Preparing business rules templates and object role modelling for transformations

Olegas Vasilecas, Sergejus Sosunovas

Abstract: *It is common for business people to operate with natural language statements. BR (BR) in this sense can be presented using the variation of structural language – BR templates. However, BR templates are not formal enough to be automatically processed. Furthermore, BR templates have to be in consistence with enterprise model, especially with its static structures – business objects. Object role modelling (ORM) language in this paper is selected for the representation of business objects and their roles. Since presented integrated metamodel of ORM and BR templates conforms to MDA requirements it enables automatic transformation of BR templates and underlying ORM models during automation of information system development lifecycle.*

Key words: *ORM, BR templates, metamodels, transformation.*

INTRODUCTION

Today's economy is commonly named as a knowledge economy denoting that stake-holders of the adequate knowledge may gain competitive advantage over its rivals. It is widely recognized that BR are one of the most important part of the enterprise knowledge [1]. The usual form for BR to appear in the enterprise is to be buried in the numerous guidelines, policies and other documents. However, BR specified only in natural language are largely inaccessible to computer programs, decision making, quality assurance initiative and management of the enterprise.

The problem of the accessibility of BR for the automatic processing can be partly solved using BR templates. BR template is intended to define the structure of BR of some particular type. There are gaps left in BR templates to be filled later when the actual activities on specifying business rules is executed. In addition to automatic processing of BR another expected advantage of such approach is that users feel comfortable with BR templates as if they were working with natural language statements [2],[3].

But it is very difficult to define BR templates acceptable for each enterprise in advance. Especially it becomes crucial in the world wide context when adaptation to different cultures and as consequence languages is a must. Therefore it is necessary to enable specification of custom user centric BR templates. For custom templates to be still available for automatic transformation they should be built on the basis of well defined template definition constructs.

One more fact related to the BR template constructs is that BR are rarely used stand alone, on the contrary, they tend to refer to other enterprise models in particular business data and business process models. As a consequence it is necessary to enable BR template constructs to refer to other enterprise models. In this paper we will concentrate on integration of BR templates with data model represented using object role modelling ORM [4] mostly.

There are several possible integration levels of ORM and BR templates [5]: technical integration of tools considering APIs and tools interfaces, conceptual integration of metamodels of description formalisms combined with hard and soft constraints, semantically integration of semantics of description techniques using a common semantic model, and finally methodical integration by an embedding in the development process. In this paper metamodels based integration of ORM and BR templates was selected as most appropriate.

During the last decade the term metamodel became very popular and is used in different areas sometimes with different meaning. In this paper metamodel is understood as a precise definition of the constructs and rules needed for creating semantic models [6]. As it is pointed out in [6], metamodelling has been around at least since the late 1980s, but

with the advent of the Internet, business process and data integration is obviously a first-order priority. One of the most important reasons for selecting metamodel based integration is model driven architecture (MDA) [7] proposed by object management group (OMG). MDA assumes development of platform independent model and its transformation to the platform specific models. Because the next step from specification of BR is often their implementation into information system (IS) MDA approach can be used to transform BR and related models from business level to the IS level. In order to make ORM and BR templates available for the MDA transformation proposed integrated metamodel should satisfy MOF requirement [8].

Generally, there is distinction of three kinds of BR: dynamic rules aimed to govern business actions, static rules aimed to manage behaviour of business objects and derivative rules – expressions that specify how certain statements can be derived [9]. Only static rules which directly relates to ORM are the subject of this paper.

The purpose of this paper is to describe integrated ORM and business rules templates metamodel that enables automatic transformation of BR defined on the basis of user specific BR templates.

The rest of the paper proceeds as follows. In the section 2 metamodel of ORM is presented. Section 3 describes metamodel of BR templates. Section 4 provides examples that demonstrate application of integrated metamodel to different types of BR templates and ORM model.

ORM LANGUAGE ELEMENTS

ORM is primarily a method for conceptual fact modelling. In Europe the method is often called NIAM (Natural Language Information Analysis Method) [4]. ORM is so called because it pictures the world in terms of objects (entities or values) that play roles (parts in relationships). In contrast to other modelling approaches e.g. Entity-Relationship (ER) and Object-Oriented (OO) approaches, ORM makes no explicit use of attributes [4]. ORM can be used as a kernel for object role modelling techniques, upon which different drawing styles can be based. One of the prominent advantages of such a kernel is the possibility to build a CASE-tool supporting multiple methods [10]. In this context existence of ORM metamodel would allow to use for the transformation not only ORM itself but related modelling techniques as well (e.g. ER).

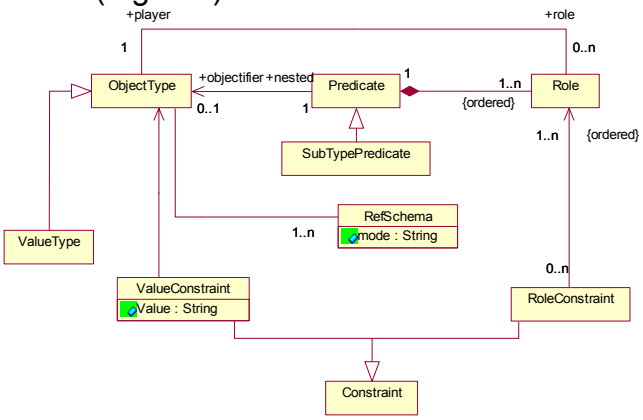

Figure 1 ORM metamodel (fragment)

Comparison of ORM and UML can be found in [11]. ORM compared to UML is identified as more suitable to express different kinds of BR found in data-intensive applications [11]. Entity relationship modelling and ORM are compared using framework for comparison of conceptual modelling language in [12]. Both comparisons identified ORM as more advanced in the sense of expressive power, practical convenience efficiency and learnability. BR specified in ORM can be easily transformed to natural language [4]. ORM has a flexible notation for BR representation. Examples and analysis of ORM usage for BR modelling are presented in [13]. However BR that can be expressed in

ORM are limited to the schema level. An exception is BR related to possible values of the value entity. It is not possible to present instance level BR with pure ORM. In this case ORM can be supplemented with BR templates.

In the presented metamodel ORM model is presented by ORMmodel metaclass. General ORM model element is denoted by ModelElement metaclass. It is abstract metaclass. All other ORM model elements inherits name attribute from ModelElement. ObjectType is used to represent entity type and ValueType denotes a lexical object type. RefSchema metaclass indicates how each instance of the entity type may be mapped via predicates to a combination of one or more values. Its attribute mode (RefSchema.mode) indicates how values relate to the entities (e.g. plus sign "+" may be added if the values are numeric). Predicate metaclass is used to present n-ary predicate. Each ORM element used in predicate plays a role which is presented by Role metaclass, name of the role is stored in the name attribute. Association relationship between Role and ObjectType metaclasses denotes object holes of ORM predicate. Some times it is necessary for the relation to be treated as object it is called objectification. Association between Predicate and ObjectType metaclasses is used for the objectification of relationship. All ORM specific constraints are inherited from abstract metaclass Constraint. It is further specialized to ValueConstraints dedicated to constraint values of object types and RoleConstraint metaclass – constraints used to constraint different ways of objects can participate in relationships.

Table 1 BR template metamodel metaclasses

ORM Metaclass	Meaning
TemplateExp	Abstract Metaclass representing all possible terms
BRTemplate	BR template, metaclass aggregating BR terms
DeterminerExp	The determiner for the subject; from the following, the one that makes the best business sense in the statement. Possible values for the name attribute: each, any, etc.
SubjectExp	A recognizable business entity. SubjectExp metaclass contains reference to the ObjectType metaclass from ORM metamodel
CharacteristicExp	The business behaviour that must take place or a relationship that must be enforced. CharacteristicExp metaclass refers to the ORM predicate metaclass of the ORM metamodel
FactExp	A relationship between terms identifiable in the fact model, together with defined constants. The relationship may be qualified by other descriptive elements in order to specify the applicability of the rule precisely. Refers to the predicate metaclass of the ORM metamodel
FactListExp	A list of fact items
LiteralExp	Instance of metaclass LiteralExp contains character string in the name attribute
NumericExp	Instance of metaclass NumericExp contains numeric value in the name attribute
NumParamExp	Numeric parameters.
ClassificationExp	A definition of a term in the fact model. This typically defines either the value of an attribute, perhaps called "state" or something similar, or a subset of the objects in an existing class. May contain reference to the value entity from ORM model storing "state" value
KeywordExp	Represents keyword expression of the business rule template

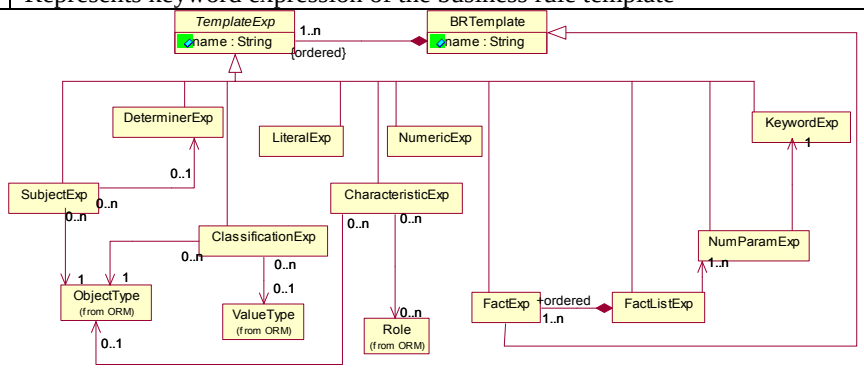

Figure 2 BR template metamodel

BR TEMPLATE LANGUAGE ELEMENTS

BR templates is a common way to present BR in a business community. Defining BR it is necessary to select particular template type from a list and provide necessary references to the data model elements. Besides saving analysis time, this helps enforce consistency of model. Properly defined BR templates can be processed automatically [3]. BR templates are formed from different kind of terms [3], [4]. The BR template metamodel is presented in . Similarly as in ORM metamodel, BR template metamodel contains metaclass which aggregates template expression elements. The other template elements are inherited from abstract "TemplateExp" metaclass. Fact expression (metaclass FactExp) is also inherited from BRtemplate metaclass. It is done in order to reuse fact sub-templates, as it is demonstrated in the examples presented in this paper.

Three metaclasses from ORM metamodel are referred in BR template metamodel. Subject expression refers to the ORM Object type. Usually subject expression is mandatory expression in BR templates. Classification expression refers to ORM Value type to denote where actually the classification value is stored, however this reference is optional and the value can be stored by the instance of classification expression. Characteristic expression refers to the Role metaclass from ORM metamodel.

It is important to note that the main aim of the presented metamodel is to provide precise semantics for the BR templates not for the BR resulting from using the template. However it is clear from the metamodel that the semantics of only one element – "KeywordExp" affects the meaning of the instances of BR templates and consequently the meaning of BR. Therefore it is recommended to use semantically well defined vocabulary SBVR [14] for construction of BR templates. From presented terms it is possible to construct variety of BR templates. BR templates can be defined in natural language but in order to achieve higher formality level it is better to use OCL.

METAMODEL APPLICATION EXAMPLES

As it was mentioned before BR template metamodel do not provide any specific templates, therefore before processing with specification of BR it is necessary to select or construct new BR templates. Consequently one more purpose of this chapter is to present how the BR templates can be formally defined within the limits of provided metamodel. Conference organisation domain which is common for the majority of readers is selected for the examples. Figure 3 presents several ORM facts types from this domain.

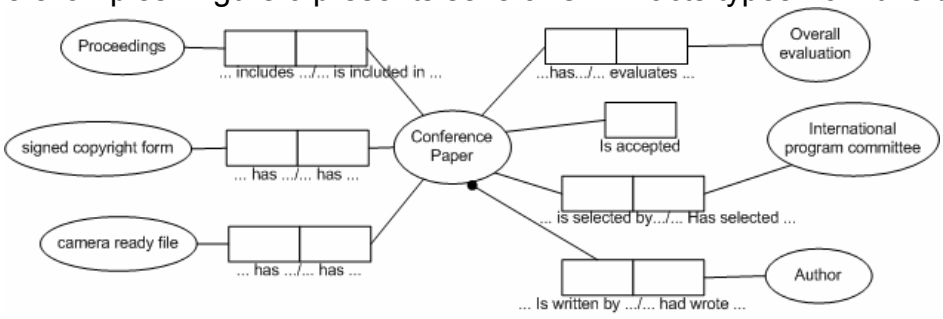

Figure 3 Fragment of ORM model presenting fragment of conference conceptual model

Basic constraint template [3] is the most common business rule template, establishes a constraint on the subject of the rule. The Morgan notation for this template is presented in table 2. The table 2 also gives the formal definition of the template in OCL. There are two OCL constraints presented with different context metaclasses. The former (context: BRTemplate) defines the general basic constraint template structure, with fact expression at the end. It is important to notice that Morgan do not formalize fact expression, however the fact expression is not ordinary template element since it contains several components that can be classified as other metaclasses. This dual nature of fact expression results in its independence from the template and enables reuse. Therefore it

is feasible to define fact expressions separately, as it is done with the second OCL statement of the definition of basic constraint template (context: FactExp). Resulting business rule template and ORM metamodel of basic constraint rule is presented in figure 7.

Presented OCL constraints follow the simple rules of defining BR template, in particular they constraint the position of template expression of particular type within ordered set and provide the allowed variations of keyword expressions. Despite of simplicity of provided OCL statements they unambiguously define the structure of the BR templates. But it is possible to employ the full power of OCL for specifying complex custom templates based on integrated metamodel. Additionally it is important to note that none of the presented BR templates can be expressed in pure ORM. At the same time specification of BR without ORM model would be less expressive. Whereas using integrated metamodel the smooth and harmonic integration of ORM and BR templates becomes possible.

Table 2 Detailed description of basic constraint BR template

Template name	Basic constraint.
Morgan notation	<det> <subject> (must | should) <characteristic> [(if | unless) <fact>]
OCL template definition	Context: BRTemplate Inv: If name="Basic constraint" then TemplateExp->at(1)->oclIsKindOf()=DeterminerExp and TemplateExp->at(2)->oclIsKindOf()=SubjectExp and TemplateExp->at(3)->oclIsKindOf()= KeywordExp and (TemplateExp->at(3).name="must" or TemplateExp->at(3).name="should") and TemplateExp->at(4)->oclIsKindOf()= CharacteristicExp and TemplateExp->at(5)->oclIsKindOf()= KeywordExp and (TemplateExp->at(5).name="if" or TemplateExp->at(5).name=" unless") and TemplateExp->at(6)->oclIsKindOf()= FactExp TemplateExp->at(6)->oclAsType(BRTemplate).name ="Greater than" EndIf Context: FactExp Inv: if name="Greater than" then TemplateExp->at(2)->oclIsKindOf()=CharacteristicExp TemplateExp->at(3)->oclIsKindOf()=KeywordExp TemplateExp->at(4)->name="is greater than" TemplateExp->at(5)->oclIsKindOf()=NumericExp EndIf
Example	Each Conference Paper must be accepted if overall evaluation is greater than 5

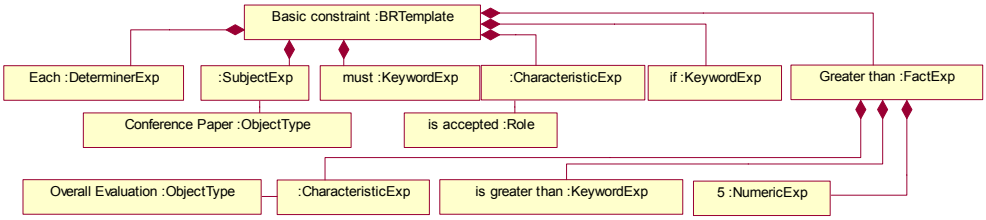

Figure 4 Business rule presented as an instance of basic constraint BR template metamodel.

CONCLUSION

In this paper we have presented an integrated metamodel for the object role modelling language and BR template. The application of the metamodel is demonstrated on the concrete BR templates statements. The metamodel delivers a more precise and detailed view of the ORM and BR templates. As a result of using only well-known modelling concepts of UML which are compliant with Meta Object Facility (MOF) the metamodel can easily be read by everybody familiar with UML.

Provided integrated metamodel allows precise and consistent with data model definition of BR. It is important to note that although the metamodel provides a precise description of the abstract syntax of ORM and BR templates, it does not define any specific template. Additional OCL constraints have to be provided in order to define the most often used BR templates.

REFERENCES

[1] Bajec, M., Krisper., M. A methodology and tool support for managing BR in organizations. *Information systems journal,* 2004.

[2] Herbst H. A meta-model for BR in systems analysis. *J. Iivari, K. Lyytinen, M. Rossi (Eds.): Proceedings of the Seventh conference on advanced information systems engineering (CAiSE'95)*, Berlin et al.: Springer 1995, pp.186-199

[3] Morgan, T. *BR and Information Systems,* Addison Wesley, 2002

[4] Halpin, T. Object-Role Modeling (ORM/NIAM) *P. Bernus, K. Mertins and G. Schmidt (Eds.): Handbook on Architectures of Information Systems*, , Springer-Verlag, Berlin, 1998,

[5] Braun,P., Lötzbeyer, H., Schätz, B., Slotosch, O., Consistent Integration of Formal Methods *S. Graf, M. Schwartzbach (Eds.): Tools and Algorithms for the Construction and Analysis of Systems: 6th International Conference, TACAS 2000*, LNCS 1785, Springer-Verlag Berlin, Germany, March/April, 2000

[6] Overview on Concepts for metamodeling URL: http://www.metamodel.com

[7] Siegel, J. Developing in OMG's Model-Driven Architecture. OMG document: 01-12-01, OMG, 2001. URL: http://www.omg.org

[8] OMG. Meta Object Facility (MOF), 1.4. OMG, 2002. URL: http://www.omg.org/cgi-bin/apps/do_doc?formal/02-04-03.pdf

[9] Boman, M., Bubenko, J., Johannesson, P., Wangler, B. *Conceptual Modelling,* Prentice Hall, 1997.

[10] Brouwer, S.,J., Martens, C.,L.,J., et al. Towards a Unifying Object Role Modelling Theory. In *T.A. Halpin and R. Meersman (Eds.) Proceedings of the First International Conference on Object-Role Modelling (ORM-1)*, Magnetic Island, Australia, July, 1994, pp. 259–273,

[11] Halpin ,T. Information analysis in UML and ORM: a comparison *Advanced topics in database research,* vol. 1, 2003, pp: 307 - 323

[12] Halpin, T.,A., Oei, J.,L.H. Tech. Report 92-29, Dept. of Informatics, Uni. of Nijmegen, Nov. 92.

[13] Hargreaves A. Expressing BR using Object Role Modelling *Proceedings of the 17th NACCQ 2004*
URL: http://www.naccq.ac.nz/conference05/proceedings_04/hargreaves.pdf

[14] Adaptive et all. Semantics of Business Vocabulary and BR (SBVR) Revised Submission to BEI RFP br/2003-06-03, 2005. URL: http://www.omg.org/cgi-bin/doc?bei/05-01-01

ABOUT THE AUTHORS

Prof. Olegas Vasilecas, PhD, Department of Information Systems, Vilnius Gedinimas Technical University, Phone: +370 52744860, E-mail: olegas@fm.vtu.lt.

Sergejus Sosunovas, PhD student, Department of Information Systems, Vilnius Gedinimas Technical University, Phone: +370 68686381, E-mail: sergejus@isl.vtu.lt.