

THE REENGINEERING OF THE COMPUTERISED ACTIVITIES INTO E-BUSINESS ACTIVITIES

Tadas Žobakas, Saulius Gudas, Diana Žobakienė

*Information system department, Kaunas University of Technology
Studentų St. 50-309, LT-51368 Kaunas, Lithuania*

Abstract. The paper presents a case study of joint implementation of the reengineering method of the computerised activities and the method of user interface engineering. The case study represents the general principles of reengineering process of the computerised activity into the e-business activity with the user interface. The analysis and design phases of the reengineering method are discussed in detail. At the end of the paper, the examples and general principles of developing of the task-dialog model for e-business “E-order” user interface is presented.

1. Introduction

E-business systems should be treated as a new kind of information system. In development and implementation of this kind of information systems not only new technologies but also new methodologies are used. The Web service is a new internet technology used in the implementation of an e-business IS. Adding user interface to the Web services allows the implementation of Web services based e-business activities not only in peer-to-peer but also in interactive environments.

However, integration of new e-business with old but still working information system is already now one of the main problems confronting modern organisations. In general, the new system should add new functionality to the old system without changing it completely. The objects of an e-business system are computerized activities of an old information system, i.e. formerly computerized activities need be adapted to the new internet-based technological environment in which case the methods of reengineering (adaptation) of existing computerized activities should be used.

Known hybrid IS reengineering methods [4], [5] are not suited to use specific, i.e. e-business, knowledge because they have been developed to support the reengineering of a general (not a specialized) information system. The majority of traditional and hybrid reengineering methods are empirical and specify general reengineering principles only. The IS designers who use such methods and CASE systems which support the methods of developing e-business systems are limited to using e-business patterns that help to develop typical situations easier and faster. For this reason, integration problems arise between typical e-business

activities and the computerized activities of the old IS that should be reengineered.

It is very important to ensure the correctness of the process model in e-business IS development, because not every redesigned computerized activity of the old IS can be implemented using selected internet technology such as Web services. The most popular and frequently used CASE systems which implement reengineering methods do not assure or only partly assure the correctness of the process model’s suitability for practical implementation. For this reason, incorrect process models are the direct cause of longer duration of e-business IS implementation and therefore the development of the e-business IS does not run smoothly.

The purpose of this paper is to present the method of reengineering the computerised activities [8] by an example. The example describes the process of the reengineering of the computerised activity “Order entry” of the old IS into the e-business activity “E-order” and presents the general principles of the engineering of the user interface for the “E-order” activity using UI engineering method [10].

This paper has the following structure. An overview of the general principles of the reengineering method (except generation and implementation stages) is provided in Section 2. The engineering process of an e-business activity with user interface is described in Section 3. The conclusions are presented in Section 4.

2. The general principles of the reengineering method

The method of reengineering of the computerised activities is designed for transforming the computerised activities of the old IS which is usually treated as value chain into the activities of the e-business IS. One of the main features of the method is the algorithm of the correctness assurance of the process model [9]. The algorithm uses the meta-model, which is based on EMC model [1] and specifies the functional and technical characteristics of the e-business problem area. These specifications are called process specifications and ensure that e-business process model is suitable for implementation using Web services architecture.

The quality assurance of the process model is apart of the IS life cycle used in reengineering method. This life cycle is an improved traditional life cycle of the IS [2]. The improved life cycle is specified formally and is oriented at engineering a Web service based e-business IS reengineering the computerised activities of the old IS.

A traditional IS life cycle has 3 major phases, i.e. analysis, design and implementation. The reengineering method of computerised activities distinguishes the last phase as two phases, i.e. generation and deployment (Figure 1). That allows to expose the peculiarities of the deployment phase which are important in deployment of Web services based e-business systems in order to create a secure IS. The e-business IS life cycle used in reengineering method of computerised activities is an iterative IS life cycle with additional correctness control of process model at the design phase. The correctness control is also iterative. It is considered that IS developer can return to previous phases of the IS life cycle, i.e. from design to analysis or from deployment to analysis or design.

The new user requirements are identified and specified by problem area experts and/or IS analysts (further – IS developers) during *analysis stage*. All identified requirements should be directly associated to supplement the computer activities of the old IS with e-business functionality. The typical e-business activities and the UML models of computerized activities of the old IS are the data that IS developers have at the beginning with analysis. According to the reengineering method of computerised activities, the task of the experts is to define how the new requirements should be implemented (the requirements must be specified using other methods). The result of the analysis is a list of e-business activities of organisation created by the IS analysts according to the new user requirements. The list includes not only the e-business activities but also the actions and information sources needed to develop each of these activities. The list is incoming data for the next stage of the new reengineering method.

The process models of the e-business activities of organisation are created at *design stage*. This is

performed using the process models of the typical e-business activities and the computerised activities of the old IS. The process models are specified in adopted UML graphical notation [7]. The UML graphical notation has been chosen due to its popularity among IS developers and flexibility specifying e-business processes. Usually, the process models of the computerised activities of the old IS are not physically divided and have one class and one deployment diagram and some sequence and activity diagrams for each computerised activity. The meta-model of the adopted UML graphical notation of one activity consists of one class, one sequence and one activity diagram.

The design phase is divided into two parts: development of process model and development of IS architecture which is also known as IS model. The development of the e-business process model also includes the assurance of the syntactic and partly semantic correctness of the model [8], [9]. The correct process model is supplemented with user interface (if needed) and data model developing the architecture of e-business IS.

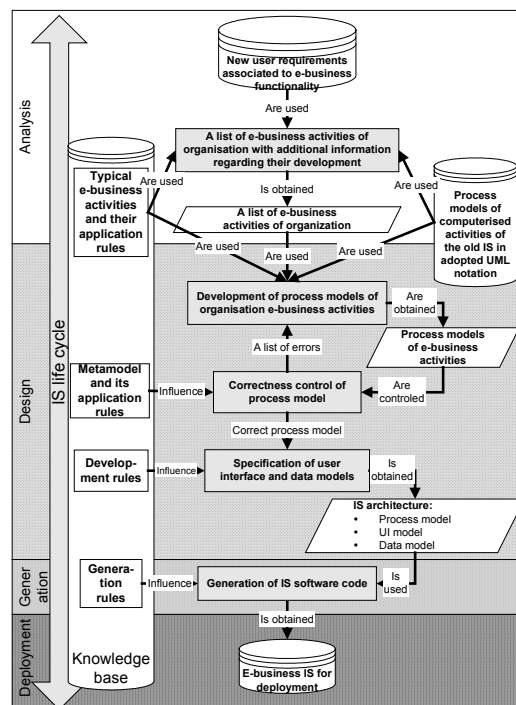


Figure 1. The general steps of the re-engineering method

The knowledge base of the reengineering method consists of 4 parts. The typical e-business activities are stored as process models in adopted UML graphical notation. The application rules specify their integration with the computerised activities of the old IS. The meta-model is based on EMC method and its application rules specify its application algorithm in the correctness assurance of the e-business process model. The development rules of the user interface (UI) and data models specify the development process of IS architecture. The generation rules specify the

algorithm of the generation of the software code of an e-business IS, based on Web services architecture.

IS architecture is the primary data in the *generation stage* for the application code generation. The generation algorithm uses process, data and user interface (if needed) models to generate application code and other information which is necessary for the information system to function, e.g. graphical user interface, database scripts, Web service contract. The application code is logically divided into definite functions which correspond to the architecture of the Web service. The „pieces of the code” should be deployed in at least two servers in order to create a secure IS [3]. This deployment architecture enables the separation of user interface, business logic and data.

In the *deployment stage* IS developer or IS implementer should install the e-business application code and other generated files at appropriate servers.

3. The engineering process of an e-business activity

The experimental CASE system based on the re-engineering method of the computerised activities has been tested in practice developing the e-business project of a trading organization. The realization of the demand of e-business was identified at the beginning of the project as the main purpose of the project. The further requirements to the e-business were stated as follows:

- the new e-business activities should be based on the computerised activities of the existing and functioning IS;
- the new e-business and the old computerised activities should function parallel at the same time;
- the new e-business activities should be integrated into the old IS in terms of data.

3.1. The reengineering process of the computerised activity “Order entry”

At the analysis stage, the new user requirements related to the e-business functionality were specified working together with the executives of the organisation. One of the requirements was specified as “get orders through the internet”. This requirement expresses a demand of the organisation to get the orders from the registered internet customers electronically through the internet. The requirement was specified in the following way:

- These orders are e-orders and are different from other orders because they should have an attribute identifying their internet genesis.
- Item balance and delivery terms should be calculated using an additional algorithm provided by the employees of the organisation.

- The e-order can enter the organisation’s IS through the user interface or as an information package.

The main basis of the new e-business activities are the computerised activities of the old IS which usually have one class diagram (Figure 2). The re-engineering method treats each computerised activity as a process model having one class diagram, one sequence diagram and one activity diagram. So the global class diagram was divided into three logical pieces representing three computerised activities, i.e. order entry, items’ catalogue representation and item balance.

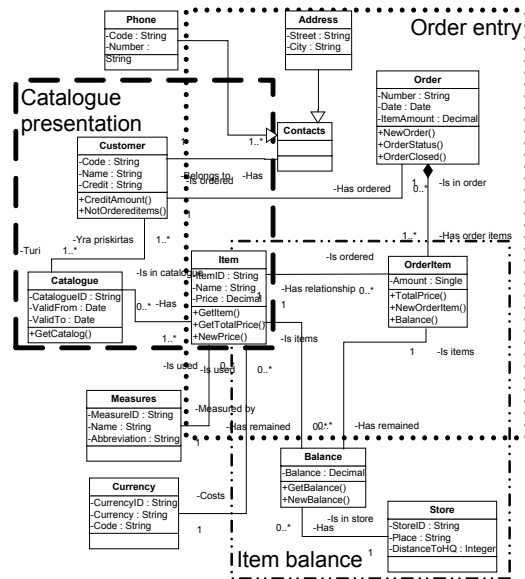


Figure 2. The class diagram of the computerised activities of the old IS

The computerised activity “Order entry” was treated as a basis for the new e-business activity “E-order” in design phase and was specified in the following way:

- The activity is responsible for receiving the customer order from the employee using GUI of the old IS and forwarding it.
- An order could be received from the registered customers. In a case of a new customer it must be registered (the registration process is not realised by this activity). An order consists of the customer code, order number, date and the list of ordered items with amounts.
- An order can be processed when the credit limit of the customer is not exceeded or the order is already paid (the payment is not the part of this activity).

The computerised activity “Order entry” is directly related to five classes presented in Figure 2. The order can be received from the registered customers which data (code, name and credit limit) are stored in the class “Customer”. The class “Order” stores the general order information, i.e. number, date and the number of ordered items.

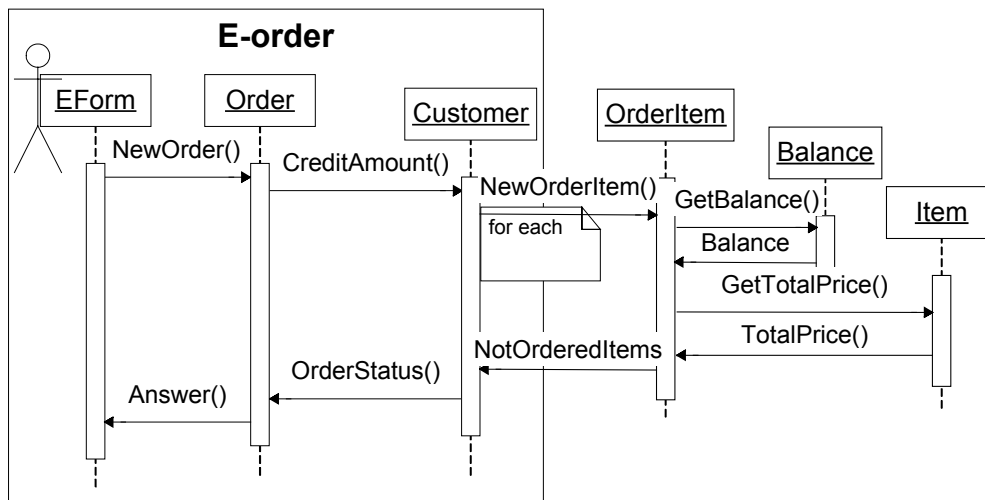


Figure 3. The sequence diagram of the computerised activity „Order entry“

Each order must have at least one order item (class “Order item”) with the information of ordered amount. Each order item is linked to the class “Item” which stores item related information, i.e. unique item number, name and price. The item can be ordered until the balance of the item in the store is greater than or equal to the ordered amount. According to the requirements, the classes of the computerised activity “Order entry” must be extended by one attribute, i.e. the class “Order” should have one additional attribute “Internet order” which can be true or false. According to the specification of the reengineering method, the typical e-business activity “E-order” can be used to speed up the development process of the e-business IS.

The sequence diagram of the “Order entry” is presented in Figure 3. In the specification of the reengineering method of computerised activities [8], [9] it is said that a sequence diagram must have 3 objects where the third object and the rest objects following this object are specified in the activity diagram. So the first three objects in Figure 3 are marked as a basis for the new e-business activity “E-order”.

Figure 3 shows that the data of the new order are obtained from the screen form electronically. The data are forwarded to the function “New order”. The next function returns an amount of the credit available to the customer. The function “New order item” processes the information for each order item, proving item’s balance (function “Get balance”) and obtaining item’s price (function “Get total price”). The rest functions (“Balance”, “Total price”, “Not ordered items”, “Order status” and “Answer”) process the returning information.

The design of activity diagram was complicated because the computerised activity had one activity diagram for each function specified in the sequence diagram. IS developers have united all activity diagrams in one activity diagram presented in Figure 4.

Figure 4 depicts the internal logic of order processing. The function “Credit amount” in the sequence diagram is represented as a database function and used

to get the credit of the customer. The other 4 functions in the sequence diagram “Get balance”, “Balance”, “Get total price” and “Total price” are specified as a set of operations and logical proofs in the activity diagram implementing user requirements which were defined in the analysis stage.

Summarising the both stages of the reengineering method, the following steps were done developing e-business activity “E-order”. In the analysis stage, it was defined that:

- The computerised activity “Order entry” is the basis for the new e-business activity “E-order”. The typical e-business activity “E-order” is treated as a subsidiary activity because the main business logic is already implemented in the computerised activity.
- The class “Order” must be extended by one attribute from the typical e-business activity defining the type of the order (internet order or common order).
- The delivery term should be calculated according to the specified algorithm.

There were made the following steps in the design stage while developing the e-business process model:

- The class diagram was separated into 4 logically separated diagrams.
- The “E-order” class diagram was corrected by adding the new attribute “Internet order” to the class “Order”.
- The sequence diagram was cut by taking only the first three objects.
- The activity diagram was joined using separate activity diagrams used in the computerised activity “Order entry”.
- The delivery algorithm was developed for the function “Order status” by changing the existing algorithm (not presented there due to commercial reasons).

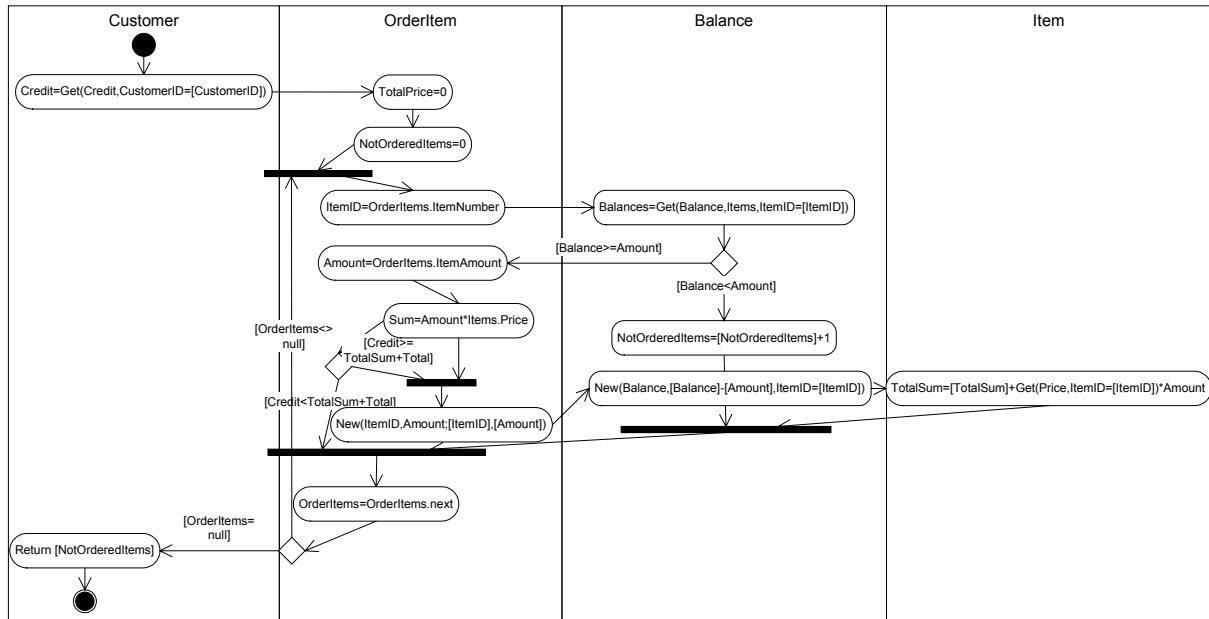


Figure 4. The joint activity diagram

The second part of the design stage is the development of data and user interface models. The data model is developed in two steps. At first, the data sources, their data tables and data fields are identified. Then each attribute of the class diagram is assigned to the appropriate field of the table in the data source. There is only one limitation – the type and size of attribute and assigned field must match.

The user interface is developed specifying HTML elements for each incoming and outgoing attribute. In this case we'll use user interface modelling method in order to specify the UI development process.

3.2. The engineering of the user interface

There are some models used in the engineering of the user interface. The models assist IS developers in better understanding of separation of processes and user interface. The model-based methods use four models [6]. The process model describes the process which the user interface should serve. Task-dialog model specifies the tasks which the user can perform using the IS and their interconnections. The abstract presentation model specifies the structure and behaviour of the visual user interface using abstract objects. The concrete presentation model specifies the parts of the user interface in detail.

In our case, the UI process model is developed from the knowledge already specified in the process model using reengineering method. The UM graphical notation is used to specify the UI process model [10]. The class diagram of e-business activity “E-order” got at the design stage of the reengineering method is used in UI modelling method. Meanwhile, the sequence diagram should be specified once more, because it should describe only the process, which is related to the user interface (Figure 5).

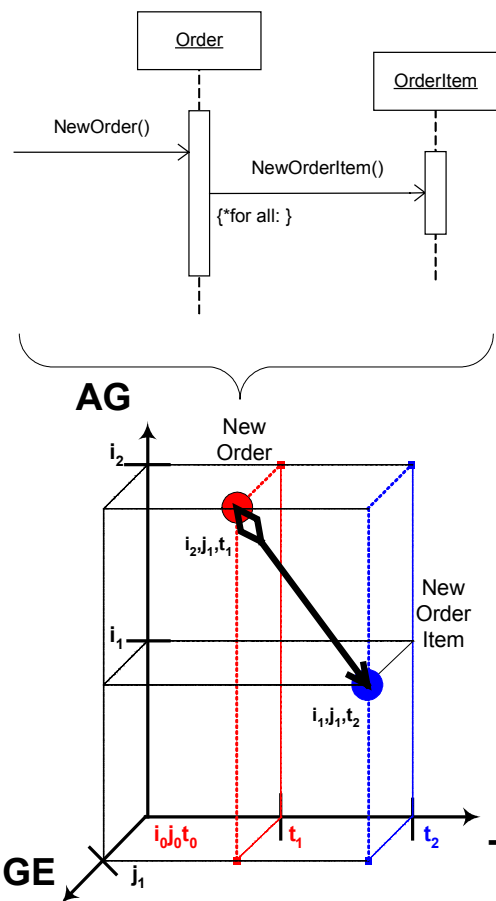


Figure 5. Transformation of the UI process into the task-dialog mode

The task-dialog model is developed in the space of processes [1]. Its main element – formally defined elementary management cycle (EMC) – is used to specify the user interfaces. The hierarchical (aggregation

and generalisation) and non-hierarchical relationships between two EMCs represent the transition from one user interface to another.

In the case of e-business activity “e-order” the data of each order are gathered using two screen forms. The first screen form is used to get order date. The second screen form is used to gather information about each order item. The association between the second and the first form is a generalisation which becomes the cardinality from the class diagram, i.e. “1..*”. The cardinality describes the rules of association of the two forms, i.e. it must be at least one valid order item entered in order to process order data. This relationship also shows that the second screen form can be used one and more times.

4. Conclusions

The key steps of re-engineering including two methods are presented, namely the method of re-engineering of computerised activities and the method of user interface engineering. The examples illustrate the basic principles of re-engineering of computerised activity “Order entry” into e-business activity “E-order” and engineering of its user interface.

Both methods are presented using the same process model notation (UML) and are formally based on the EMC model. The same basis notation allows to develop the user interface of e-business IS re-engineering the computerised activities of the old IS faster and more smoothly than using logically not associated e-business and user interface development methods.

References

- [1] **S. Gudas.** A framework for research of information processing hierarchy in enterprise. *Mathematics and Computers in Simulation*. ISSN 03784754, Elsevier, No.33, 1991, 281-285.
- [2] **S. Gudas, T. Zobakas.** The models of the life cycles’ of the processes of organisation activity. *Information sciences*, ISSN 1392-0561. Vilnius: VU, No.10, 1999, 73-82 (in Lithuanian).
- [3] **J. Kao.** Developer’s Guide to Building XML-based Web Services with the Java 2 Platform, Enterprise Edition (J2EE). *The Middleware Company*, 2002 (internet source <http://www.theserverside.com/resources/pdf/J2EE-WebServices-DevGuide.pdf>).
- [4] **R. Kazman, S.G. Woods, S.J. Carrière.** Requirements for Integrating Software Architecture and Re-engineering Models. *CORUM II*.
- [5] **L.H. Rosenberg, L.E. Hyatt.** Hybrid Re-engineering. *UnisysTM Federal Systems*, 1997 (http://satc.gsfc.nasa.gov/support/ISRE_JAN97/rengart6.pdf).
- [6] **P.P. Silva.** User Interface Declarative Models and Development Environments. *A Survey. In Proceedings of DSV-IS 2000*. ISBN 3-540-41663-3, Telos, Springer, 2001, 204-226.
- [7] Unified modelling language (UML). *Version 2.0*, 2003. (<http://www.omg.org/technology/documents/formal/uml.htm>).
- [8] **T. Zobakas, S. Gudas.** The approach for object-oriented model verification Information sciences. ISSN 1392-0561, Vilnius, VU, 2002, No.23, 103-112 (in Lithuanian).
- [9] **T. Zobakas, D. Zobakiene, S. Gudas.** A role of meta-model in a quality assurance of process model developing Web service based IS. *Information technologies for business-2004*. ISBN 9955-09-649-7, Kaunas, Technologija, 2004, 250-255.
- [10] **T. Zobakas, D. Zobakiene, S. Gudas.** Composition of task-dialog model in the space of processes. *Information technology and control*. ISSN 1392-124X. Kaunas: Technologija, 2004, No.4(33), 67-75.

Received February 2006.