

# Classification and Representation of Business Rules

Marcel Weiden<sup>1</sup>, Leo Hermans<sup>1</sup>, Guus Schreiber<sup>2</sup> and Sven van der Zee<sup>3</sup>

<sup>1</sup> *Everest,  
Reitscheweg 55, 5232 BX 's Hertogenbosch, The Netherlands  
E-mail: {M.Weiden,L.Hermans}@everest.nl*

<sup>2</sup> *Social Science Informatics, University of Amsterdam,  
Roeterstraat 15, 1018 WB Amsterdam, The Netherlands  
E-mail: schreiber@swi.psy.uva.nl*

<sup>3</sup> *CIBIT  
Arthur van Schendelstraat 570, Utrecht, The Netherlands  
E-mail: svdzee@cibit.nl*

## Abstract

In this article we look at the relatively new notion of “business rules”. We propose a new classification scheme for business rules, which is aimed at providing more support for business-rule elicitation. We show how this scheme was applied in a domain of offering mortgages. As a result we formulate guidelines for eliciting business rules and for choosing the most suitable representation format for business rules.

## 1. Introduction

### 1.1 Business rules

Business rules are a relatively recent addition to the information-engineering repertoire. Business rules are somewhat similar to integrity constraints in databases: they define some business invariants in a declarative manner. Within the Guide project [4] a business rule was defined as “a statement that defines or constraints some aspect of the business”. Business rules reside at the borderline between business engineering and software engineering. The Business Rules Group therefore gives two definitions of business rules:

“From the business perspective a business rule is a directive, which is intended to influence or guide business behavior, in support of business policy that is formulated in response to an opportunity or threat. From the information-system perspective a business rule is a statement that defines or constrains some aspect of the business. It is intended to assert business structure, or to control or influence the behavior of the business.” [1]

The role of business rules is mainly to describe constraints on the business structure and process in a “what” manner and not in a “how” manner. For example, in a business rule we would say that activity A and activity B need to be performed on the same day, without making any statements about their ordering unless this is really needed.

In information-system development business rules typically play the role of requirements, which the final system should meet. Here lies also much of the rationale behind the business-rule approach. It is hoped that through their declarative nature business rules are less likely to change. From the database perspective business rules are very much like database integrity constraints [2].

Early work on business rules has been done in the Guide project [4]. In this project three types of business rules were distinguished:

1. *Structural assertions*: defining concepts and relations between concepts.
2. *Action assertions*: defining when and how certain actions are done.
3. *Derivations*: knowledge statements that help to derive new facts from known facts.

This classification scheme has been extended by Ross [6]. He describes several types of action assertions such as “instance verifiers”, “type verifiers”, “functional evaluators”, and “calculators”. A problem with the Guide/Ross classification scheme is that it is mainly based on differences in syntactical rule structure, e.g., verifying a type, evaluating a function. A business-rule classification scheme should preferably support the elicitation of business rules in a domain. It would be difficult to go to a business worker and ask, for example, “what functions do you evaluate?”.

## 1.2 Aim and approach

In this article we investigate some methodological aspects of business rules. More specific, we are interested in developing and testing a more semantically-oriented classification scheme for business rules. The semantic classification scheme we propose is based on the business-modeling approach of the knowledge-engineering methodology CommonKADS [7]. The CommonKADS “context models” enable respectively an “organization”, “task” and “agent” perspective on an application domain. This approach appeared suitable as a basis, because the context models describe several complementary views on a business process without going into any detailed system-analysis process. Subsequently, we used the classification scheme in a case study. The case study gave rise to a number of modifications and refinements of the classification scheme. In this article we only show the classification scheme after modification (see [9] for a description of the initial scheme). From this case study we are able to define some guidelines for using the classification scheme.

It will be clear that there is considerable overlap between “business rules” and existing notions in system analysis. The link with integrity constraints was already mentioned. Other business rules represent type constraints for which ER modeling and UML class diagrams provide representations. In this article we will initially look at everything as “rules”, but it will become clear that not everything that falls under the definition of “business rule” is necessarily best represented in a textual “rule” format. Therefore, we also formulate guidelines for choosing an adequate representation format for business rules.

This article is structured as follows. In Section 2 the proposed business-rule classification scheme is discussed. Section 3 describes the case study in which we applied this classification scheme. From the experiences gained we formulate in Section 4 a number of guidelines for representing business rules. In Section 5 we discuss strengths and limitations of this work and define goals for further research.

## 2. Business-Rule Classification Scheme

Our classification scheme is based on the CommonKADS context-modeling framework [7]. This framework has been used extensively to model knowledge-intensive business processes. Using this framework we can classify rules based on the role they play in the business process. Typically, the framework assumes that an initial process model of the business has been created in terms of the main processes and tasks. In CommonKADS this business process model is typically represented in the form of a UML activity diagram (see Figure 1).

In the classification scheme we use the following meanings of the terms “process” and “task”:

- **Processes** are the most general way of describing behavior in a business. Processes are not linked to a certain location or actor. A process can serve more than one business goal.
- A **task** is defined as behavior that can be linked to a certain (type of) actor. Tasks are performed in the context of processes. Although one can make a distinction between “task” and “activity”, we will use these two terms as synonyms in the context of this article.

Table 1 shows the business-rule classification scheme that we developed. The scheme can be used to classify business rules according to their semantic properties, (their role in the business process). In the business-rule classification scheme fifteen business-rule *types* are defined. The rule types are grouped into three categories; *structural*, *behavioral* and *managerial* rule types. These categories represent different views on the business. The structural business rules are concerned with describing static aspects of a business. Behavioral rules define the conditions on the execution of tasks in the business. The rules in the managerial category define higher-level constraints on the business. For every rule type, a typical question indicating the nature of the business-rule type is given. These questions can be used as a starting point in business-rule elicitation.

Category	Rule Type	Elicitation question
Structural	Concept Structure	What are the central concepts and their relations in the domain?
	Persistency	How long should a representation of a certain object be kept?
	History	Should the history of an object in the domain be recorded?
Behavioral	Information Flow	Which type of information does a task need from other tasks?
	Control Flow	Which rules control the execution order of tasks?
	Pre-conditions	What conditions must be present for a task to be able to execute?
	Post-condition	Which conditions are present after a task has finished executing?
	Frequency	How often will a certain task be performed?
	Duration	How long does or may it take to perform a certain task?
	Task knowledge	What kind of knowledge is needed to perform a certain task?
Managerial	Organization	What can be said about policies in the organization?
	Goal & Value	What is the goal of a process?
		What is the expected added value of a process?
	Actor <sup>1</sup> Competences	Which skills does an actor need?
	Actor Responsibilities	Which tasks is an actor allowed to perform?
Resources	Which rules are used to govern the use of resources?	

**Table 1: Business-Rule Classification Scheme**

The distinction between a structural and a behavioral viewpoint is well known and underlies many system-analysis methods including UML. Together, these two viewpoints define the *internal* view of

---

<sup>1</sup> An actor can either be a human or an information system. Both types of actors have certain skills and are expected to perform a task (or several tasks) within the organisation. Human actors can have a responsibility to perform a task. Depending on your view, information system actors can also have responsibilities.

the business process. The managerial viewpoint adds the *external* view, introducing notions such as goal, value, and resources needed.

The different rule types will be described in more detail with the use of examples from a car rental domain. These examples are derived from the EU-Rent case study concerning a car-rental company described by the GUIDE Business Rule Project. (see Appendix D of [4]).

## 2.1 Structural Rules

Structural rules describe static aspects of a business. Information can be seen as a static aspect. Various dimensions concerning information within a business lead to the following three rule types.

### Concept Structure

Concept structure rules describe types of concepts and relationships between concepts in the business. Initially, concepts are typically used to describe the vocabulary of terms for the business.

In the EU-Rent organization, the cars that can be rented are divided into a number of categories. The categories determine the price of a rental. The rules describing what types of cars are in a certain category belong to the concept-structure rule type. An example of such a rule is: "Each car type has a rental class. For example, a Ford Focus is a class two rental".

### Persistency

Rules of the persistency type determine how long certain pieces of information in an organization should be kept available. For example, it might be useful to keep information about all customer-related transactions. On the other hand, the law might prevent keeping records of customers longer than a certain period.

EU-Rent is interested in the behavior of its customers. Therefore, all kind of information about the customers should be recorded by EU-Rent within legal boundaries. The legal boundary is defined through the following business rule: "No information about a rental should be kept for longer than five years".

### History

In some cases, it is important to keep track of an object's history. For example when the business is interested in the previous state of an object in a certain process. History-type business rules can also be used to specify that the different states an object goes through should be recorded.

If the same customer rents several cars at different times, it might be useful to link these rentals to a representation of the customer. This way, the information about the customer can be used to predict the customer's needs. Predicting customer's needs can only be done if each rental is regarded as linked to a certain customer. This could be expressed by the following business rule: "All rentals made by some customer should be recorded as belonging to that same customer".

## 2.2 Behavioral Rules

The category of behavioral business rules defines rule types that are concerned with the execution of tasks in the business. Behavioral rules describe dynamic aspects and govern the behavior of actors in the business.

### **Information Flow**

Information-flow rules are used to specify the required information flow between tasks. These rules describe situations in which a task needs information from other tasks to be able to execute.

EU-Rent example: "When handing over a car, always use the car that was previously assigned to this rental".

### **Control Flow**

The rules in this category are concerned with the control over the execution of tasks, such as the sequencing of tasks within a certain process.

EU-Rent example: The subtask of allocating a car to a certain rental can only be performed if the reservation for the car is checked and accepted. A business rule that expresses this constraint is: "Cars will only be allocated to customers, if the reservation for the rental is accepted".

### **Pre-conditions**

Pre-condition rules indicate conditions that must be met before a task is performed.

EU-Rent example: "A car must be physically present in a EU-Rent branch, before it can be assigned to a rental."

### **Post conditions**

A post-condition rule states a condition that must hold after execution of the task. Other rules might specify what happens if the post-condition of a certain task is not met.

EU-Rent example: "After a car has been serviced, the car is considered to be in a legal and roadworthy condition."

### **Frequency**

These rules define how often a task is executed.

EU-Rent example: "Each car must be serviced every three months or 10.000 kilometers, whichever occurs first".

### **Duration**

Most business rules concerned with the duration of a task specify that a task must be done within a certain period of time.

EU-Rent example: "Assessment of a car reservation must be done on the same day the reservation was received".

### **Task knowledge**

These rules define what knowledge is needed to correctly perform a task. The business rules in this category concern the decision rules that are used in the execution of a task.

EU-Rent example: "Rented cars must meet local legal requirements for mechanical condition and emissions for each country that may be visited during the rental".

## 2.3 Managerial Rules

Managerial rules are business rules, which the organization uses to ensure correct performance. The rules express the way the organization should behave from a managerial perspective. These rules define high-level constraints on the business processes.

### Organization

Organization-type rules describe the policies and culture of an organization. All other business rules have to be in accordance with the policies as expressed by rules of this type.

EU-Rent example: "We only rent cars in legal, roadworthy condition to our customers".

### Goal and Value

Business rules of this type describe the goals of the organization and the (expected) added value of the processes with respect to achieving these goals. This type of information can also be expressed in a policy charter. Goal and value rules are expressions of the policies, as described by the organization rules, aimed at one particular process or task.

EU-Rent example: "We want to make sure that there will be no rentals turned down because there are no cars available".

### Actor Competences

These rules define what skills are needed in an actor in order to let the actor perform a certain task correctly. An actor-competence rule can be used to describe what skills a certain type of employee should possess, but can also be used to describe the minimum requirements for an information system carrying out a certain task.

EU-Rent example: "All EU Rent employees must have a valid drivers license".

### Actor Responsibilities

These rules are concerned with the responsibilities of actors in the organization. Actor responsibility rules often say what types of tasks can be performed by a certain type of employee. They are also useful to describe the boundaries of what the organization expects from an information system.

In the EU Rent organization, there could be a class of cars for weddings etc. that are more expensive than the regular cars. A senior employee must always deal with the request for such a car. The following business rule would express this: "Special cars can only be assigned to a rental by senior employees".

### Resources

Resource rules define constraints about the amount and type of resources that are used in the business.

The most important resource for EU-Rent are the cars used for rentals. For every car category, there is certain quota indicating the number of cars needed for normal operation. The number of cars in a branch can change easily; because cars can be returned to a branch different from the one the car was rented from. In this case, the ownership of the car switches to the branch that received the car from the customer. There are rules that describe what must be done when the number of cars differs more than 10% from the quota. The rule for car losses states that: "In each car group, if a branch loses cars to take it more than 10% of below its quota, it must increase

the number back to within 10% of quota by transferring cars from other branches, or buying some cars”.

### 3. Case Study

This chapter describes a case study in which the business-rule classification scheme was applied to a realistic domain. Details of the case study can be found elsewhere [9]. In fact this case study has been used to refine an initial version of the classification scheme, thus resulting in the one proposed in the previous section.

#### 3.1 Domain of investigation

The domain for this case study is the development of a Mortgage Administration System to be used by a new Shared Service Centre for Mortgages (SSCM) of a large financial holding in the Netherlands. The customers of the SSCM are the different mortgage labels that act as independent commercial organizations within the holding. Each of these labels targets a different segment of the market and has its own commercial strategy. The business rules for the labels may therefore differ substantially and may even be conflicting. The SSCM should perform all administrative tasks involved in processing mortgage applications, producing an offer, setting up a loan and maintaining it until the loan is finished many years later. We limited this case study to the offering process.

At the start of the project, the business processes of the different labels have been integrated into a “harmonized” process model. A simplified part of this process model is shown in Figure 1 in the form of a UML activity diagram. The event starting the offering process is an incoming application for a mortgage. This application is registered, checked and subsequently assessed using a number of different criteria. Depending on the outcome of the assessment, the applicant will receive either an offer for a mortgage or a letter with the reason for the rejection of the application.

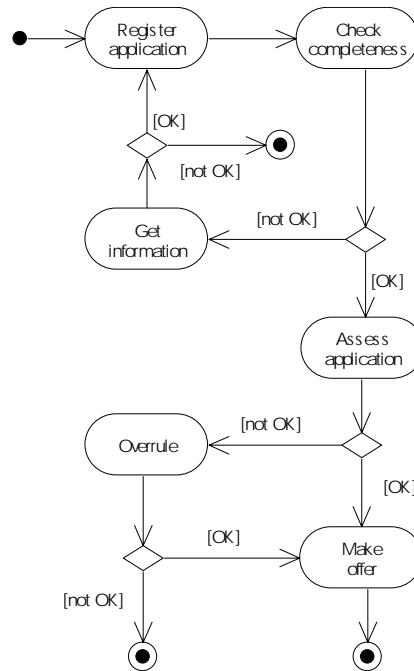


Figure 1: UML activity diagram for processing a mortgage application

## 3.2 Data Collection

Data about business rules was collected through analysis of domain documentation and a number of interviews. The business rules found were collected in a relational database. This offered an easy way to get different views on the set of rules. Every recorded rule consisted of (i) a description of the actual rule in natural language, (ii) the rule type according to the classification scheme (iii) the label for which the rule is valid, (iv) and an optional field for information about the source of the rule or other comments. The database form for entering business rules is shown in Figure 2.

The screenshot shows a window titled "Input Rules" with a subtitle "Business Rule input". The main form area contains a text input field with the text "There must be at least one applicant with the Dutch nationality who also lives in the Netherlands". Below this is a table with three columns: "Rule Type", "Extra info (opt.)", and "Source (opt.)". The "Rule Type" dropdown is set to "Task Knowledge". Below the table are four checkboxes labeled "Label A", "Label B", "Label C", and "Label D". The "Label C" checkbox is checked. At the bottom, there is a record navigation bar showing "Record: 305 of 319".

Figure 2: Database screen for entering a business rule. The checkboxes in the left corner indicate for which commercial label the rule holds.

Unfortunately, business people still working in the mortgage business of the labels were not available for interviewing. Therefore, the interviews were held with business consultants and system analysts working on the project. At the start, an unstructured exploratory interview was held with a business consultant working on the project to gain insight into the way mortgage applications are handled. The purpose of this interview was not to find business rules, but to get an overview of the domain.

Subsequently, two structured interviews were held with business analysts working on the project. This time, the aim of the interviews was to find business rules for the mortgage domain. The process model was used as a guide in the elicitation process. The process model was shown to the person being interviewed and the rules were collected by talking about the different tasks. The classification scheme was used to explain what types of business rules can be identified for tasks and actors involved. For every category, the interviewee was asked if he knew any rules of this type in the mortgage organization. Sometimes the interviewee was also presented with a potential example rule. Often, looking for rules in a certain category also produced rules belonging to other categories. As the process of entering the rules into the database was done after the interview, the order in which the rules were found did not matter.

There was also a huge amount of project documentation describing the offering process. These project documents were also analyzed in order to find business rules controlling this process. The main problem with the documents was that they focused on an information system under development. Business rules typically should have a broader scope, as they are not tied to a particular software-development project. To find out more about these missing parts, the interviewees were contacted again. They were asked to describe the activities that were not going to be supported by the system and why these activities have to be performed by humans.

### 3.3 Results

In total 320 business rules dealing with the mortgage offering process were found. These rules deal with different aspects of the process. Some concern the way the assessment should be performed, such as the following duration rule used for one of the labels: "The actual assessment (after all necessary data is collected) should not take more than two minutes". Other rules are concerned with the objects used. For instance: "The default pay off value for existing contracts is Euro 0". There were also business rules that only apply to specific situations such as this actor responsibility rule: "Offers between Euro 170,000 and Euro 240,000 must be signed by the head of the acceptance team or a credit analyst".

All the rules found in this case study were classified into one of the categories described in the classification scheme. Most of this was done after the rules were elicited and written down. Table 2 shows the number of business rules found in each category of the scheme. Also, some sample business rules are listed.

Category	Rule Type	#	Sample Rule
Structural	Concept Structure	26	If an applicant is under 18 years old, but is or has been married, the applicant is considered an adult
	Persistency	8	A valuation report is valid for one year
	History	4	All changes in the value of an asset should be recorded
Behavioral	Information Flow	3	The offer must contain the same information about the customer as was on the application
	Pre-conditions	24	An applicant must have a 'giro' account in order to receive a bank guarantee
	Post-condition	5	After the input has been checked, it is believed to be correct
	Frequency	20	All information about a customer should only be entered in an information system once
	Duration	5	The actual assessment (after all necessary data is collected) should not take more than two minutes
	Control Flow	10	An offer is only made after the assessment is completed
	Task knowledge	133	If the applicant occurs on the EVA or IVA list, the judgment on the applicants morality is negative
Managerial	Organization	28	Information collected for an assessment can not be used for commercial purposes
	Goal & Value	4	All bank employees should use the same information system to guarantee consistent information delivery and efficient use of information
	Actor Competences	29	Customers should be able to inquire information about their account via the internet
	Actor Responsibilities	18	Only senior employees can change or add secured notes to a customers file
	Resources	0	...

Table 2: Numbers of business rules found in the case study

### 3.4 Analysis

Based on the experiences two changes were made to the initial scheme we developed prior to the case study, thus resulting the classification scheme of Table 1:

1. The structural rule type "object occurrence" was dropped. All rules of this type were about cardinality constraints and were considered by analysts as object-structure rules. Consequently, the "occurrence" rules were all moved to this category.

2. The original scheme also contained a managerial rule type “quality and performance”. It turned out that it was not possible to define criteria for distinguishing these rules from the “organization” and “goal & value” rules. Therefore, we also dropped this rule type and moved “quality & performance” rules that were linked to a particular task to the “goal & value” rule set, and added the other rules to the “organization” rule set.

For some business rule categories specific reasons can be given for a relatively low respectively high number of rules:

- We did not collect many business rules of the “history” type. This can be explained by the characteristics of the process that was chosen for the study. Of all the activities concerned with mortgages, only the offering process was looked into. This process is typically performed only once for a particular application.
- There were no rules about resources collected in this case study. This can be explained by the kind of people interviewed and by the kind of project deliverables studied. No business managers were available for interviewing and the project was not paying any attention yet on the implementation of the system in the business. Thus, the absence of “resource” rules can be considered an artifact of this case study.
- The number of “task knowledge” rules was high (130). This is probably due to the knowledge-intensive character of the assessment task in the offering process. In one of the domain documents the knowledge required for mortgage assessment was described in detail for each label.

Task-knowledge rules seem to be a special kind of business rules. These do not deal with the business at large, but are only concerned with the knowledge that is used in a single task. Therefore, task knowledge rules are not the type of things that are usually expressed in business rules during business modeling.

- It was sometimes difficult to decide whether a rule should be classified as a task knowledge rule or as another type of rule. Take for example, the rule “Applicants must be over 18 years old”. If this rule is seen as part of the knowledge that is used during the assessment of the applicant, it should be classified as a task knowledge rule. But, it could also be seen as a more general statement that only people over 18 years old can apply for a mortgage. In this case, the rule could also be seen as a pre-condition for starting the assessment. The decision on the rule type can in this case only be made on the basis of detailed information about the offering process.

Based on our experiences in the case study, we also formulated a number of guidelines for the elicitation of business rules.

**Guideline 1:** *Make sure that it is possible to speak to the right people.*

*Rationale:* Some types of business rules are hard to find without the right people to interview. Especially the rules in the managerial categories are often not known to all people in the organization. For rules concerned with the execution of tasks it is advisable to talk to both the people that carry out the task as well as their managers. It is possible that they have different views on the business rules governing a particular task.

**Guideline 2:** *Start with getting an overview of the business process as a whole and of the business rules of the “Goal & Value” type.*

*Rationale:* It is difficult to start the elicitation of business rules when one has no overview of what happens in the organization and why this is done. Especially when interviews are used to elicit business rules, it is best to have a business process model at hand. The process model acts as a structuring mechanism for business-rule elicitation. A UML activity diagram is an adequate technique for representing the process model. The diagram does not have to be fully completed yet. The rules about the goals and values of processes can be used as starting point for the elicitation of more specific rules about the goals of tasks.

**Guideline 3:** *Establish the objects structure rules of the domain at an early stage.*

*Rationale:* The rules concerning object structure define the domain vocabulary. It is important that the terms used for the objects in the domain are decided upon early on in the elicitation process. This way, there is less risk of terminological confusion during elicitation.

**Guideline 4:** *If there are two different ways to model a part of the business, choose the one that describes the business in the terms the person being interviewed understands best.*

*Rationale:* This situation occurred a number of times during the case study when we had to decide whether a business rule was of the information/control-flow type or of the pre/post-condition type. Both types of rules can be used to model inter-task dependencies. Most people from the business find it easier to use the flow-type rules. Therefore, during an interview it is easier to talk about flows between tasks than try to define pre- and post-conditions of tasks. Later on, the statements from the interviewee can be transformed into a pre/post-condition format, if deemed more appropriate.

The guideline we used in the final classification was to opt for a “pre/post-condition”-type only if the rule could be stated in terms of a single task. If a rule contained a statement about more than one task, it was classified as a flow-type rule.

**Guideline 5:** *The task knowledge category can be used to classify those rules that specify how a task must be performed, although a business model should not address information about the internal execution of a task.*

*Rationale:* The task-knowledge rules may seem too detailed to be included in a business model, but are useful to record anyway, Task-knowledge rules typically act as a store of information, which is used at some later point in the project. For example, we found many rules about the way the assessment task needed to be performed. Such rules provide valuable input for a follow-up project to develop a knowledge system for carrying out the assessment task.

## 4. Representing Business Rules

In principle, all business rules can be represented in free-format natural language. However, as was already pointed out in the introduction, this is not a effective representation for all types of business rules. For example, it seems rather cumbersome to represent object-structure rules in a textual format while appropriate diagramming techniques exist. From our experiences in the case study, we have formulated a number of representation guidelines for business rules based on the following assumptions:

- Given the widespread adoption of UML, we prefer a UML representation whenever this is suitable. Textual formats are only introduced if a UML representation is cumbersome.
- If a textual format is required, we have tried to formulate a “rule template”, i.e. a prototype sentence structure in which, from our experience, most or all of the business rules of a particular type can be phrased.

Templates support rule elicitation by showing the type of information that is needed. Also, using business-rule templates ensures that business rules of a certain type have (on the whole) the same structure.

- Only if no template were found, free-text format is recommended.

Table 3 provides an overview of the preferred representations of the proposed rule categories.

Category	Rule Type	Representation
Structural	Concept Structure	<i>UML class diagram.</i> Use the stereotype <<concept>> to indicate that domain concepts are being represented. Because constraints on associations may be quite complex an alternative is to use free format textual rules linked to one or more associations.
	Persistency	<i>UML stereotype</i> When more information about the persistency has to be specified the following templates could be used to specify a textual business rule as a property of a class or attribute or as a <<business rule>> annotation linked to a class or attribute. -[Information type/attribute] is valid for [period] -[Information type/attribute] must be destroyed after [period or task]
	History	<i>Boolean property or a stereotype of classes and attributes.</i> When more information about the history has to be specified the following template could be used to specify a textual business rule as a property of a class or attribute or as a <<business rule>> annotation linked to a class or attribute. <i>All changes in the value of [information type or attribute] must be recorded</i>
Behavioral	Information Flow	<i>UML activity diagram with object flows</i> Data-flow diagram is the obvious alternative, although not included in UML
	Control Flow	<i>UML activity diagram or state diagram</i> Textual alternatives: -[Task X] can only be performed when [task Y] is completed -[Task] must be performed until / while [event] -If [event], perform [task] -If [task], then [event] or pseudo-code
	Pre- & Post-conditions	The following templates can be used to specify a textual business rule as a property of an activity/action or as a <<business rule>> annotation linked to an activity/action. -[Task] must only be performed when [condition] -After the execution of [task], [condition] must be true
	Frequency	<i>UML interaction diagram</i> In addition, the following templates may be useful: -It is expected that [task] will be performed X times every [time measure] -[Task] will be performed by X [human actors]
	Duration	The following template could be used to specify a textual business rule as a property of an activity/action or as a <<business rule>> annotation linked to an activity/action. <i>[Task] should be performed within [time constraint]</i>
	Task Knowledge	<i>Free format, linked to task.</i> Knowledge-engineering methodologies such as CommonKADS [7] and ARC [10] provide methods for this type of rule modeling. This is however outside the scope of business-rule modeling. The guideline is to group these rules by task, and refrain (at this stage) from any detailed rule modeling.
Managerial	Organization	<i>Free format.</i> These business rules are diverse by nature, and it is thus not possible to define a general template.
	Goal & Value	<i>Free format, linked to task or process</i> This kind of information can be expressed in many different ways. No template is given for this category either. Business rules of this type should be linked to the task or activity that the rules describe.
	Actor Competences	-[Actor] that performs [task] must have [skill or knowledge] The actor-competence rules usually state what a knowledge or skills an actor needs to possess to perform a certain task.
	Actor Responsibilities	-[If [condition], [task] must be performed by [actor] Actor responsibilities state who is responsible for the correct execution of the task. Such a rule can for example describe that under normal conditions an information system should perform a task. If a certain condition changes, a different rule could say that a human actor should perform the task.
	Resources	<i>Free format, linking a resource to a task.</i> Resource type business rules are also hard to catch in a general template.

**Table 3: Preferred representation of business rules**

## 5. Discussion

The case study in this article is only a first step in ascertaining the added value of the classification scheme proposed. Nevertheless, it can be said that the classification scheme appears to provide reasonably intuitive rule categories. This is also due to the fact that it is based on a well-established business-modeling framework for knowledge-intensive organizations. Together with the process and representation guidelines, the classification scheme may form a core element for a full-fledged business-rule methodology. Such a methodology is a prerequisite for successful application of the business-rule concept. In this section we discuss a number of potential extensions that would fit into such a methodology.

### Business-rule representation

With respect to business-rule representation we have limited ourselves in this article to informal (free text, templates) and semi-formal (UML) notations. A business-rule methodology should also provide formal rule notations, that allow consistency and completeness checking as well as some (limited) simulation. The OCL language [8] used for formal UML constraints is a candidate, but its syntax is rather cumbersome. Alternatives are knowledge-representation languages such as KIF [3]. It should be stressed, however, that formal rule languages can never replace the informal/semi-formal notations, as the latter ones are the ones through which communication with the user/supplier needs to take place.

Rules have an intrinsic metalevel flavor and are therefore cumbersome to represent with the standard UML primitives. Specialized rule modeling techniques can be found in the work of Wilson [10] and in the CommonKADS knowledge-engineering methods [7].

### Business rule in perspective

There is still a way to go to a full-scale business-rule methodology. Process support in the form of life-cycle models and elicitation tools is needed. An area that also deserves attention is the validation of rule validation. There is a link here with enterprise modeling, and techniques such as used in the TOVE project [ref] could be useful here.

There is a clear link between business rules and the fields of knowledge engineering. Knowledge engineering provides techniques for complex rule modeling [7]. If business rules need to be encoded explicitly into a target system, a system-development process similar to that of knowledge systems is required.

Does the notion of “business rule” add anything to our modeling repertoire? Based on our case study we would say “yes”. The main added value is the focus on a *declarative* specification of the business. It forces one to concentrate on the essentials, namely the required competence and limitations of a part of the business. Therefore, we see business rules mainly as a modeling paradigm. The actual representation is often best represented in a non-rule format.

### Acknowledgements

We are grateful for the comments from Ellen Gottesdiener and Kirk Wilson on prior versions of this article. We would also like to thank Wim van der Velden, Wilfred Bosch and Tamara van der Weij for their input in the case study.

## 6. References

- [1] Business Rules Group (2001). WWW: <http://www.businessrulesgroup.org/brgdefn.htm>.

- [2] Date, C. J. (2000) *What Not How: The Business Rules Approach to Application Development*. Reading, MA, Addison Wesley.
- [3] Genesereth, M. R. and Fikes, R., editors (1992). *Knowledge Interchange Format*. Version 3.0. Reference Manual. Stanford University, Computer Science Department Report Logic-92-1, June.
- [4] Hay, D., Kolber, A., & Anderson Healy, K. (1997). Guide Business Rule Project: Final Report.
- [5] Martin, J. (1997). *Object/Oriented Methods – A Foundation. UML edition*. Upper Saddle River, NJ, Prentice Hall.
- [6] Ross, R. G. (1997). *The Business Rule Book*. Business Rule Solutions, second edition. ISBN: 0941049035
- [7] Schreiber, A. Th., Akkemans, J. M., Anjewierden, A. A., de Hoog, R., Shadbolt, N. R., Van de Velde, W., Wielinga, B. J. (2000). *Knowledge Engineering and Management: The CommonKADS Methodology*. Cambridge, MA, MIT Press.
- [8] Warner, J. and Klepke, A. G. (1999). *The Object Constraint Language: Precise Modeling With UML*. Addison Wesley.
- [9] Weiden, M. (2000). A Critique of the Business-Rule Approach. M.Sc. Thesis, University of Amsterdam, Department of Social Science Informatics. URL:  
<http://www.everest.nl/library/onderzoek/downloads/BRapproach.pdf>.
- [10] Wilson, K. The ARC Methodology.