UDC 371.134

Ilyasova F.S., Shkarban F.V.

THE OBJECT-ORIENTED APPROACH AS THE ARCHITECTURAL DESITION OF DESIGN SYSTEMS WHITH USING IBM RATIONAL SOFTWARE ARHITECT & MS VISUAL STUDIO 2010

У статті розглянута методологія об'єктно-орієнтованого програмування. Проаналізовані об'єктно-орієнтовані системи, а також розглянуті методи розробки систем.

Ключові слова: *програмування, об'єктно-орієнтоване програмування, об'єктно-орієнтовані системи, алгоритмічне мислення.*

В статье рассмотрена методология объектно-ориентированного программирования. Проанализированы объектно-ориентированные системы, а также рассмотрены методы разработки систем.

Ключевые слова: *программирование, объектно-ориентированное программирование, объектно-ориентированные системы, алгоритмическое мышление.*

In article the methodology of object-oriented programming is considered. Object-oriented systems are analysed and also methods of system engineering are considered.

Key words: *the programming, the object-oriented programming, object-oriented systems, algorithmic thinking.*

Statement of the problem. The current stage of development of information technology is characterized by improved methods and software development technology, based on the object-oriented approach in preparing students of computer specialists. During last years, the methodology of object-orientation-convertible programming has taken a leading position in science. Becouse the methods of teaching programming in line with recent developments in the field of modern computer science, it appears, we need to understand the detailed study of the methodology of object-oriented programming.

Analysis of the literature. Research in teaching object-oriented programming aimed at identifying approaches to teaching students computer specialties and detection conditions improve methods of training for specialists in computer technology, based on object-oriented programming.

For the first time the term object-oriented programming was given by A. Keyyem shortly G. Butch, who presented to society a clear definition of object-oriented programming (OOP), according to which the PLO – a «programming method based on the representation of programs as a set of interacting objects, each of which is an instance of a class, and classes are members of a particular inheritance hierarchy» [1, p. 69].

Exploring the current state of teaching programming at universities that train specialists in computer technology, we can conclude that the basic programming courses is still only a structural approach. The object-oriented approach is the development of structured programming techniques. It accumulates recent advances in methodology and programming languages, advances in computer technology and takes a step toward clarity and efficiency on the basis of modularity. This use of object-oriented programming in teaching higher education programming course will transfer to the new, more sophisticated level.

The article aims: to reveal and justify the methodology of object-oriented programming, in particular, to analyze object-orientation-convertible systems and their use in teaching in high school.

The main material. OOP is derived from a fixed methodology of structured programming and not something completely new. The term "programming" is closely bordered by concepts and methodology. In this study, the concept of the methodology as a set of mechanisms used in the software development process and united by one common philosophical approach. Actually "method" is defined as the process of creating a set of models that describe the various components of the developed software system.

The term "programming" is defined more than once in the pedagogical and technical literature and has no specific definition. However, most authors tend to believe that this concept is related to the development of problem solving training on information and computer technology and software, through which information and computer implemented process and exchange of information between the computer and man.

Speaking of object-oriented programming, it should be noted concepts such as function and procedure parameter passing mechanism, synchronous and asynchronous procedure. In OOP reduced global data area, and the main structure is a class.

All these factors prove beyond a doubt that the object-oriented approach is the logical continuation of the structural approach in programming and exceeds its capabilities create reusable code and simulations of

the real world. Development of methods of object-orientation-convertible programming significantly influenced the way of improvement of modern programming languages.

Since programming techniques based on manipulation of objects in a global sense, arising from the philosophical and psychological theories, this method could not leave indifferent many researchers theoretical and philosophical aspects of programming.

Object-oriented programming incorporates the achievement of other programming methods, thereby allowing under one methodology to solve problems and build models of different classes. It could be argued that object-oriented programming has absorbed a recent achievements of other programming methodologies in recent years. In fact, object programming is a methodology of system analysis and design, but the ideas that contains simulations are necessary for the practical use of the language data representation and programming tools.

Thus, object-oriented programming solves the problem of exploring different approaches to programming in general, appeared due to the emergence of different types of tasks. Adding these methods in the educational process management, as it allows to increase the level of training of students in the software industry and information technology, as well as to bring together the content and methodology of science, science content and methods of teaching high school educational courses related to computer science.

For the introduction of object-oriented programming in the learning process of modern high education requires a certain quality software support, including language should be used programming environment that would satisfy the requirements for object-oriented systems for teaching programming.

Object-oriented systems can be classified according to several criterias. In particular, ER Tello [2] proposes to select features that determine the degree of object oriented programming system:

1) set of classes and instances;
2) encapsulation of functions and data;
3) binding of the implementation period;
4) multiple inheritance;
5) messaging.

Several other researchers consider the system of object-oriented if it supports four object concepts:

- abstraction;
- encapsulation;
- ınheritance;
- polymorphism.

Modern scholars believe that object-oriented systems are divided into object and hybrid systems. In the Object System is an object: classes are objects, instances of a class. Just as objects are created from classes, which is their model or a model, and most classes are objects that are created in accordance with the pattern entered into a certain class. This class is commonly referred metaklassom. The hybrid system is a system in which objects exist with the normal elements of programming languages.

Thus, an object-oriented language is a high-level language for object-oriented programming systems. In such a system, the concept of subroutines and data that are used in conventional programming systems replaced the concepts of «subject» and «message».

Initially, the study of object-oriented languages, you can use a graphical representation of their basic characteristics. This approach is used to provide insight into the initial programming concepts: basic data types and control structures and OOP modular, recursive algorithms and data structures.

Characteristics of object-oriented software:

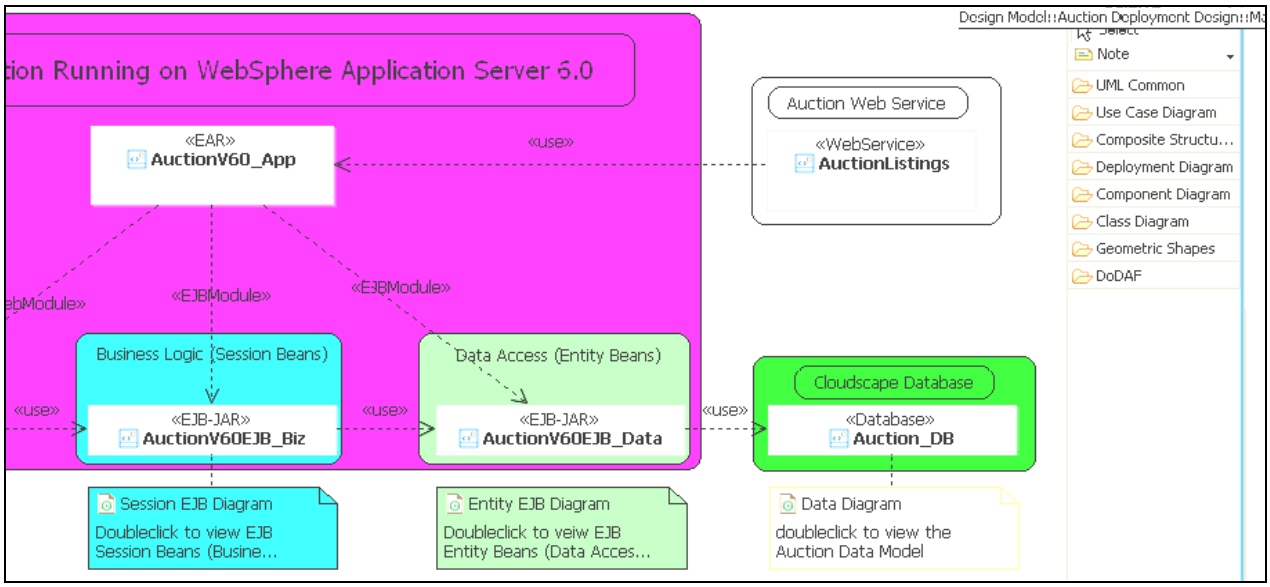
- platform independence;
- effectiveness;
- class design;
- hierarchy of classes;
- exclusion;
- methods for high-level design of the system [3].

Consider a more high-level methods of system development. Language modeling is necessary for successful design – this is one of the most important phases of the object-oriented approach. Unified Modeling Language (UML) allows a higher level of design automation.

Designing software using software tools Rational Software Architect and Microsoft Visual Studio 2010 is required to create architectural solutions using UML-diagrams at early stages of development [4, p. 115].

The software package IBM Rational Software Architect is an integrated environment design and development of software applications and services using models based on UML 2.0 [5]. Flexible management model allows to perform parallel development and change in architecture – divided into parts,

combine, compare and merge models and model fragments (pic. 1) [6].



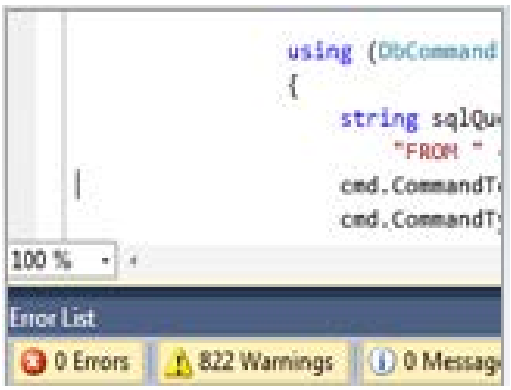
Pic. 1. Example main chart.

Design begins with a main diagram (main diagram), which includes components such as Web-diagram (Web Diagram), Session EJB Diagram (Logical Model), Entity EJB Diagram (component model), Data Diagram (database model) and uses database «Cloudscape».

MS Visual Studio 2010 – a powerful IDE that ensures quality code throughout the software development cycle, from design to development. Whatever program you develop for SharePoint, Web, Windows, WindowsPhone, and other platforms, Visual Studio is a universal comprehensive solutions [7].

New tools of Visual Studio 2010 allows you to visualize requirements and work out appropriate architectural solutions using UML-diagrams at early stages of development. During the development process, tools, code visualization and analysis of relationships between different components of helping developers to significantly improve the quality of generated applications. The combination of DSL and UML tools with code generation capabilities based on templates and extensions development environment using the appropriate SDK expands portfolio of tools developers and improves the performance of their work [5, 8, c. 196-206].

Code analysis tools in Visual Studio 2010 Premium avoid common programming errors before you run applications in production. Code metrics are used to measure the complexity of the code, ensuring its simplification and ease of maintenance (pic. 2).



Pic. 2. Revealing mistakes.

Conclusions. Using object-orientation-quests of understanding the structure of the program much easier to help reduce the complexity of the problem and speed up the development process. Using a universal tool set simplifies development programs intended for independent developers and working groups. These systems allow you to create scalable and high-quality programs. When writing code, creating databases, testing and debugging can increase productivity through the use of effective tools, convenient it is for the developer and working groups

References

1. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений / Г. Буч, Р. Максимчук, М. Энгл ; [пер. с англ.]. – [3-е изд.]. – М. : ООО «И.Д. Вильямс», 2008. – 720 с.
2. Телло Э.Р. Объектно-ориентированное программирование в среде Windows / Э.Р. Телло ; [пер. с англ. Д.М. Арапова, А.К. Петренко]. – М. : Высшая школа : Наука-Уайли ; СПб. : Акме, 1993. – 347 с.
3. Онлайн семинары по современным технологиям [Electronic resource]. – URL : http://www.techdays.ru/videos/2535.html.
4. Брауде Э. Технология разработки программного обеспечения / Э. Брауде. – СПб. : Питер, 2004. – 655 с.
5. Website of IBM. Руководства по Rational [Electronic resource]. – URL : http://www.ibm.com/developerworks/ru/.
6. Website of Microsoft Visual Studio 2010 [Electronic resource]. – URL : http://microsoft.com/visualstudio.
7. Website of IBM [Electronic resource]. – URL : http://www.ibm.com.
8. Brooks F.P. The mythical man-month: essays on software engineering / Frederick P. Brooks. – Addison-Wesley Professional, 1995. – 336 p.