Харківський національний
університет радіоелектроніки

Kharkov National
University of Radio Electronics

Державне підприємство
"Південний державний
проектно-конструкторський
та науково-дослідний інститут
авіаційної промисловості"

State Enterprise
"Southern National Design
&
Research Institute
of Aerospace Industries"

СУЧАСНИЙ СТАН НАУКОВИХ ДОСЛІДЖЕНЬ ТА ТЕХНОЛОГІЙ В ПРОМИСЛОВОСТІ

INNOVATIVE TECHNOLOGIES AND SCIENTIFIC SOLUTIONS FOR INDUSTRIES

№ 3 (9), 2019

No. 3 (9), 2019

Щоквартальний
науковий
журнал

Quarterly
scientific
journal

Харків
2019

Kharkiv
2019

INNOVATIVE TECHNOLOGIES AND SCIENTIFIC SOLUTIONS FOR INDUSTRIES

No. 3 (9), 2019

ЗАСНОВНИКИ
Харківський національний університет радіоелектроніки,
Державне підприємство "Південний державний проектно-конструкторський та науково-дослідний інститут авіаційної промисловості"

АДРЕСА РЕДАКЦІЇ:
Україна, 61166, м. Харків, проспект Науки, 14
Телефон: +38 (057) 704-10-51
Інформаційний сайт: http://itssi-journal.com
E-mail редколегії: journal.itssi@gmail.com

ESTABLISHERS
Kharkiv National University of Radio Electronics,
State Enterprise "National Design & Research Institute of Aerospace Industries"

EDITORIAL OFFICE ADDRESS:
Ukraine, 61166, Kharkiv, Nauka Ave, 14
Phone: +38 (057) 704-10-51
Information site: http://itssi-journal.com
E-mail of the editorial board: journal.itssi@gmail.com

Журнал включено до "Переліку наукових фахових видань України, в яких можуть поблікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора і кандидата наук" наказом Міністерства освіти і науки України від 16.07.2018 №775 (додаток 7).

Затверджений до друку Науково-технічною Радою Харківського національного університету радіоелектроніки (Протокол № 7 від 05 вересня 2019 р.).

Свідоцтво про державну реєстрацію журналу Серія КВ № 22696-12596Р від 04.05.2017 р.

СУЧАСНИЙ СТАН НАУКОВИХ ДОСЛІДЖЕНЬ ТА ТЕХНОЛОГІЙ В ПРОМИСЛОВОСТІ

№ 3 (9), 2019

ЗМІСТ

Економіка та управління підприємством

5 ***Аванесова Н. Е., Марченко О. В., Колодяжна Т. В.***
Сучасні тенденції забезпечення кадрової безпеки підприємств (eng.)

12 ***Андрєєва Т. Є., Гетьман О. О., Терещенко Д. А.***
Вдосконалення методичних засад управління організаційними змінами (рус.)

22 ***Белхадер А.***
Кількісна оцінка оптимальної політики в ендогенній моделі: теоретичний аналіз (eng.)

34 ***Момот Т. В., Родченко С. С.***
Оцінка фінансової безпеки банків в умовах макроекономічної нестабільності (eng.)

44 ***Хілуха О. А.***
Особливості впровадження електронного навчання в контексті корпоративного розвитку персоналу

53 ***Чех Н. О., Панов В. В., Шпілько В. Л., Савенко К. С.***
Поняття регуляторного комплаєнсу: значення та місце в системі економічної безпеки підприємства (на прикладі підприємств водопостачання та водовідведення)

Інформаційні технології

62 ***Великодний С. С.***
Реінжиніринг відкритої програмної системи тривимірного моделювання BRL-CAD (eng.)

72 ***Давидовський Ю. К., Рева О. А., Артюх О. В., Косенко В. В.***
Моделювання параметрів навантаження комп'ютерної мережі протягом заданого періоду часу (eng.)

81 ***Єлізєва А. В., Артюх Р. В., Персіянова О. Ю.***
Цільові та системні аспекти програми розвитку транспортної інфраструктури (eng.)

91 ***Литвиненко Д. П., Малєєва О. В.***
Комплексний метод балансування та гармонізації інтересів стейкхолдерів у проектах розвитку транспортних систем

99 ***Пітерська В. М., Логінов О. В., Логінова Л. В.***
Механізм формування ефективного портфелю наукових проектів закладу вищої освіти (eng.)

109 ***Чайковський І. В.***
Концептуальна модель формування рішень з контролю і аналізу перевезень і роботи флоту (рус.)

Машинобудування, автоматизація та робототехніка

118 ***Шаповалов В. Ф, Пермяков О. А., Клочко О. О.***
Моделювання дуплексної фрези і умов формоутворення обробки колісних пар магістральних електровозів, тепловозів, електросекцій (eng.)

Телекомунікаційні системи та комп'ютерні мережі

127 ***Лемешко О. В., Євдокименко М. О., Єременко О. С.***
Модель швидкої QoS-перемаршрутизації трафіку даних в інфокомунікаційних мережах (eng.)

135 **Алфавітний показчик**

INNOVATIVE TECHNOLOGIES AND SCIENTIFIC SOLUTIONS FOR INDUSTRIES

No. 3 (9), 2019

CONTENTS

Economics and Enterprise Management

5 ***Avanesova N., Marchenko O., Kolodyazhna T.***
Modern trends in personnel safety of enterprises

12 ***Andrieieva T., Hetman O., Tereshchenko D.***
Improvement of the methodical basis of organizational change management

22 ***Ahmed Bellakhdhar***
Quantifying optimal policy in an endogenous model: a theoretical analysis

34 ***Momot T., Rodchenko S.***
Assessment of financial security of banks in conditions of macroeconomic instability

44 ***Khilukha O.***
Features of implementation of e-learning in the context of corporate development of personnel

54 ***Chekh N., Panov V., Shpilko V., Savenko K.***
The concept of regulatory compliance: the meaning and place in the economic security system of the enterprise (example of water supply and supply)

Information Technology

62 ***Velykodniy S.***
Reengineering of open software system of 3D modeling BRL-CAD

72 ***Davydovskyi Yu., Reva O., Artiukh O., Kosenko V.***
Simulation of computer network load parameters over a given period of time

81 ***Yelizyeva A., Artiukh R., Persiyanova E.***
Target and system aspects of the transport infrastructure development program

91 ***Lytvynenko D., Malyeyeva O.***
A comprehensive method of balancing and harmonizing the interests of stakeholders in transport systems development projects

99 ***Piterska V., Lohinov O., Lohinova L.***
Mechanism for forming an effective portfolio of research projects of institution of higher education

109 ***Tchaikovsky I.***
Conceptual model of decision making for control and analysis of transportation and operation of the fleet

Mechanical Engineering, Automation & Robotics

118 ***Shapovalov V., Permyakov A., Klochko A.***
Modeling of a duplex miller and conditions of formation of treatment of wheel pairs of main electric cards, dual cars, electro sections

Telecommunication Systems & Computer Networks

127 ***Lemeshko O., Yevdokymenko M., Yeremenko O.***
Model of data traffic QoS fast rerouting in infocommunication networks

135 **Alphabetical index**

The author is responsible for the accuracy of the facts, quotations and other information

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2019. No. 3 (9)

UDC 004.4'24 DOI: https://doi.org/10.30837/2522-9818.2019.9.062

S. Velykodniy

REENGINEERING OF OPEN SOFTWARE SYSTEM OF 3D MODELING BRL-CAD

Computer Graphics is an up-to-date industry in the design and application of rapidly evolving computing systems. The **subject** of the work is designing a graphical user interface. The **purpose** of the work is to perform reengineering (evolutionary improvement while maintaining the positive qualities that are confirmed by the operation) of an open-source three-dimensional software system with inheritance of full functionality and principles of operation. BRL-CAD is a specialized open source cross-platform system that is a powerful 3D computer aided design system for bulk body modeling. The software system includes an interactive geometric editor, parallel ray tracing, rendering and geometric analysis. **Objectives**: To summarize the results of experimental studies at the level of representation of classes, components and use cases, which must be submitted using a unified modeling language – UML, with the processing and interpretation of results at the level of CASE-tool; to analyze the source code translation of the redesigned BRL-CAD graphical user interface. **Methods**. The process of designing or redesigning new software products is ineffective without using the UML methodology, but with its adherence – the speed of development is increased several times over. In this paper, we use the UML 2.5 methodology using the Enterprise Architect 14.0 CASE toolkit. **Results**. The main focus is on three diagrams: classes, use cases and components. Based on these diagrams, code generators and programmers continue to work, while other (auxiliary) diagrams are intended to explain some of the complex specifications of the project, which does not, however, diminish their relevance within the project. The present study summarizes the results of experimental studies; source code translation results are analyzed and summarized, the main one being the reduction in the complexity of creating open source software using the BRL-CAD example. **Conclusions**. BRL-CAD is acceptable for the experienced designer, but for the beginner or student, its application process will be very complicated. A thorough analysis of the environment revealed the presence of two modules contained in the structure of the system that help the potential user to quickly design and model. Also fundamental to the package is the ability to support the design and analysis of visual models based on complex objects, consisting of a large set of graphic primitives. The powerful side of the system is the extraordinary speed of visual means, ray tracer and rendering. Compared to analogs, it can be state that the visualization process is one of the fastest among the existing ones.

Keywords: reengineering; software system; computer graphics; 3D modeling; diagram; CASE tool; class.

Introduction

Nowadays, there are a large number of software tools that perform a large number of specialized tasks. Some of them are tied to only one industry, others are used in large numbers, but the trend is by specializing in software as a whole.

One of the important components of computer aided design (CAD) is computer graphics, which is a collection of tools and techniques used to input, convert, and output from specialized graphic information environments. Computer Graphics is an up-to-date area of design and application of advanced computer systems. The term "computer graphics" means the computer processing of information and the output of results in the form of various graphic images. The data needed to display the results in a graphical format is created on the basis of graphical information. Particular interest in computer graphics began to emerge in connection with the intensive development and implementation of CAD not only in engineering, instrumentation, radio electronics, interior design, but also in other industries and training.

A distinctive part of the tasks of computer graphics is the processing of graphic databases (GDB), which are essentially "ordinary" databases, but which are based on mathematical algorithms for image recovery based on the generated statistical coordination data. Not every SAP is able to do this, but current trends simply require it. A large number of software systems (SS) are developed with a wide range of modeling characteristics, BRL-CAD is one of such SS.

Formulation of the problem

Graphic information is the most capacious and visual representation of a large amount of information, however, the practical application of machine graphics has long been constrained by the lack of appropriate equipment and mathematical support. The logic and formality of computer models allows us to identify the main factors that determine the properties of the object under study (or a whole class of objects), in particular, to investigate the response of a physical system, which is modeled on changes in its parameters and initial conditions.

Building a computer model is based on an abstraction of the specific nature of the phenomena or object under study and consists of two stages: first, creating a qualitative and then a quantitative model. Computer simulation itself is the conduct of a series of computational experiments on a PC, the purpose of which is to analyze, interpret and compare the simulation results with the actual behavior of the object of study and, if necessary, further refinement of the model.

The solution of the problems is impossible without deep penetration into the physical essence of the phenomena studied, the development and improvement of the relevant theoretical provisions, implementation of the achieved results in production. Geometric methods have long and successfully been used in many industries. An important role here should play new methods of geometric modeling and their implementation in computer graphics systems, which will allow to solve the problems of special disciplines.

Designing engineering, industrial, civil engineering and radio electronics is entering a new phase of its development, when, along with the increasing complexity

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2019. № 3 (9) ISSN 2524-2296 (online)

of projects, shortening the design time and reducing the number of designers should be ensured largely by the automation of design and computerization of engineering work.

One of the main advantages of 3D modeling is the rapid formation of drawings. It is possible to use the results of modeling in the further stages of product development - this is another advantage of solid state modeling [1]. Open-source software is software for which (source) software is available that provides the best conditions for learning about such software and for making further changes (improvements) to it [2].

Quite often, this concept is considered to be free software, which is not absolutely correct. The most significant difference is that free software licenses stipulate that all further modified versions of such software must also be distributed as free software, while most open source licenses grant complete freedom to the authors of the modified versions. As a result, free software is always open source software, but the opposite is not always true.

Analysis of research and publications

When conducting research and publication analysis, particular attention was paid, in a scientific sense, to the methods used in dealing with complex graphical objects. The basic scientific geometrical methods used in graphical modeling of objects, processes, phenomena in engineering and tendencies of their development are considered below [3]. For the direction of scientific research solid modeling of objects that are formed (and change in time) under the influence of various external factors is selected.

Computer modeling is one of the effective methods of studying any complex systems that can be visualized. Computer models are easier and more convenient to explore because of their ability to perform so-called computational experiments [4], in cases where real experiments are complicated by financial or physical obstacles or may produce unpredictable results [5].

3D Object Description is a three-dimensional representation of an object. As a rule, these measurements are represented as X, Y, and Z coordinates. It is possible to have data with identical X and Y coordinates with a different Z coordinate. For example, 3D is used for digital representation of ocean flows [6].

Solid-state modeling is the most sophisticated and reliable method of creating a copy of a real object, a natural way of expressing the essence of a product [7].

Rendering in computer graphics is the process of obtaining a model image using a computer program [8]. Here, the model is a description of three-dimensional objects (3D) in a strictly defined language or as a data structure. Such a description may include geometric data, position of the observer point, information about lighting. An image is a digital bitmap. Typically, rendering refers to the imposition of a texture on an already-finished solid-works model in mechanical engineering [9] and on a framework in engineering graphics.

Ray tracing in computer graphics is a way of creating an image of three-dimensional objects or scenes by tracking the progress of a light beam through a point of screen and simulating the interaction of that beam with imaginary objects to be displayed [10]. This method allows to create extremely realistic images, usually of a much higher quality than the typical Scanline algorithm or Ray casting, but has a much higher computational complexity. For this reason, ray tracing algorithms are used where there are no significant restrictions on rendering time.

Boundary representation is a description of the boundaries of an object or the absolute analytical task of the faces that describe the body [11]. This method allows to create a high-quality image of a geometric solid to establish mutual alignment, you need to specify the borders or contours of objects, as well as sketches of different types of objects, and specify the lines of connection between these species. Methods for the determination of complex contours and vectorization of raster models were considered in [12].

There are methods of boundary (B-Rep) and constructive (C-Rep) representation for the creation of GDB [13]. In a B-Rep system, models are built from solid-state primitives. These primitives are determined by size, orientation, shape, and point of attachment. C-Rep construction tools are Boolean operations; they are based on algebraic set theory [14]. The most commonly used operations are difference, intersection, and union. Each of these methods for creating volumetric models has pros and cons compared to others. For systems with a C-Rep view, the advantage is the primary model formation. In addition, this presentation provides a more convenient description of the models in the GDB. The B-Rep method is relevant in the formation of complex structures that are very difficult to reproduce with the C-Rep method. The advantage of B-Rep systems is to simply change the boundary view into a frame model and reverse it. The reason is that the description of the boundaries is similar to the description of the frame model. For example, the design of injection molding and molding is a traditional area of solid, three-dimensional motion simulation. The most obvious difference from two-dimensional drawing is the accurate creation of a large-scale computer model.

Highlighting the unsolved parts of a common problem. The goal of the work

BRL-CAD is a specialized open source cross-platform system. It is a powerful 3D CAD for bulk body modeling by CSG. This CAD includes an interactive geometric editor, parallel ray tracing, rendering, and geometric analysis. BRL-CAD has been in development for close to 40 years and has been deployed in the US military. The whole BRL project works from source code, so it can be used on any platform: GNU/Linux, MacOS, Solaris and Windows.

Here are some of the defining features of open source software and design technologies for GDB that we will rely on in the following sections of the overall study. Raw code (usually just "raw", also "source code", "program code", "program text", English: "source code") – any set of instructions or ads written in programming

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2019. No. 3 (9)

language and in a form that can be read by a person. The source code allows the programmer to communicate with the computer using a limited set of instructions. A program code is a collection of files that are required to be converted from readable form to some type of computer executable code. There are two possible ways to execute raw code: translate to machine code using a compiler (designed for a specific computer architecture) or run directly from text using an interpreter.

One of the first CAD with such characteristics appeared because in 1979, ballistic research laboratory U.S. army (U.S. Army Ballistic Research Laboratory (BRL), now – United States Army Research Laboratory, expressed an urgent need for tools and means that could help in computer simulation and engineering analysis of combat weapon systems (tanks, missiles, planes, etc.) and their working conditions. When none of the CADs that existed at that time, was not ready to achieve this goal, the developers of BRL started to collect a Suite of utilities capable of interactively review and edit trees for geometric models. Programmers have begun to develop their own Suite of applications designed to display, edit, and combine geometric models. In this way was established BRL-CAD application package for solid modeling (constructive solid geometry – CSG).

The first public release was made in 1984. In December 2004, BRL-CAD became an open source project. It is very important that BRL-CAD is licensed under the * BSD and GNU LGPL licenses. Since then, this CAD has been evolving and new opportunities are emerging.

Today, thanks to about a million lines of C code, BRL-CAD has become a powerful graphical simulation package that came the use of more than 2 thousand organizations around the world. BRL-CAD simultaneously supports two methods of user interaction: command-line and graphical user interface (GUI). The system also supports a variety of geometric tools with graphical information: a large set of traditional CSG primitive solids (APSDI, cones, tori), as well as a clear solid (from private collections) of the uniform b-spline surfaces nonuniform rational b-splines (NURBS), n-diverse geometry, the faceted mesh, and the like. All geometric objects may be combined using Boolean set-theoretic operations, including the CSG-unions and intersections.

The most important feature of a system is its ability to design and analyze realistic models based on complex objects consisting of a large set of primitive shapes. Boolean operations are used to construct complex objects: join, subtract, and intersect. Another powerful side of BRL-CAD is the speed of rendering tools and the ray tracer, which is one of the fastest among the existing ones. Finally, BRL-CAD users can design models with high precision, from subatomic to galactic scales on the principle of "seeing all the details all the time".

However, one of the big drawbacks of BRL-CAD, which is a huge problem for the user, is the lack of a clear graphical user interface for the product, which has been anticipated and improved for several years in the perspective of the developer corporation of this software. In addition, the very linguistic support for the submission of GDB ("C") in the BRL-CAD CAD system requires the transition (re-engineering) into high-level languages ("C++" or "C#") [15].

From here it is possible to formulate the *purpose of the work* – to perform reengineering (evolutionary improvement while maintaining the positive qualities, which is confirmed by the operation) of the open-source three-dimensional design with inheritance of full functionality and principles of operation.

Achieving the goal is possible when performing a set of *established research objectives*: to summarize the results of experimental studies at the level of representation of classes, components and use cases (UC), which must be submitted using a unified modeling language – UML, with the processing and interpretation of results at the CASE level; analyze the source code translation of the redesigned BRL-CAD.

Materials and methods

Today, the process of designing or redesigning new software products is not efficient without the use of the UML methodology, but with its observance – the speed of development is increasing several times [16]. To achieve this goal, use the UML 2.5 methodology [17] using the Enterprise Architect 14.0 CASE toolkit.

Use case diagram. In order to disassemble and create a chart, you need to decide on the actors and precedents of the chart, so let's start by looking at the main "exe" files of the software. There are only two of them: "Archer" and "MGED". These will be our actors. We will deal with adjusting their specifications later, and it is now important to determine the precedents that fit each one. To do this, we run each executable file and look at their capabilities. "MGED" is the main software module responsible for designing, modifying and tracing beams. It looks like that at a fig.1.

In addition to the console and GUI (graphical user interface) there are additional features in the top menu, we list the main:

- File – basic commands, including ray tracing;
- Edit – is responsible for changing simple and complex shapes using different methods;
- Create – create simple figures and complex hierarchies of simple figures;
- View – change the angle and viewpoint;
- Settings – general settings for work;
- Modes – modules (one of the unofficial mandatory features of free software), which are paid and free;
- Tools – tools for working with figures and graphics;
- Help – help files and manuals.

These are all functions that will be the first level of precedents in the diagram, so we will create them immediately (fig. 2).

Сучасний стан наукових досліджень та технологій в промисловості. 2019. № 3 (9)
ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

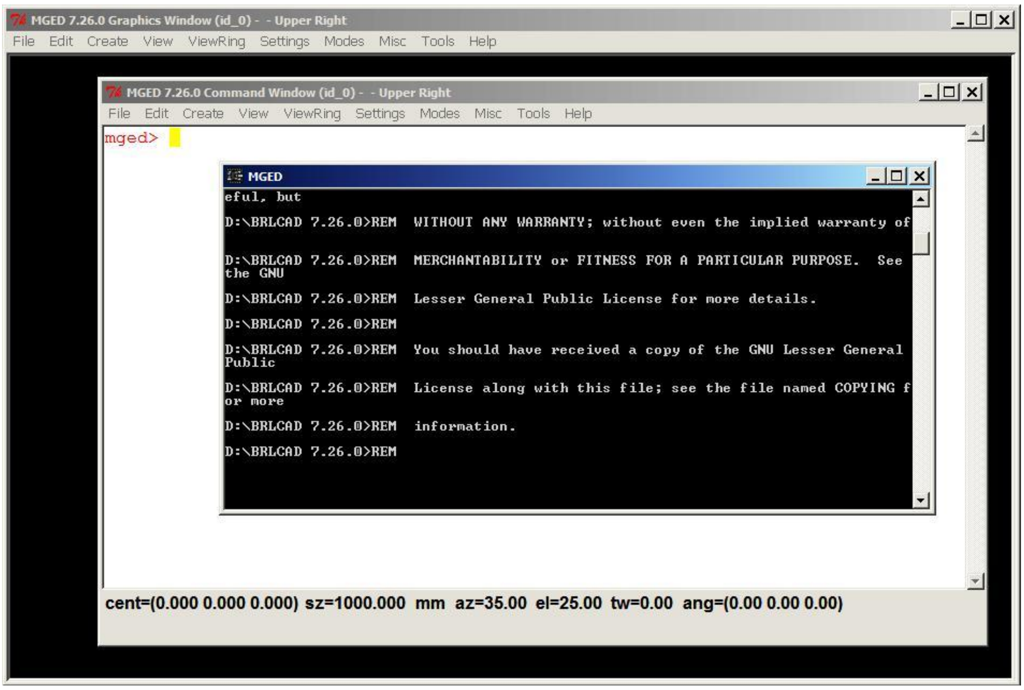

Fig. 1. Interface "MGED"

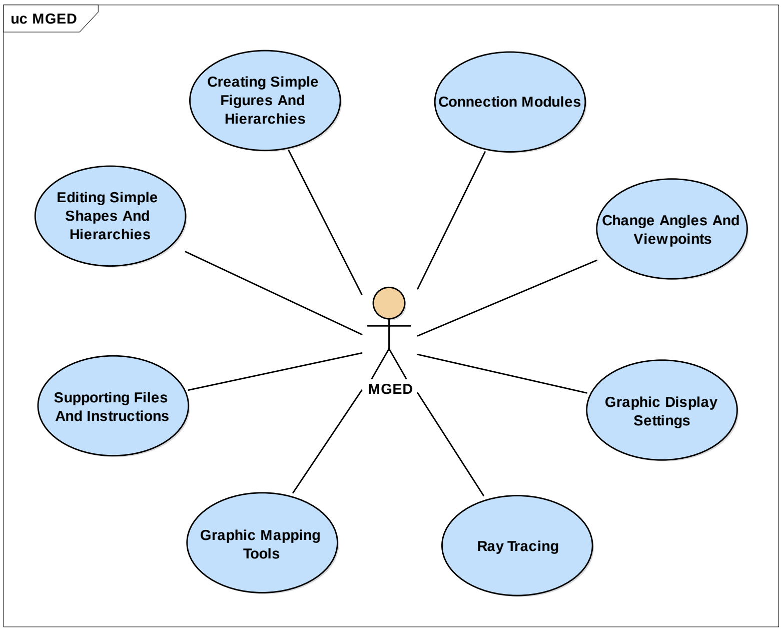

Fig. 2. The first level of precedents for "MGED"

At this stage of the work it is important to determine the type of actor and precedents and their relationship with each other [18]. All objects are user-interacting modules, and they have special stereotypes called "business use case" for precedent and "business actor" for diagram. In this particular case, we will show the business stereotype only for the actor, so it will be logical and clear from the links that all the following precedents are connected with it. We have decided on the stereotypes, so we go for connections.

The first level precedents are the user capabilities of this software product. There is a special type of use in the connections of this diagram, the essence of which is that it shows the use of the capabilities of any actor or precedent [19]. It describes this level in the most detail, so it shows that precedents of the "simple figure" type are one of the advanced features of the "MGED" actor. The only difference between the precedents will be "External modules" – they are not required to work, so we will use another type of communication called "subscribe", which means "description". In essence, it helps to describe any module and in our case this module is the actor "MGED". After the arrangement of relations and stereotypes we get the result as in fig. 2.

The second layer of the UC-diagram will be the creation of a package diagram (fig. 3) containing these UC and realizing the capabilities of the first layer [20]. To do this, open the specification of each of the functions in the command graph and detail the commands as precedents. The type of connections used in the diagram:

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2019. No. 3 (9)

"nesting" – shows the content and additional empowerment of actors;

"compose" – composite connection;

"realize" – implementation of functions.

However, if you implement all the precedents in the diagram, you will get an overload of elements, which is incorrect, so we consider each precedent of the first level as a package.

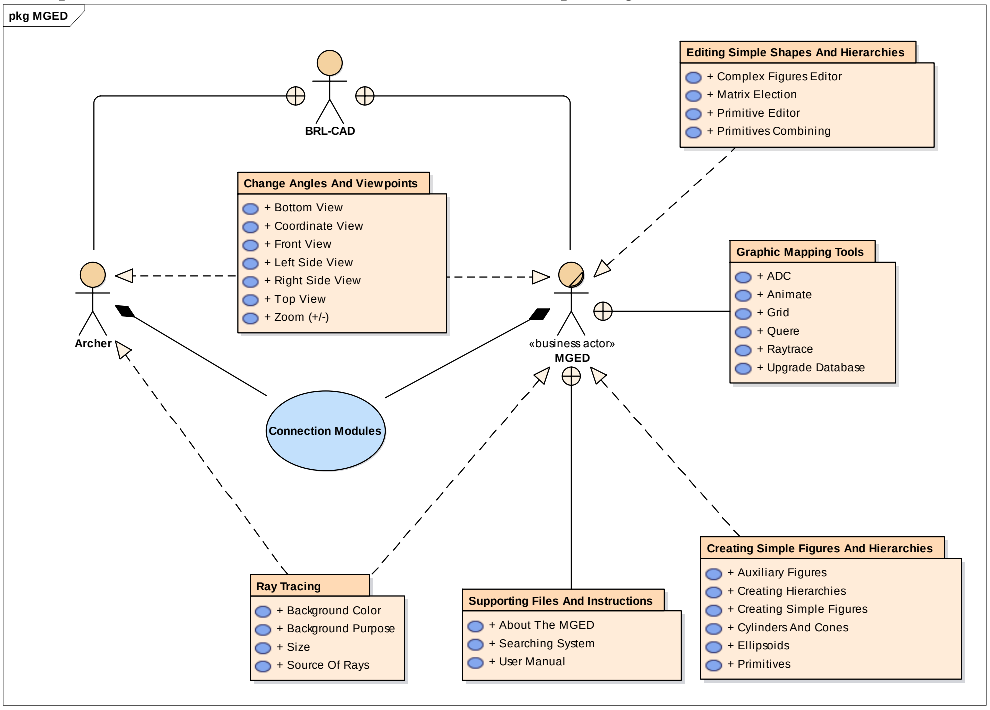

Fig. 3. Package diagram for MGED

The last step, small but not least, is to set the specifications for the actors – all the actors are software modules that are involved with the user, that is, they represent the interface. For this type of modules there is a special specification called "business", we will use it. This concludes the construction of the UC-diagram.

Generic steps for developing a new product architecture. To develop the architecture of a new product, we use three main diagrams: classes, components, and UC.

Step 1. First of all, it is necessary to determine the functionality of the future SS, how much it will differ from the original product, what changes will be made. Therefore, according to the principle of diagramming already used [21], we construct new UC that are functionally integrated into packages (fig. 4).

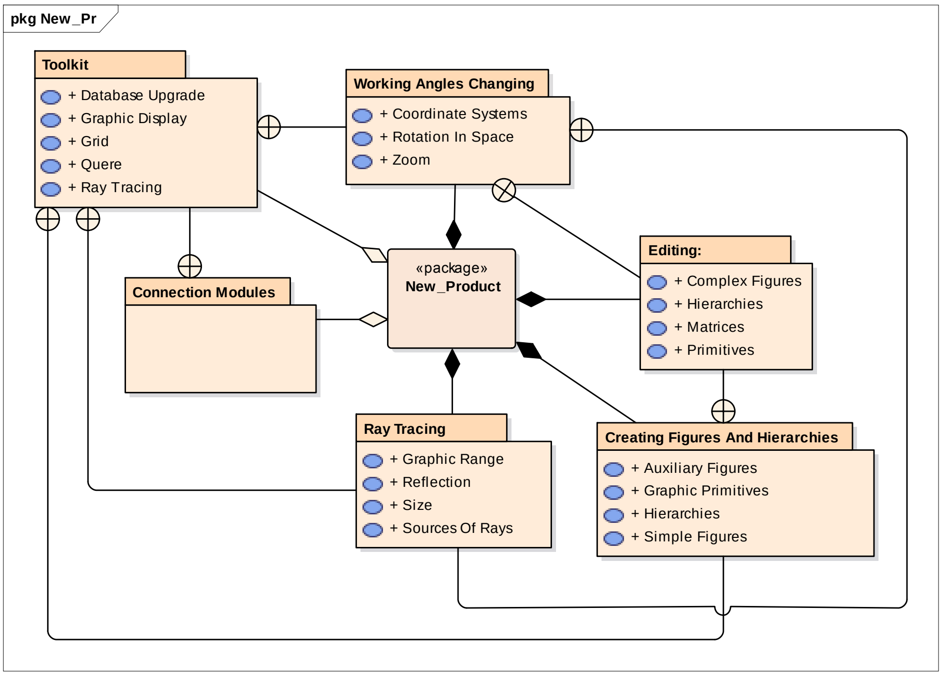

Fig. 4. Package diagram containing the UC of a new product

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2019. № 3 (9) ISSN 2524-2296 (online)

In the process of developing this diagram, it was decided that the functionality of the central module ("New_Product") should be left approximately the same as it was, and all additional complex functions should be included with the help of external modules. This allows you to not overload the SS with unnecessary commands, which will have a good effect on its performance level.

The packages responsible for performing the functions have the appropriate names: "Changing the Working Angles", "Edit", "Creating Shapes and Hierarchies", "Ray Tracing". They refer to the central module "New_Product" as composite dependencies. Each of these packages has UC directly related to the function of the specific package. The packages that are responsible for connecting the additional tools also have the appropriate tools named: "Tools", "Connection modules" and the relation to the central module "New_Product" in the form of aggregations. Also, the toolkit package includes UC, with the help of which graphical, logical, informational and other transformations and changes are performed (fig. 4).

Consider the chart using a relation of "Nesting" type, it means a hierarchical arrangement within one package function calls another. That is, if to take for a basis diagram in fig. 4, then, for example, the function "Edit" is called only after creating the shape or hierarchy (package "Creating shapes and hierarchies") or also "Edit" is also available as a component of changes after activation of the package "Changing the working angles". Also call functions of the package "Ray Tracing" is available after activation of the package "Changing the working angles", which, in turn, is visualized result (result) package "Toolkit" that is a set of software modules (package "plug-ins") which is the corresponding ratio "Nesting".

Analyzing the package diagram further, it should be added that the Toolkit uses the "Nesting" relations to include the "Ray Tracing" and "Create Shapes and Hierarchies" packages, which means that the functions of these two packages are only possible using the appropriate toolkit.

Step 2. After defining the functionality of the software, the next step is to draw up a working class diagram. The class diagram is made according to the specification of the new language and the successful solutions of the primary software product. As a result, the sketch diagram of the new software product looks like the one in fig. 5. Unfortunately, the number of classes and, accordingly, the size of the diagram is so large that the author is not able to present it in full view within the A4 sheet. If to decompose the diagram, the complete representation of the class model is lost, so we give only its sketchy representation (fig. 5). The main purpose of the sketch of the diagram: not to show the filling of classes, but to illustrate a model of hierarchical relations of boundary combinations of classes of a new software product.

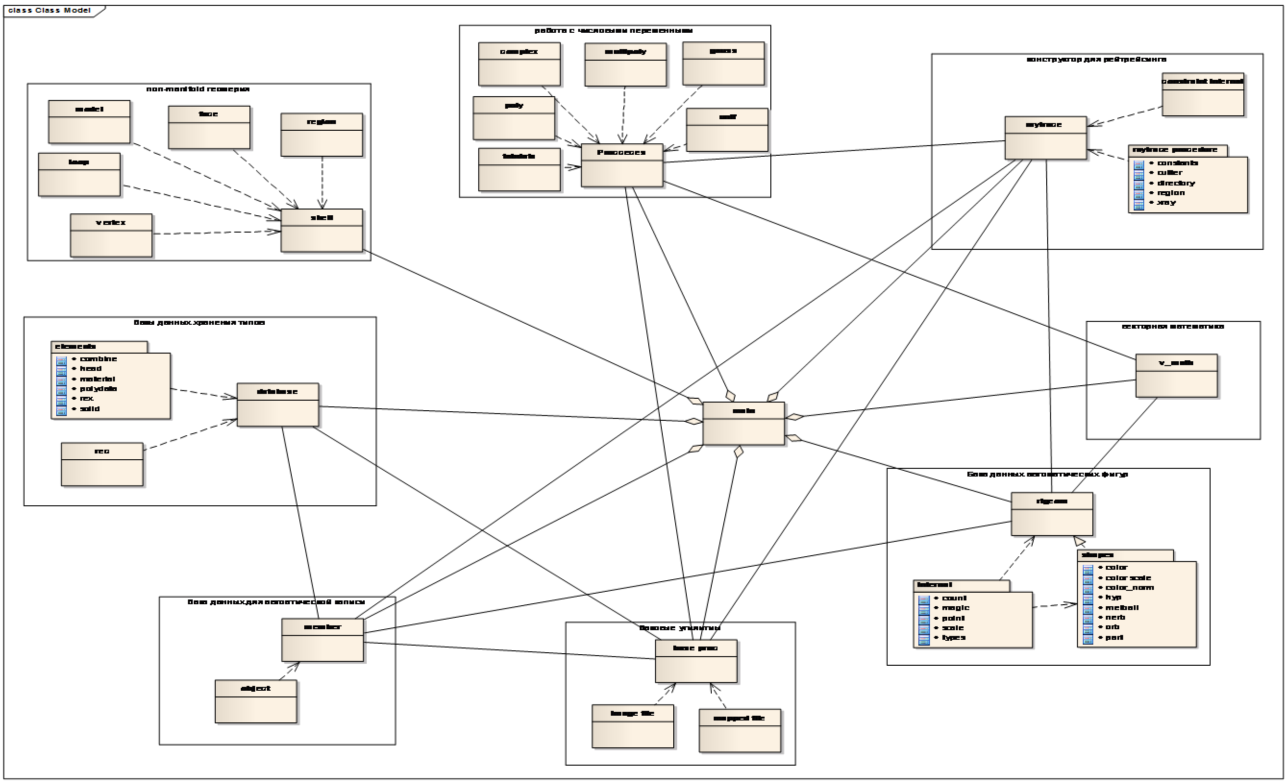

Fig. 5. A complete diagram of the hierarchy of classes of a new software product

The architectural changes made to this diagram sketch began with a full visual breakdown of the modules. Each unique module is assembled into its own separate rectangle – boundary, which will undergo the entire process of calculation. Another feature: there is a central class "Main", which is responsible for the interaction between the modules. Each module has its own boundary class, which interacts with all other parts of the SS and closed computing systems and databases. This method was chosen to minimize the number of connection errors and to help programmers improve the product.

Step 3. The class diagram is modeled; we go to the code generation process. It will be unnecessary to dwell on the details of the technical implementation of the generation process, so we will illustrate only the final result (fig. 6). After code generation, for example, take the class "main" (fig. 7). To understand how well the code is generated, let's look at the number of links to plugins. Comparing them with the class diagram, we come to the conclusion that everything is right. This development of the architecture of the new software product can be considered complete.

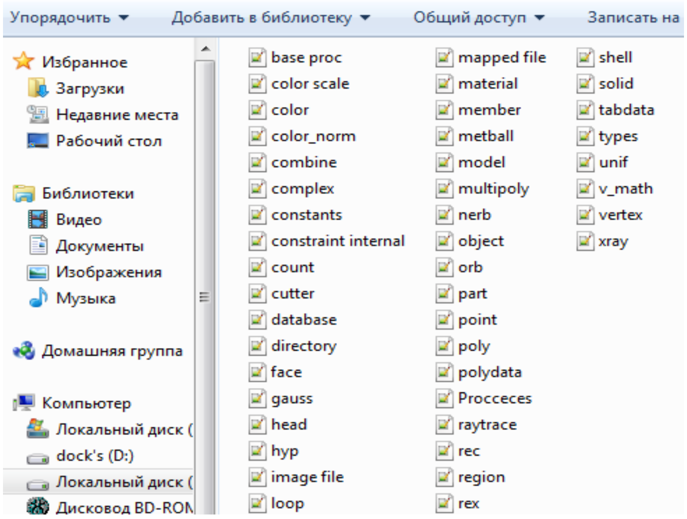

Fig. 6. Location of the code

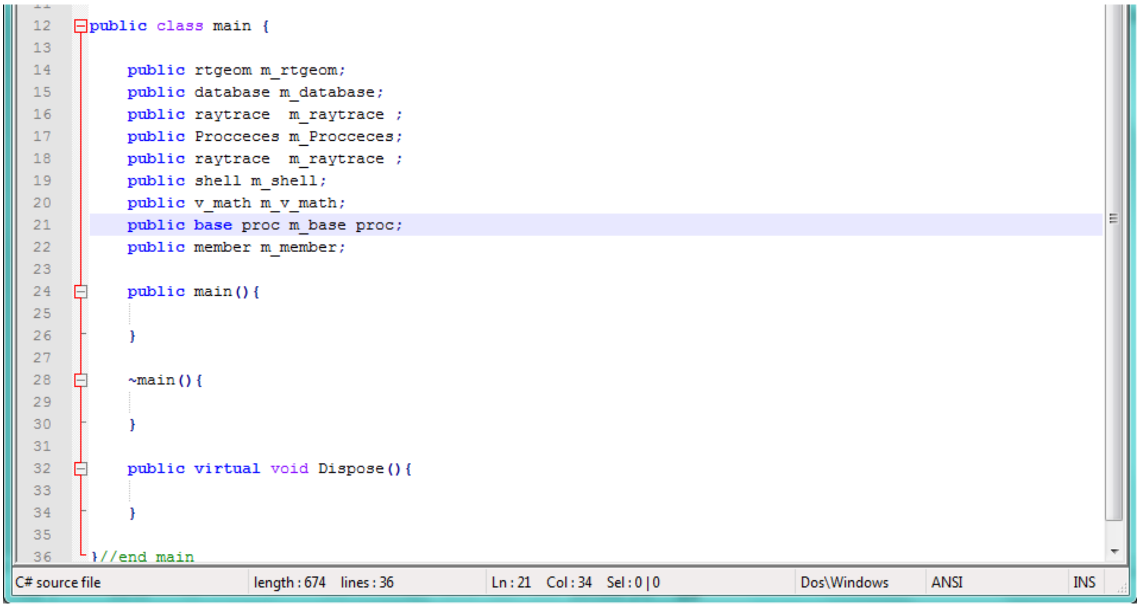

Fig. 7. Program code of the class "main"

Research results and discussion

Results. This article summarizes the research on the reengineering of open-ended CAD using the BRL-CAD example. The study was performed and its results were simulated using the UML 2.5 methodology using the Enterprise Architect 14.0 CASE tool. The UML methodology is quite voluminous and several diagrams that are used to design a new software product have been considered in the project.

The main focus is on three diagrams: classes, UC and components. This is due to the fact that directly on the basis of these diagrams, code generation and further work of programmers takes place, while other (auxiliary) diagrams are intended to explain some complex specifications of the project, which does not however diminish their significance within the project.

Summarizing the results achieved, it can be stated that at this study:

a) the results of experimental studies at the level of representation of classes, components and UC are presented, which are presented using the unified modeling language – UML, with the processing and interpretation of results at the level of CASE tools;

b) the results of source code translation were analyzed and summarized, the main of which was to

Сучасний стан наукових досліджень та технологій в промисловості. 2019. № 3 (9) ISSN 2522-9818 (print)
ISSN 2524-2296 (online)

reduce the complexity of creating an open-source software using the CAD of BRL-CAD type.

Discussion. In the most progressive countries of the world, new products have not been developed from scratch for a long time, and tools are being used to help create the structure you need much faster and more efficiently. The UML methodology and related software are for this purpose – to increase development efficiency and to structure data. This methodology has been actively used recently (close to 10 – 15 years), but very quickly integrated into the overall structure of the development. The convenience of the existing SS reengineering methodology is that it is not rigidly tied to any of the development methods and is flexible to use.

The development of a UML methodology for reverse engineering is characteristic of the West and parts of Europe. At the beginning of the research (that is, in 2009), our country's specialists have just started working on the active exploitation of this methodology as it is presented now.

The open and free BRL-CAD was a great experimental model for the job. The advantage of such systems is that they are licensed under a free license and there are no legal problems with copying, modification and other software-related actions. It should also be noted that since the code is open source, the developers tried to make it even understandable - this is facilitated by a large number of comments in the code.

Conclusions and prospects for further development

Conclusions. BRL-CAD is acceptable for the experienced designer, but for the beginner or student, its application process will be very complicated. The WAN does not have any materials in Ukrainian or Russian that at least superficially describe the operation of the system in the "user manual" mode. The English language materials are superficial and contain only a few dozen console commands.

A detailed analysis of the environment revealed the presence of two modules contained in the structure of the CAD, which help the potential user of the system to quickly construct the necessary GDB. Also fundamental to the package is the ability to support the design and analysis of visual models based on complex objects consisting of a large set of graphic primitives.

In general, when writing this article, the goal was achieved, which was to perform reengineering (evolutionary improvement while maintaining the positive qualities that are confirmed by the operation) of open-source 3D design with inheritance of full functionality and principles of operation.

After research, we can conclude: the powerful side of the system is the extreme speed of visualization tools, ray tracer and rendering. Compared to its counterparts, one can claim another advantage of BRL-CAD, namely: the visualization process is one of the fastest among the existing.

Perspectives. The last advantage is the broad prospects for the application of BRL-CAD in various fields: military, industrial or educational applications, such as design and analysis systems in the fields of engineering, mechanical units, architectural structures, molecular structures, etc.

The prospects for the study also include reengineering: the BRL-CAD utility tool group, the CAD graphics libraries, the command-and-use system, file naming conventions, and geometry. Still need improvement: simple-body creation processes, logical operations, combined-body operations, rendering and ray tracing.

The author wishes to express his gratitude to the developer company of the BRL-CAD 3D modeling software for the possibility of open use and testing of source files, assemblies and systems, as well as for support of cross-platform methodology.

References

1 Potemkin, A. V. (2002), *3D Solid Modeling* [*Trekhmernoe tverdotel'noe modelirovanie*], Komp'yuter-Press, Moscow, 296 p.

2. Velykodniy, S. S., Tymofieieva, O. S., Zaitseva-Velykodna, S. S., Nyamtsu, K. Ye. (2018), "Comparative properties analysis of open, free and commercial software" ["Porivnialnyi analiz vlastyvostei vidkrytoho, vilnoho ta komertsiinoho prohramnoho zabezpechennia"], *Information Technology And Computer Engineering*, No. 1 (41), P. 21–27.

3. Velykodniy, S. S. (2014), "Methods of software systems reengineering" ["Metody reinzhynirynhu prohramnykh system "], *Instrumentation Technologies*, Spec. No, P. 65–68.

4. Velykodniy, S. S. (2019), "Method of presenting the assessment for reengineering of software systems with the project coefficients help" ["Metod predstavlennia otsinky reinzhynirynhu prohramnykh system za dopomohoiu proektnykh koefitsiientiv"], *Innovative Technologies and Scientific Solutions for Industries*, No. 1 (7), P. 34–42. DOI: https://doi.org/10.30837/2522-9818.2019.7.034

5. Norenkov, I. P. (2000), *Computer Aided Design* [*Avtomatizirovannoe proektirovanie*], MGTU im. N. E. Baumana, Moscow, 188 p.

6. Li, Dzh., Uer, B. (2002), *Three-dimensional graphics and animation. Ed. 2nd* [*Trekhmernaya grafika i animatsiya. Izd. 2-e*], Vil'yams, Moscow, 640 p.

7. Kontsevich, V. G. (2007), *Solid Modeling Engineering Products in Autodesk Inventor* [*Tverdotel'noe modelirovanie mashinostroitel'nykh izdeliy v Autodesk Inventor*], DiaSoftYuP, DMK Press, Kiev, Moscow, 672 p.

8. Khern, D., Beyker, M. P. (2005), *Computer Graphics And OpenGL Standard. Ed. 3rd* [*Komp'yuternaya grafika i standart OpenGL. Izd. 3-e*], Vil'yams, Moscow, 1168 p.

9. Nevlyudov, I. Sh., Velykodniy, S. S., Omarov, M. A. (2010), "Use Of CAD / CAM / CAE / CAPP In The Formation Of Control Programs For CNC Machines" ["Ispol'zovanie CAD/CAM/CAE/CAPP pri formirovanii upravlyayushchikh programm dlya stankov s ChPU"], *Eastern European Journal of Enterprise Technologies*, No. 2/2 (44), P. 37–44.

10. Endzhel, E. (2001), *Interactive Computer Graphics. Introductory Course Based On OpenGL. Ed. 2nd* [*Interaktivnaya komp'yuternaya grafika. Vvodnyy kurs na baze OpenGL. Izd. 2-e*], Vil'yams, Moscow, 592 p.

ISSN 2522-9818 (print)
ISSN 2524-2296 (online) Innovative technologies and scientific solutions for industries. 2019. No. 3 (9)

11. Snuk, G. (2007), *Real-Time 3D Landscapes In C ++ And DirectX 9. Ed. 2nd* [*3D-landshafty v real'nom vremeni na C++ i DirectX 9. Izd. 2-e*], Kudits-press, Moscow, 368 p.
12. Bondy, J. A., Murty, U. S. R. (2008), *Graph Theory, Springer, San Francisco,* 655 p. DOI: https://doi.org/10.1007/978-1-84628-970-5
13. Velykodniy, S. S., Burlachenko, Zh. V., Zaitseva-Velykodna S. S. (2019), "Graphic databases reengineering in BRL-CAD open source computer-aided design environment. Modeling of the structural part" ["Reinzhynirynh hrafichnykh baz danykh u seredovyshchi vidkrytoi systemy avtomatyzovanoho proektuvannia BRL-CAD. Modeliuvannia strukturnoi chastyny"], *Transactions of Kremenchuk Mykhailo Ostrohradskyi National University*, No. 3 (116), P. 130–139. DOI: 10.30929/1995-0519.2019.3.130-139
14. Jungnickel, D. (2013), *Graphs, Networks and Algorithms. 4th Ed.*, Springer, Berlin, 677 p. DOI: https://doi.org/10.1007/978-3-642-32278-5
15. Velykodniy, S. S., Burlachenko, Zh. V., Zaitseva-Velykodna S. S. (2019), "Graphic databases reengineering in BRL-CAD open source computer-aided design environment. Modeling of the behavior part" ["Reinzhynirynh hrafichnykh baz danykh u seredovyshchi vidkrytoi systemy avtomatyzovanoho proektuvannia BRL-CAD. Modeliuvannia povedinkovoi chastyny"], *Transactions of Kremenchuk Mykhailo Ostrohradskyi National University*, No. 2 (115), P. 117–126. DOI: 10.30929/1995-0519.2019.2.117-126
16. Troelsen, E. (2011), *C # 2010 Programming Language And .Net 4 Platform* [*Yazyk programmirovaniya C# 2010 i platforma .NET 4*], Vil'yams, Moscow, 1392 p.
17. Object Management Group (2013), *Unified Modeling Language (OMG UML). Version 2.5*, OMGroup, 831 p.
18. Weilkiens, T., Oestereich, B. (2006), *UML 2 Certification Guide: Fundamental & Intermediate. Exams Morgan Kaufmann*, The MK/OMG Press, 320 p. ISBN: 0123735858
19. Babich, A. V. (2016), *Introduction to UML* [*Vvedenie v UML*], NOU INTUIT, Moscow, 209 p. ISBN: 978-5-94774-878-9
20. Samek, M. (2008), *Practical UML Statecharts in C / C++: Event-Driven Programming for Embedded Systems Newnes. 2nd Ed.*, 728 p. ISBN: 0750687061
21. Yang, H. (2005), *Advances In UML And XML-based Software Evolution*, Idea Group Publishing, 362 p.

Received 20.08.2019

Відомості про авторів / Сведения об авторах / About the Authors

Великодний Станіслав Сергійович – кандидат технічних наук, доцент, Одеський державний екологічний університет, професор кафедри автоматизованих систем моніторингу навколишнього середовища, Одеса, Україна; e-mail: velykodniy@gmail.com; ORCID: https://orcid.org/0000-0001-8590-7610.

Великодный Станислав Сергеевич – кандидат технических наук, доцент, Одесский государственный экологический университет, профессор кафедры автоматизированных систем мониторинга окружающей среды, Одесса, Украина.

Velykodniy Stanislav – PhD (Computer Science), Associate Professor, Odessa State Environmental University, Professor of the Department of Automated Environmental Monitoring Systems, Odessa, Ukraine.

РЕІНЖИНІРИНГ ВІДКРИТОЇ ПРОГРАМНОЇ СИСТЕМИ ТРИВИМІРНОГО МОДЕЛЮВАННЯ BRL-CAD

Комп’ютерна графіка – актуальна галузь проектування та застосування засобів обчислювальних систем, що інтенсивно розвиваються у останній час. **Предмет** роботи – проектування графічного інтерфейсу користувача. **Мета** роботи – виконати реінжиніринг (еволюційне удосконалення зі збереженням позитивних якостей, що підтверджені експлуатацією) відкритої програмної системи тривимірного проектування із наслідуванням повного функціоналу та принципів роботи. BRL-CAD – це спеціалізована крос-платформова система з відкритим кодом, що являє собою потужну 3D систему автоматизованого проектування для моделювання об`ємних тіл. Програмна система включає в себе інтерактивний геометричний редактор, паралельне трасування променів, рендеринг та геометричний аналіз. **Завдання**: узагальнити результати експериментальних досліджень на рівні представлення класів, компонентів та варіантів використання, які необхідно подати за допомогою уніфікованої мови моделювання – UML, із обробкою та інтерпретацією результатів на рівні CASE-засобу; виконати аналіз перекладу вихідного коду перепроектованого графічного інтерфейсу користувача BRL-CAD. **Методи**. Процес проектування чи перепроектування нових програмних продуктів є неефективним без використання UML-методології, але при її дотриманні – швидкість розробки підвищується у рази. У роботі, використовується методологія UML 2.5 із використанням CASE-інструментарію Enterprise Architect 14.0. **Результати**. Основний акцент поставлено на три діаграми: класів, варіантів використання та компонентів. На підставі цих діаграм, відбувається генерація коду та подальша робота програмістів, у той час як інші (допоміжні) діаграми призначено для пояснення деяких складних специфікацій проекту, що втім не зменшує їх значущість у рамках проекту. У поданому дослідженні виконано узагальнення результатів експериментальних досліджень; проаналізовано та узагальнено результати перекладу вихідного коду, головним з яких стало скорочення працемісткості створення відкритого програмного продукту на прикладі BRL-CAD. **Висновки**. BRL-CAD є прийнятною у застосуванні для досвідченого проектувальника, проте для початківця або студента процес її застосування виявиться дуже ускладненим. При детальному аналізі середовища було виявлено наявність двох модулів, що містяться у структурі системи, які допомагають потенційному користувачеві швидко конструювати та моделювати. Також фундаментальною властивістю пакету можна назвати здатність підтримувати конструювання та аналіз візуальних моделей на основі складних об'єктів, що складаються із великого набору графічних примітивів. Потужний бік системи – це надзвичайна швидкість засобів візуалізації, трасувальника променів та рендерингу. Після порівняння з аналогами, можна стверджувати, що процес візуалізації є одним із найшвидших серед існуючих.

Ключові слова: реінжиніринг; програмна система; комп’ютерна графіка; тривимірне моделювання; діаграма; CASE-засіб; клас.

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2019. № 3 (9) ISSN 2524-2296 (online)

РЕИНЖИНИРИНГ ОТКРЫТОЙ ПРОГРАММНОЙ СИСТЕМЫ ТРЕХМЕРНОГО МОДЕЛИРОВАНИЯ BRL-CAD

Компьютерная графика – актуальная отрасль проектирования и применения средств вычислительных систем, интенсивно развивающихся в последнее время. **Предмет** работы – проектирование графического интерфейса пользователя. **Цель** работы – выполнить реинжиниринг (эволюционное усовершенствование с сохранением положительных качеств, подтвержденными эксплуатацией) открытой программной системы трехмерного проектирования с наследованием полного функционала и принципов работы. BRL-CAD – это специализированная кроссплатформенная система с открытым кодом, представляющая собою мощную 3D систему автоматизированного проектирования для моделирования объемных тел. Программная система включает в себя интерактивный геометрический редактор, параллельную трассировки лучей, рендеринг и геометрический анализ. **Задачи**: обобщить результаты экспериментальных исследований на уровне представления классов, компонентов и вариантов использования, которые необходимо представить с помощью унифицированного языка моделирования – UML, с обработкой и интерпретацией результатов на уровне CASE-средства; выполнить анализ перевода исходного кода перепроектированного графического интерфейса пользователя BRL-CAD. **Методы**. Процесс проектирования или перепроектирования новых программных продуктов является неэффективным без использования UML-методологии, однако при ее соблюдении – скорость разработки повышается в разы. В работе используется методология UML 2.5 с использованием CASE-инструментария Enterprise Architect 14.0. **Результаты**. Основной акцент поставлен на три диаграммы: классов, вариантов использования и компонентов. На основании этих диаграмм, происходит генерация кода и дальнейшая работа программистов, в то время как другие (вспомогательные) диаграммы предназначены для объяснения некоторых сложных спецификаций проекта, однако это не уменьшает их значимость в рамках проекта. В предлагаемом исследовании выполнено обобщение результатов экспериментальных исследований; проанализированы и обобщены результаты перевода исходного кода, главным из которых стало сокращение трудоёмкости создания открытого программного продукта на примере BRL-CAD. **Выводы.** BRL-CAD является приемлемой в применении для опытного проектировщика, однако для начинающего или студента процесс её применения окажется очень сложным. При детальном анализе среды было выявлено наличие двух модулей, содержащихся в структуре системы, которые помогают потенциальному пользователю быстро конструировать и моделировать. Также фундаментальным свойством пакета можно назвать способность поддерживать конструирования и анализ визуальных моделей на основе сложных объектов, состоящих из большого набора графических примитивов. Мощная сторона системы – это чрезвычайная скорость средств визуализации, трассировщика лучей и рендеринга. После сравнения с аналогами, можно утверждать, что процесс визуализации является одним из самых быстрых среди существующих.

Ключевые слова: реинжиниринг; программная система; компьютерная графика; трехмерное моделирование; диаграмма; CASE-средство; класс.

Бібліографічні описи / Bibliographic descriptions

Великодний С. С. Реінжиніринг відкритої програмної системи тривимірного моделювання BRL-CAD. *Сучасний стан наукових досліджень та технологій в промисловості*. 2019. № 3 (9). С. 62–71. DOI: https://doi.org/10.30837/2522-9818.2019.9.062.

Velykodniy, S. (2019), "Reengineering of open software system of 3D modeling BRL-CAD", *Innovative Technologies and Scientific Solutions for Industries*, No. 3 (9), P. 62–71. DOI: https://doi.org/10.30837/2522-9818.2019.9.062.

ISSN 2522-9818 (print)
Сучасний стан наукових досліджень та технологій в промисловості. 2019. № 3 (9) ISSN 2524-2296 (online)

АЛФАВІТНИЙ ПОКАЖЧИК

Аванесова Н.Е.	5
Андрєєва Т.Є.	12
Артюх О.В.	72
Артюх Р.В.	81
Белхадер А.	22
Великодний С.С.	62
Гетьман О.О.	12
Давидовський Ю.К.	72
Євдокименко М.О.	127
Єлізєва А.В.	81
Єременко О.С.	127
Клочко О.О.	118
Колодяжна Т.В.	5
Косенко В.В.	72
Лемешко О.В.	127
Литвиненко Д.П.	91
Логінов О.В.	99
Логінова Л.В.	99
Малєєва О.В.	91
Марченко О.В.	5
Момот Т.В.	34
Панов В.В.	53
Пермяков О.А.	118
Персіянова О.Ю.	81
Пітерська В.М.	99
Рева О.А.	72
Родченко С.С.	34
Савенко К.С.	53
Терещенко Д.А.	12
Хілуха О.А.	44
Чайковський І.В.	109
Чех Н.О.	53
Шаповалов В.Ф.	118
Шпілько В.Л.	53

ALPHABETICAL INDEX

Avanesova Nina	5
Andrieieva Tetiana	12
Artiukh Olesia	72
Artiukh Roman	81
Bellakhdhar Ahmed	22
Velykodniy Stanislav	62
Hetman Olha	12
Davydovskyi Yurii	72
Yevdokymenko Maryna	127
Yelizyeva Alina	81
Yeremenko Oleksandra	127
Klochko Alexander	118
Kolodyazhna Tatyana	5
Kosenko Viktor	72
Lemeshko Oleksandr	127
Lytvynenko Dmytro	91
Lohinov Oleh	99
Lohinova Liliia	99
Malyeyeva Olga	91
Marchenko Olha	5
Momot Tetiana	34
Panov Vitalij	53
Permyakov Alexandr	118
Persiyanova Elena	81
Piterska Varvara	99
Reva Oleksandr	72
Rodchenko Svetlana	34
Savenko K.	53
Tereshchenko Dina	12
Khilukha Oksana	44
Tchaikovsky Ivan	109
Chekh Natalia	53
Shapovalov Victor	118
Shpilko Valerij	53

НАУКОВЕ ВИДАННЯ

СУЧАСНИЙ СТАН НАУКОВИХ ДОСЛІДЖЕНЬ ТА ТЕХНОЛОГІЙ В ПРОМИСЛОВОСТІ

Щоквартальний науковий журнал

№ 3 (9), 2019

Відповідальний за випуск *А. А. Коваленко*
Комп'ютерна верстка *О. Ю. Персіянова*

АДРЕСА РЕДАКЦІЇ:
Україна, 61166, м. Харків, проспект Науки, 14
Тел.: +38 (057) 704-10-51
Веб-сайт: http://itssi-journal.com
E-mail: journal.itssi@gmail.com

Формат 60×84/8. Умов. друк. арк. 15,81. Тираж 150 прим.

Відруковано з готових оригінал-макетів в типографії ФОП Андреєв К.В.
Єдиний державний реєстр юридичних осіб та фізичних осіб-підприємців.
Запис №24800170000045020 від 30.05.2003.

61166, Харків, вул. Серпова, 4, тел. 063-993-62-73
e-mail: ep.zakaz@gmail.com