ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2006, Vol.35, No.3A

THE USE OF BUSINESS RULES FOR THE SPECIFICATION OF DYNAMIC ASPECTS OF IS

Kęstutis Kapočius, Tomas Danikauskas

Kaunas University of Technology, Department of information system
Studentų st. 50-308, LT-5136, Kaunas, Lithuania

Abstract. In the paper Output driven Information systems (IS) requirements specification method (ODRES) is overviewed emphasizing the problems of recording certain requirements. It is suggested that the method can be extended with business rules recording capabilities. Business rules-based requirements specification method is used to extend ODRES by incorporating method's rules submodel into an ODRES metamodel in order to support non-structural business rules as additional knowledge about static and dynamic aspects of IS. Presented extension of ODRES concerns only the specification of dynamic aspects of IS. BRS RuleSpeak, which is the basis for a suggested rules submodel, is also overviewed in the paper and the benefits of the extension are outlined.

1. Introduction

Currently one of the key problems of Information systems (IS) requirements specification is a wide gap between analysts and stakeholders [12]. Usually this means that requirements provided by the analyst to a designer significantly contravene with the actual needs of the future system users. On the other hand, informally specified requirements, although acceptable for the stakeholder, are virtually unusable for the designer, who is likely to find them too ambiguous.

This paper discusses the merger of elements from two different requirements specification methods, thus trying to find a compromise between the needs of stakeholders, analysts and designers. The first one is the Output Driven Requirements Specification Method (ODRES), which is under development in Information Systems Department of Kaunas University of Technology. According to ODRES, IS requirements are specified analysing the input and output data flows. All collected requirements are stored in a specification repository, which was designed for this method. Analysis of ODRES features revealed that in the repository a significant number of non-functional requirements remain unrecorded while some of the recorded ones are too informal.

The solution to these problems can be found in business rules (BR) approach. This brings us to the second method – BR-based requirements specification method, which is also being developed in Kaunas university of Technology. BRS RuleSpeak-based BR submodel of this method's requirements metamodel was applied for the extension of ODRES. RuleSpeak [16] is a BR structuring model which offers natural language templates for action assertion and derivation type rules (according to GUIDE typology [9]). This concept fully met our needs, because in ODRES both static and dynamic aspects of IS can be recorded in a formalised manner as action assertions and derivations.

This paper presents ODRES extension with business rules for the specification of dynamic aspects of IS. Most important features of ODRES as well as its shortcomings are overviewed in chapter 2. The motives behind the choice of RuleSpeak driven BR model are outlined in Section 3 and the short presentation of the extension is given in Section 4. Note that the objective of the presented research was to create the means for storing business rules as additional constraints of ODRES, while the process of rule specification has not been elaborated at this stage.

2. Output Driven Requirements Specification Method

Output driven requirements specification method is based on the analysis of data flows in the organization. Every IS can be viewed as a system of separate input and output information flows. By operating these flows, information flows specification can be created. Output information requirements should be considered as key requirements for the computerised IS [2]. Therefore the analysis of IS requirements starts with the discovery of output information flows. Knowing the structure of these flows analyst can go on and identify necessary input flows (data resources) and their composition. From this point of view, IS can be considered as a system, the aim of which is to ensure the efficiency of organization's information flows management.

Basically, specification of information flows is a system consisting of the following six models:

1. Information system context model;
2. Results/data resources structure model;
3. Model of information flows between O, I and structure of those links;
4. Results/data resources processing stages model;
5. Results/ data resources state transition model;
6. Model of elaboration of links between O, I and links between O, I states.

Models 1-3 (see Figure 1) are used for the specification of static aspects of IS, while the dynamic ones are specified in models 4-6.

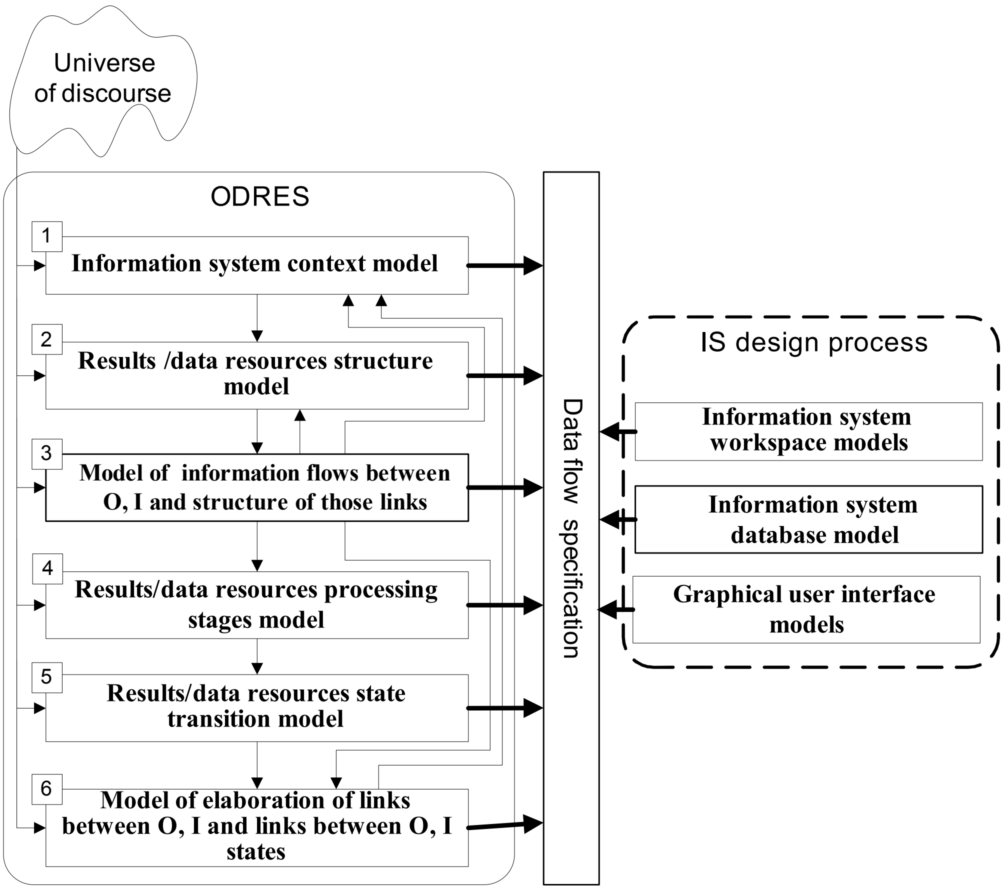

Figure 1. Conceptual schema of ODRES method

Under ODRES requirements specification is an iterative process. Transitions to every stage are sequential but from each stage the analyst can return to the previous one in order to correct or complement the known information. Specification process starts with the definition of the context of the activity to be computerised. To achieve this goal, a modified Functions Hierarchy model has to be developed [1]. Lowest level functions are related with data resources and results. Data structure is specified for each result. During the specification, for each result, structure model and results entity relationships (ER) model have to be developed [3; 5]. Because the data resource structure corresponds to the structure of computerised IS functionality result (they are composed of the same type elements) [4] the same specification process is carried out for the data resources.

After the aforementioned steps, the specification of data flows and their structure has to be performed [17]. Data flows can be viewed as informational relationships between data resources or between data resources and results. One of the major rules of data flows specification is the requirement for at least one data resource to exist for each data flow.

The fourth stage of the method is concerned with the specification of data resource processing sequence specification. The result is the specification of data resource processing stages and transitions between these stages. This stage of the method is directly related to IS design process. It often happens that data resource is being filled with information gradually by defining the values of certain attribute sets. Furthermore, data resources can be processed in parallel and several actors can be processing each resource. Note that after the design of computerised IS all these features will have a direct or indirect impact on the final product and especially its user interface elements.

The fifth stage is basically an elaboration of the previous stage. The processing of data resource is elaborated to the level of data resource's states. During the states specification all possible ways of data resource processing can be defined. Due to this feature, a system, which fully supports all relevant processes of the organisation, can be developed. Note that in stages four and five a communicational actions and transitions between objects graphical notation and their modelling principles proposed by R. Gustas [10] have been applied.

The last stage of the presented process involves the transfer of relationships, which were identified and described during the third stage, to the data resources state model on the level of entities, which were specified during the second stage. Also, during this stage, previously defined relationships between data resources and results are elaborated.

After all of the aforementioned models have been created, the system analyst/designer has an information flows specification, on the basis of which computerised IS can be designed. Results of the specification can be used to automate the IS design process. ODRES method-based CASE tool prototype is currently under development. The tool consists of independent modules for specification and design tasks as well as a repository for all results of the specification [6].

In ODRES specification repository, in many cases constraints are stored as non-formal natural language statements. This makes the use of these constraints during the IS design quite problematic, therefore this issue had to be addressed in one way or the other.

The aim of the presented work was to extend the specification of dynamic elements by business rules. It is important to note that BR approach was already applied for the specification of data resources structural elements [8]. The key decisions that were made during the extension are overviewed in Section 3 of this paper. Examples of business rules used in the description of the extension (Section 4) are from the Lithuanian State Forest Survey Service Information System requirements, which were specified with participation of Information Systems Department of Kaunas University of Technology using ODRES method.

3. Selection of a BR structuring model

In order to extend ODRES method with BR capturing capabilities a proper BR model had to be chosen. Two basic requirements were set for such model [8]:

1. Business rules expressed according to the model have to be unambiguous and as close to natural expressions as possible;
2. BR classification should be based on functional properties of rules.

As one can see, we were looking for a model, which would allow the expression of rules according to natural language patterns. Because rules were to be captured during the specification of requirements, such a level of formalisation is considered as sufficient [13]. The use of rigid formalism would not be advisable in this phase because of risk of losing the understandability of requirements as well as burdening the communication between analyst and stakeholder.

After the inspection of known BR models, which included GUIDE project [9], Barbara von Halle model [18], *Link* model developed in Manchester University [20], Tempora method's BR model [19], it was decided to use RuleSpeak language by Business Rules Solutions (BRS RuleSpeak) [16]. This particular language or model was chosen because it is the most detailed and sophisticated model of those mentioned. It is worth mentioning that BR templates can be developed even according to the specifics of the particular universe of discourse – such an approach was proposed by Malcolm Chisholm [7]). However, we did not adopt such ideas, because BRS RuleSpeak is already well known in the IT community and it fully satisfied the needs. Development of another BR model in such circumstances would be very questionable indeed, as promotion of known ones should yield better results.

In BRS RuleSpeak, business rules are divided into three classes according to the way they react to events. These classes include rejectors (1 BR type), producers (2 types) and projectors (8 types). For each BR type, one or more natural language templates are defined. Each rule statement begins with a subject, which can be a term, fact, value, process or procedure [16]. Terms and facts (structural business rules) are considered as elements of a fact model [14; 15]. As an example, one of rejector templates can be given:

<Subject> MUST/should [not] <fact> [if/while <condition>].

In the given case, the subject can be either a term or a fact. Rule expressed using this template can be as follows: "Forests cadastre group manager should not be issuing warrants while on holidays".

BRS RuleSpeak is the basis for two submodels which were originally used to extend the ODRES metamodel by the ability to constrain the elements of static structures using rules. The simplified structure of the submodels is presented in Figure 2, while the detailed description can be found in [8]. Model supports modifiable BR templates (table *Template*). Each template starts with a certain type subject (table *SubjectType*), consists of certain type elements (table *TemplateElement*) and can include certain reserved text (table *ReservedText*). Business rules (table *BusinessRul*e) can only be expressed according to the chosen template and must include all elements from that template (table *BRExpressionCompnent*). Rules can also reference decision tables (table *DecisionTable*), which allow contraction of a set of rules with common pattern and purpose to one rule. [16]. Tables *BusinessRule* and *RelevantModelElement* should be related to the elements of requirements that can be the subjects of the BR or can be used in its components respectively (this will be elaborated in Section 4).

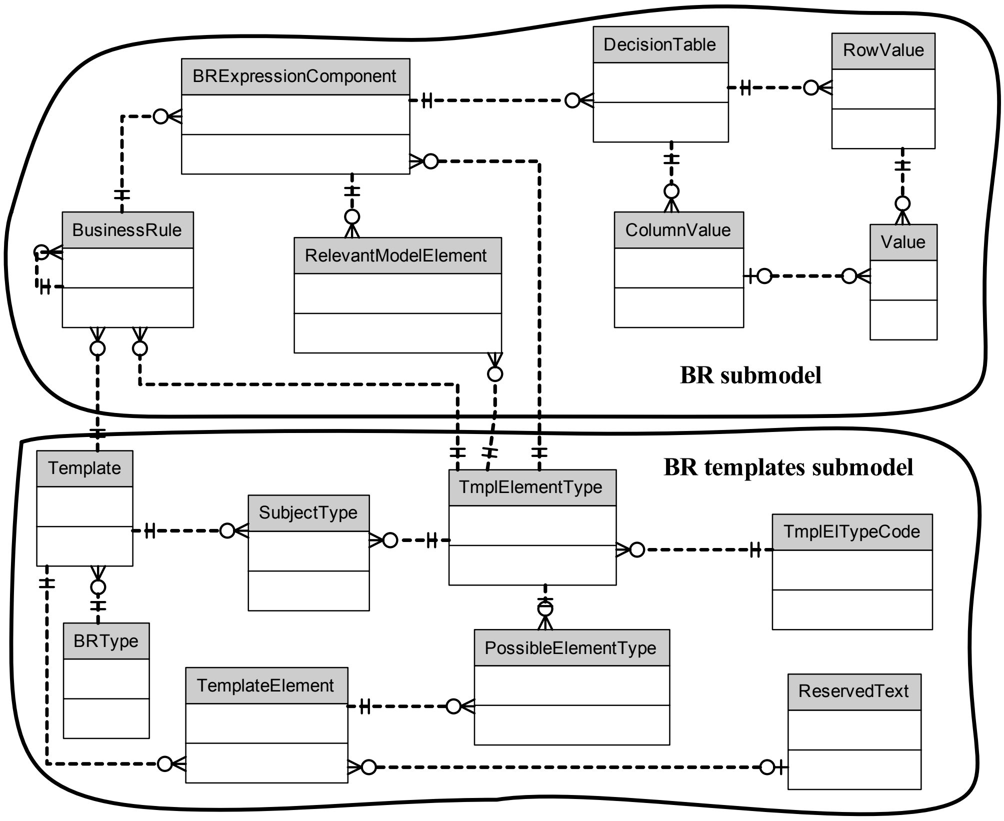

Figure 2. Simplified logical structure model of BRS RuleSpeak templates and business rules expressed according to these templates

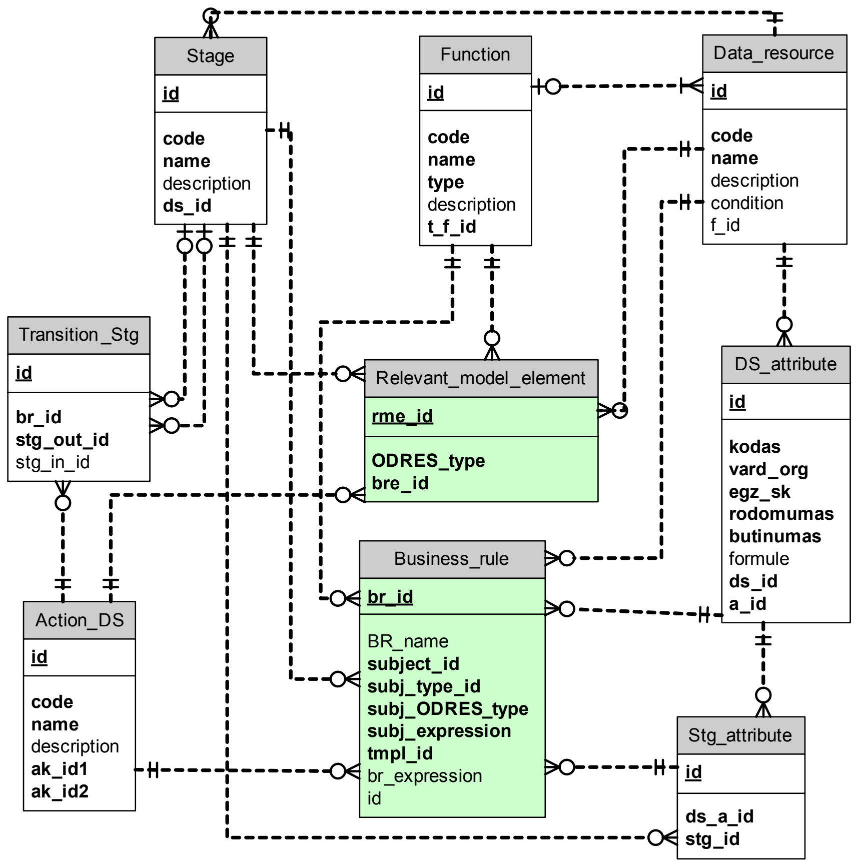

Figure 3. Relation of BR submodel to existing ODRES repository tables for the elaboration of dynamic elements of specification (darker tables represent the relevant fragment of BR submodel)

4. The extension of ODRES dynamic elements specification capabilities by business rules

To capture business rules for dynamic IS aspects specified using ODRES method, its repository model's BR submodel was related with a number of elements discovered using ODRES. A relevant fragment of requirements repository static structure model is presented in Figure 3. Here only the darker tables are from the BR submodel as they are the only ones that relate this submodel to the tables of the original ODRES repository model. During the extension process the following aspects of the use of dynamic elements in BR expressions were analysed:

1. whether a dynamic element can be a subject of BR, i.e. can it have any properties, which could be expressed in business rules (related through table *Business_rule*);
2. whether a dynamic element can be used in the body of BR (apart from subject's part), i.e. in conditions, formulas, etc. (related through table *Relevant_model_element*).

Basing on the analysis, BR submodel was related with the following dynamic aspects of requirements: functions (model table *Function*), stages (table *Stage*), data resource's actions (table *Action_DS*) and data resource attributes assigned to a certain stage and action (table *Stg_attribute*). The essence of the new relationships is best exposed by the explanations given in Table 1.

Table 1. Relation of BR submodel to the existing tables of ODRES model

ODRES model table, with which BR are being related	**BR submodel table and the explanation of the relation**	
	Business_rule	*Relevant_model_element*
Function	Function can be a suject of BR. E.g.: **BR1** "Warrant-issuing must be disabled, while application to issue the warrant has not been registered in the system".	Function expression can be used in the BR condition. E.g.: **BR6** "Change of forest subenterprises bondage with forest enterprises must be disabled, if the user is not registered as an administrator".
Stage	Stage can be a subject of BR. E.g.: **BR2** "Plot deletion must be disabled, if this plot is associated with at least one valid object of the Red Book".	Stage expression can be used in the BR condition. E.g.: **BR7** "Plot closing must be disabled, while plot's dissociation from the objects of Red Book has not been carried out ".
Data_resource	Data resource can be a subject of BR (this was discussed in [8]).	The data resource name or its form can be used in BR expression conditions and formulas or it can be a component of BR expression (this was discussed in [8].
DS_attribute	Data resource attribute can be a subject of BR (this was discussed in [8]). E.g..: **BR3** "Plot's number must be larger than 200, if the plot is a linear object".	Data resource attribute name or its form can be used in BR expression conditions and formulas or it can be a component of BR expression (this was discussed in [8]). E.g.: **BR8** "Cut down plot's volume must be computed as plot's overall volume divided by plot's section, which had an economic measure applied".
Stg_attribute	Data resource attribute, which is used during a certain stage, can be a subject of BR. E.g.: **BR4** "Plot number must be larger than 200, if the plot is a linear object".	There is no relationship because it would duplicate the relationship with a table *DS_attribute*.
Action_DS	Action can be a subject of BR E.g.: **BR5** "Saving real estate register data about the plot must be disabled while the correct register record number has not been entered".	The statement of the action can be used in BR condition. E.g.: **BR9** "Plot's functional zones cleanup must be executed when plot's protected territory cleanup is being carried out".

Note that business rules, which constrain data resources attributes (table *DS_attribute*), will be relevant at all times (according to the rules condition, provided such a condition exists). On the other hand, rules, which constrain the same attributes as they're used during the specific stage of a specific action (table *Stg_attribute*), will only be relevant for this stage. Thus one can simplify the rule's condition if it includes references to a specific action or stage. For example, BR4 in Table 1 means the same as BR3 if data resource's *plot* attribute *number* is related with a stage *registration-of-linear-objects* and action *register-linear-object*.

5. Conclusions

Functional requirements specification method ODRES is overviewed in this paper. The method simplifies analyst's job during the discovery of requirements as well as their specification. Consistent and close to natural requirements specification framework improves the quality of the resulting specification but slows down the specification process.

Constraints and rules specified under the ODRES method in majority of cases are stored in the specific requirements repository in a non-structural manner, while some of them can not be recorded at all. This results in the reduced efficiency of the method when automating the IS design process. This shortcoming is partially eliminated by introduction of BR approach. Requirements in the form of business rules are captured according to the BRS RuleSpeak BR model, which serves as the basis for the introduced extension of ODRES specification repository.

The use of BR allows for easier automation of IS design process. However, the essential benefit of such extension lies in the new capabilities to constrain dynamic requirements elements with various rules, the complexity of which depends only on the extendable BR template model.

The use of business rules narrows the gap between stakeholders and IS analysts, because BR-based requirements maintain high understandability for the future users of IS while also remaining formal enough to be used during the IS design.

The presented extension completes the ODRES extension, which now supports the use of business rules for both static and dynamic elements of specification. It can be considered as a result of a successful fusion of ODRES and BR-based requirements specification method, which were independently developed in Kaunas University of Technology.

References

[1] **R. Barker, R. Longman.** Case Method, function and process modeling. *Addison-Wesley*, 1992.

[2] **R. Butkienė, R. Butleris.** Verification Rules of Computerised Information System Model with Respect to Data Resource Processing. *Informatica, Vol.*12, *No.*3, 2001, 347-372.

[3] **R. Butkienė, R. Butleris.** The Approach for the User Requirements Specification. 5*th East-European conference ADBIS*'2001, *Research Communications, Vilnius* (2001b), 225-240.

[4] **R. Butkienė, R. Butleris, T. Danikauskas.** The approach to consistency checking of functional requirements specification. *The* 6*th* World Multiconference *on Systematics, Cybernetics and informatics. Proceedings of International Conference, Vol.*18, *Orlando*, 2002, 67-72.

[5] **R. Butleris, R. Butkienė, T. Danikauskas.** Business modeling for elicitation of information requirements. *Business operation and its legal environment: processes, tendencies and results. Proceedings of International Conference, Riga: Turima*, 2002, 67-73.

[6] **R. Butleris, T. Danikauskas.** Life cycle of information flows specification during information system design. *Information technologies*'2003, *Conference papers, Kaunas, Technologija*, 2003, XIV-41 - XIV-46 (*in Lithuanian*).

[7] **M. Chisholm.** How to Build a Business Rules Engine: Extending Application Functionality through Metadata Engineering. *Morgan Kaufman*, 2003.

[8] **T. Danikauskas, K. Kapočius, R. Butleris.** Enrichment of functional requirements specification method with business rules. *Information Systems Development. Advances in Theory, Practice and Education –* ISD'2004. *Proceedings of International Conference. Vilnius: Technika*, 2004, 194-205.

[9] GUIDE Defining Business Rules: What Are They Really? (3rd *ed.*) *Business Rules Group, (2000). http://www.businessrulesgroup.org/first_paper/br01c0.htm*.

[10] **R. Gustas**. Semantic and Pragmatic Dependencies of Information Systems. *Kaunas: Technologija*.

[11] **K. Kapočius, R. Butleris.** Business Rules Driven Approach for Elicitation of IS Requirements. *Proceedings of* 9*th World Multi-Conference on Systemics, Cybernetics and Informatics. Orlando, USA*, Vol.IV, 2005, 276-281.

[12] **M. Kirikova, J.A. Bubenko.** Software Requirements Acquisition through Enterprise Modeling. *Proceedings of* 6*th Int. Conference on Software Engineering and Knowledge Knowledge Engineering* – SEKE'94, *Jurmala, Latvia*, 1994.

[13] **T. Morgan.** Business Rules and Information Systems. *Addison-Wesley*, 2002.

[14] **R.G. Ross.** What Are Fact Models and Why Do You Need Them? *Part* 1, 2000. *BRCommunity.com, http://www.brcommunity.com/cgi-bin/x.pl/commentary/b008a.html*.

[15] **Ross R. G.** What Are Fact Models and Why do You Need Them? (Part 2). BRCommunity.com, http://www.brcommunity.com/cgi-bin/x.pl/commentary/b008b.html, (2000b) .

[16] **R.G. Ross.** Principles of the Business Rules Approach. *Addison-Wesley*, 2003.

[17] **A. Solvberg, D.C. Kung.** Information systems Engineering. *Berlin Heidelberg New York: Springer-Verlag*, 1993.

[18] **B. Von Halle.** Business Rules Applied: Building Better Systems Using the Business Rules Approach. *New York: John Wiley & Sons*, 2001.

[19] **B. Wangler.** Contributions to Functional Requirements Modelling. *Royal Institute of Technology, Akademitryck, Edsbruk*, 1993.

[20] **M.N.Wan-Kadir Wan, P. Loucopoulos.** Linking and Propagating Business Rule Changes to IS Design. *Research papers of the Thirteenth International Conference on Information Systems Development* (*ISD* '2004), *Kluwer/Plenum Press*, 2005.

Received August 2006.