of X International Theoretical and Practical Conference* (Rostov-on-Don, 10-12 May, 2015) / Edited by Pavel P. Oleynik. – Russia, Rostov-on-Don: SI (b) SRSPU (NPI), 2015. (In Russian), http://objectsystems.ru/files/2015/Object_Systems_2015_Proceedings.pdf

[21] 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

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

[23] Domain-Driven Design of Information System for Queuing System in Terms of Unified Metamodel of Object System. *International Journal of Applied Engineering Research*, ISSN 0973-4562, Volume 10, Number 15 (2015), pp. 35229-35238

[24] Pavel P. Oleynik, Nikolay V. Kuznetsov, Edward G. Galiaskarov, Natalia E. Borodina. Model-Driven Design and Implementation of Scientific Data Management *Information System. International Journal of Applied Engineering Research*, ISSN 0973-4562, Volume 10, Number 15 (2015), pp. 35239-35246

[25] Pavel P. Oleynik, Sergey M. Salibekyan. The Approaches to Implementation of Patterns of Static Object Models for Database Applications: Existing Solutions and Unified Testing Model. *International Journal of Applied Engineering Research* ISSN 0973-4562 Volume 10, Number 24 (2015) pp. 45513-45516.

[26] Pavel P. Oleynik, Sergey M. Salibekyan. Implementation Patterns of Object Static Models for Database Applications: Classical ORM-Patterns and Object-Attribute Approach. *International Journal of Applied Engineering Research* ISSN 0973-4562 Volume 10, Number 24 (2015) pp. 45559-45566.