# Towards Formal Expression of Business Rules Written in Polish

Bogumiła Hnatkowska and Anna Błasiak

Wroclaw University of Technology, Institute of Informatics, Wyb. Wyspianskiego 27, 50-370 Wrocław, Poland
Bogumila.Hnatkowska@pwr.wroc.pl, 164749@student.pwr.wroc.pl

**Abstract.** Business rules are very important part of a business model. They should be defined in an unambiguous, understandable way, especially when they are going to be implemented in a software system. There are some attempts addressing that issue but none of them considers business rules written in Polish. The paper presents existing methods and tools that support business rules written in English, and the roadmap to adapt them to Polish language.

**Keywords:** Business rules, business rules formalisms, MDA, CIM, SBVR, Polish.

## 1 Introduction

Business rules are very important part of a business model, and next important part of requirements specification for software system. However, there is a big gap between these two worlds, i.e. a specific domain which is known to domain experts, and the IT domain in which business knowledge first must be correctly understood, interpreted, and represented in the manner that fits the needs of software development.

The language in which business rules are expressed must be proper for their audience. At the business level business rules are expressed in natural language, typically in declarative way [1,2]. Natural language brings obvious benefits – it is understandable for all interested parties – but also commonly known problems: misinterpretation, incomprehension etc.

In the real enterprise there are plenty of business rules defined. In such a case completeness and consistency checking of business rules specification is challenging task. The natural solution of the above mentioned problem is to apply tools that will not only help in gathering and checking business rules but also offer support to software development. Model Driven Architecture (MDA) is an approach to software development which seems to be ideal for that [4].

The aim of the paper is to present the current state of art about tools and methods that are suitable to: (a) represent business rules in natural language, and (b) translate natural language specification into formalisms that are suitable for further development. This is the first stage of the research which is going to propose such solution for Polish language. The existing solutions support mainly English which is far less flexible, and has less complex grammar than Polish. According to [31] “the Polish

M. Bajec and J. Eder (Eds.): CAiSE 2012 Workshops, LNBIP 112, pp. 478–485, 2012.

language is one of the most difficult languages to learn with its tongue-bending pronunciation, complex gender system, seven cases, 'aspect' as a grammatical category of the verb and a tendency to avoid internationalisms for 'real' Polish words".

The paper is organized as follows. Section 2. provides the definition of basic notions. Section 3. explains the role of business rules and shortly presents languages used for business rules expression in the context of software development with MDA approach. Next, Section 4. concentrates only on methods allowing to express business rules in natural languages and presents solutions and standards in this area. Section 5. gives an overview of tools supporting business rules processing at different levels. In this section also some useful tools for Polish are presented. Last Section 6 presents the roadmap of future work.

## 2 Business Rules Overview

Business model defines basic notions from a given domain, the relationships between the notions and the way in which they are constrained. 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 [5]. According to [1,2] business rules should be expressed in *natural language*, in *declarative way*.

A business rules can belong to a specific type, however, there are no commonly accepted, one business rules classification. But most of works (e.g. [1–3, 5]) introduce following categories of business rules:

- *terms* – a definition of something,
- *facts* – a relationship between terms,
- *constraint rules* – specify policies or conditions that restrict object structure and behavior; they may always apply (invariants) or apply only under certain conditions:
  - *operation constraint rules* – typically expressed as pre and/or post conditions,
  - *structure constraint rules* – specify polices about attributes' domain, class instance numbers etc.
- *derivation rules* – specify policies or conditions for inferring or computing facts from other facts:
  - *inference rules* – specify that if certain facts are true, a conclusion can be inferred,
  - *computation rules* – derive their results by way of processing algorithms, a more sophisticated variant of inference rules,
- *stimulus and response rules* – any rule that tends to take some action automatically when a relevant event occurs. Such action might be to create or delete data, enable or disable another rule, set some value, execute some program or procedure.

## 3 Business Rules in the Context of MDA

Model Driven Architecture (MDA) is a standard proposed by OMG presenting a rigorous approach to software development in which models play a crucial role [4].

MDA introduces three types of models, presenting the system on different abstraction levels. Each subsequent model is a refined or extended version of the previous one. The models are as follows (see fig. 1):

- *Computation Independent Model* (CIM), which takes care of an environment in which the system will operate. This model mainly presents business model and it gathers business rules. It also includes requirement specification. There are some attempts to represent business rules in natural language at that level, e.g. RuleSpeak [6], Attempo Controlled English (ACE) [7,8], SBVR Structured English (SBVRSE) [9]. None of them is defined for Polish.
- *Platform Independent Model* (PIM), which presents the solution in the form that hides all platform specific details, and is so general that could be applied at any targeted platform. At that level business rules are expressed with the use of formal notations [30] (e.g. RuleML, Rule Interchange Format), and typically are spread among many UML diagrams [10]. Object Constraint Language (OCL) [11] is also used for their representation.
- *Platform Specific Model* (PSM), which presents the solution in terms specific to a selected platform. At that level business rules are typically implemented in the code, sometimes as a separate layer (e.g. [12]) or with the use of design patterns (e.g. [13]). Realization of business rules could be also delegated to special engines, e.g. JRules [14] or database servers, e.g. [15].

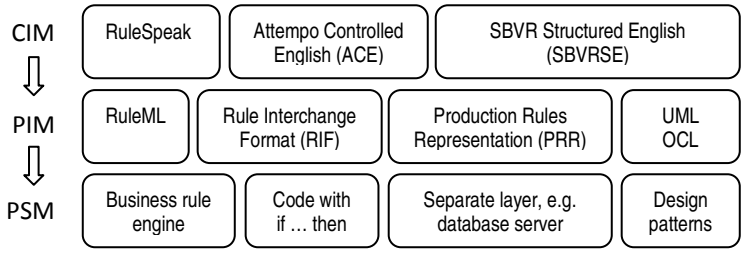


**Fig. 1.** Business rules' representation in MDA approach

The paper concentrates on business rules expressed at CIM level. The approaches used for that – pointed out at the fig. 1– are described in greater detail in subsequent section.

In the MDA transformations between models play important role. They are typically defined in terms of meta-models that provide abstract syntax for model representation. To enable CIM business rules to PIM business rules transformation the proper meta-models for them need to be either selected or elaborated from the beginning. For example, MOF is the meta-model for UML language, and Semantics of Business Vocabulary and Business Rules (SBVR) is the meta-model for SBVRSE.

## 4 Attempts to Express Business Rules in Natural Language

There are some attempts to expressing business rules in natural language, e.g. [6,7,9]. Most of them consider English as a native language of a domain expert. The attempts support the expert in different ways. Sometimes they take a form of a guideline that only help to write rules in more readable and clear manner. RuleSpeak belongs to that group [6]. Other attempts restrict the expert's language by defining the exact structure (grammar) of business rules. ACE [7,8] and SBVRSE [9] are the representatives of that group.

### 4.1 RuleSpeak

RuleSpeak introduces a concept of a sentence form which is defined as "a basic pattern or template in natural language that can be used to express a certain kind of business rule in a consistent, well-organized manner" [6]. It distinguishes two kinds of business rules (called here guided statements): business rule statements, and statement of advices. Business rule statements enable to express that something is required, disallowed or conditionally allowed. Statement of advice permits to express that something is allowed or not required.

The sentence form should be constructed in a specified manner with the use of so called rule keywords. For example, every business rule statement must use one of the following words: "must" or "only", while a statement of advice must use: "may" or "need not". Statements of advice can't be directly mapped to the taxonomy of business rules presented in Section 2.

At that moment RuleSpeak does not introduce the concept of glossary for the definition of terms used in guidance statements.

### 4.2 ACE

Attempto Controlled English (ACE) is a controlled language elaborated at University of Zurich [7,8]. ACE enables to express all types of business rules, starting from constrains to stimulus/response rules.

ACE is a formal language in the sense, that texts written in it "are computer-processable and can be unambiguously translated into discourse representation structures (DRS), a syntactic variant of first logic" [8]. DRS representation can be further translated into other languages, and RuleML is among them.

The grammar of ACE defines the syntax of ACE texts. ACE's syntax is expressed as a set of construction rules. The meaning of syntactically correct sentences is described as a set of interpretation rules. The grammar rules are build on a vocabulary which consists of predefined function words (etc. articles, conjunctions), predefined fixed phrases (e.g. 'there is') and user-defined content words.

Unfortunately, the vocabulary, and grammar rules are defined only for English, and it is doubtfully they can be easy adopted for another language, especially as morphologically reach as Polish. However, some ideas about acceptable sentence structures, and potential problems with their interpretation can be get from ACE experiences.

There are also some useful tools available for ACE. Attempto Parsing Engine (APE) is one of the most important. It checks the correctness of ACE texts, and – when they are correct – is able to generate syntax tree or DRS.

### 4.3 SBVRSE and SBVR

Semantics of Business Vocabulary and Business Rules (SBVR) is an OMG standard, providing a meta-model for documenting the semantics of business rules and vocabulary [9]. Its main goal is to enable defining business elements in friendly manner, preferably in natural language. It facilitates automation of business rules in software systems by structuring them as logical semantics formulation. SBVR standard also defines CMOF meta-model and XMI schema for interchange of business vocabulary and rules between software tools.

The standard document distinguishes two kind of business rules: structural and operative ones. Structural business rules "are rules about how the business chooses to organize (i.e., 'structure') the things it deals with" [9]. In turn, operative business rules "are rules that govern the conduct of business activity. In contrast to structural rules, operative rules are ones that can be directly violated by people involved in the affairs of the business" [9]. SBVR specifies also other elements of guidance such us advices or business policy.

Business rules in SBVR are always build on fact types, and the fact types are build on concepts expressed by terms. Fact type is an equivalent to the fact and concept is an equivalent to term in taxonomy presented in Section 2.

The major downsides of this meta-model are: its significant complexity [30], and the lack of support for standard arithmetic beyond numeric comparison [32].

It should be emphasized that SBVR distinguishes the *meaning* of rules and vocabulary from their representation (*expression*) thereby enabling representation of the same rule, term or fact in different languages. The standard itself is described with the use of not-normative English notation called SBVR Structured English (SBVRSE), which is a limited subset of English language. SBVRSE is only one of possible notations for SBVR representation. Another potential notation could be SVBR Structured Polish.

## 5 Business Rules Support in the Context of MDA

The planned final result of the work is to elaborate formal representation of business rules written in (controlled) Polish. This is the first step to enable models verification, and CIM-PIM (or PIM-CIM) model transformations. The fig. 2 demonstrates the scope of interests (marked in gray). It is assumed that transformations and verifications are done automatically. Because SBVR is native language independent standard it is considered as the first choice option for formal representation of business rules.

There are few editors of SBVR rules, e.g. SBeaVeR [17], VeTIS [18], RuleXpress [19], KnowEnterprise [20]. The first one allows to create, validate and verify the business rules and vocabulary written in SBVRSE, but it does not generate any formal

representation or PIM model. Similar functionality is offered by two commercial tools: KnowEnterprise and RuleXpress, which additionally translate business vocabulary into SBVR standard. Noteworthy is another open source tool – VeTIS, which not only covers SBeaVeR functionality but also enables a user to generate UML class diagrams from business rules and vocabulary.

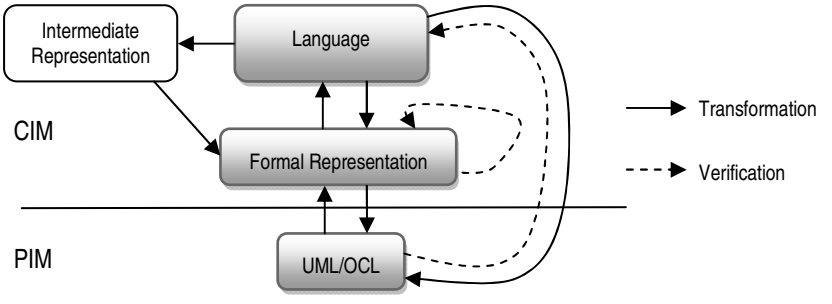


**Fig. 2.** Existing and planned areas of tool support in the context of business rules

SBVR standard was also used in NL2OCLviaSBVR tool [21], which uses it as an intermediate step in transformation from business rules written in English to OCL constraints. The tool does not provide any editor and assumes that vocabulary terms and facts are defined not in controlled English but in UML class diagram.

There are also some tools doing PIM-CIM transformations, e.g. UML2SBVR [22]. The tool takes an XMI file (representing UML class diagram) and generates another XMI consistent with SBVR standard.

Finally, Eclipse community has set up a SBVR/MDT project [23] to provide SBVR meta-model implementation and sample tools to support developers working with SBVR standard. Currently they managed to create ecore file based on CMOF meta-model and generate classes from it to enable loading and saving XMI files compliant with SBVR exchange document format.

In the context of the subject of this work (formal expression of business rules written in Polish), the problem of texts parsing and translating them into SBVR instances is the most challenging task. The problem is solved in different manners for SBVR Structured English. For example, the approach presented in [24] uses AI methods to parse rules written in SBVRSE and represent them as a model conforming to Syntax meta-model. This model is then transformed to model instantiating SBVR meta-model with usage of ATL tool. In the second approach [25] authors try to represent business rules in the form similar to semantic formulations exemplified in SBVR specification. The output of their parser can be easily transformed to XMI format. Finally, VeTIS developers have directly implemented some parsing rules in order to recognize SBVR elements in files containing vocabulary and business rules and then to represent them as objects – instances of SBVR classes. The second and third approach use deterministic algorithms that could cause some mistakes while parsing SBVRSE, because – as authors of the first approach pointed out in [24] – this notation makes it possible to interpret one sentence in different ways. That is why they search for a solution using advanced AI techniques.

There are also some tools supporting processing of Polish language. Morpheus [26] is one of the most useful offering morphological analysis of Polish sentences. For a given word it provides: all basic forms of the word, part of speech, grammar category values assigned to part of speech (e.g. gender, case etc.). It uses grammatical vocabulary of Polish language [27], which contains complete grammar characteristics of about 245 000 Polish words with their inflection forms (over 4 millions).

Another useful tool is TaKIPI [28] – a Polish language tagger, which determines for a word in some text its morpho-syntactical description and interprets its meaning depending on context. TaKIPI is also integrated with another tool Guesser [33], which performs morpho-syntactical analysis on unfamiliar words.

At the moment there is also a Polish version of WordNet – plWordNet (pl. Słowosieć) [29].

## 6 Summary

The paper deals with the problem of formal representation of business rules written in natural language. The state of art of existing approaches and tools was broadly presented. At that moment there are no solutions that give support for Polish language.

As a part of further work it is planned to adapt SBVR for Polish language, and to elaborate SBVR Structured Polish (SBVRSP) – the equivalent of SBVRSE – together with supporting tools. Because SBVR standard introduces many types of element of guidance, first structural business rules will be taken into consideration. Realization of that task requires elaboration of grammar and style rules for selected subset of Polish language (RuleSpeak for Polish). The grammar and style rules should be verified whether their application provides unambiguous, and understandable business rule definitions. Having all above mentioned steps done it will be possible to develop SBVRSP editor. The next and most complicated step is a parser development. The parser should translate business rules expressed in Polish into SBVR and maybe other formal notations.

SBVR standard was selected because it allows to separate business rules representation (Polish language) from their meaning. As a result existing tools, useful in the MDA approach, that take SBVR compliant input can be used without further modifications.

## References

1. Ross, R.G.: Principles of the Business Rule Approach. Addison Wesley (2003)
2. Morgan, T.: Business Rules and Information Systems – Aligning IT with Business Goals. Adisson-Wesley (2002)
3. von Halle, B.: Business Rules Applied – Building Better Systems Using the Business Rules Approach. Wiley (2002)
4. MDA Guide v. 1.0.1., http://www.omg.org/mda
5. Business Rules Group: Defining Business Rules – What are they really?, http://www.businessrulesgroup.org/first_paper/br01c1.html
6. Ross, R.G.: RuleSpeak® Sentence Forms: Specifying Natural-Language Business Rules in English. Business Rules Journal 10(4) (2009)
7. Attempto Project, http://attempto.ifi.uzh.ch/site/

8. Fuchs, N.E., Kaljurand, K., Kuhn, T.: Attempto Controlled English for Knowledge Representation (2008), http://attempto.ifi.uzh.ch/site/
9. Semantics of Business vocabulary and Rules (SBVR), OMG Standard, v. 1.0, (2008), http://www.omg.org/spec/SBVR/1.0/
10. OMG Unified Modeling Language (OMG UML), Superstructure Version 2.3, (2010), http://www.omg.org/spec/
11. OCL 2.0 OMG Final Adopted Specification, http://www.omg.org/docs/ptc/05-06-06.pdf
12. Hnatkowska, B., Kasprzyk, K.: Business rules modularization with AOP. Przegląd Elektrotechniczny R 86(9), 234–238 (2010)
13. Arsanjani, A.: Rule Object 2001: A Pattern Language for Adaptive and Scalable Business Rule Design and Construction. In: 39th International Conference and Exhibition on Technology of Object-Oriented Languages and Systems (2001)
14. WebSphere ILOG JRules RBMS, http://www-01.ibm.com/software/integration/business-rule-management/jrules-family/
15. Koppelaars, T.: RuleGen Quickstart Guide. RuleGen BV (2009)
16. KDM Analytics, Semantics of Business Vocabulary and Business Rules (SBVR), http://www.kdmanalytics.com/sbvr/sbvr_intro_3.html
17. http://sbeaver.sourceforge.net/
18. Kaunas University of Technology Department of Information Systems, Creating UML&OCL Models from SBVR Business Vocabularies and Business Rules. VeTIS User Guide. Kaunas (2009), http://www.magicdraw.com/files/manuals/VeTISUserGuide.pdf
19. RuleArts, LLC, RuleXpress: The business tool for expressing and communicating business rule, http://www.rulexpress.com/
20. KnowGravity Inc.: Comprehensive support for the business perspective of Model Driven Enterprise Engineering, http://www.knowgravity.ch/eng/pdf-e/KnowEnterprise-BU%20E.pdf
21. Bajwa, I.S., Lee, M.G., Bordbar, B.: SBVR Business Rules Generation from Natural Language Specification. In: AAAI Spring Symposium 2011 – Artificial Intelligence 4 Business Agility. AAAI, San Francisco (2011)
22. Transforming UML Conceptual Schemas to SBVR: Tool description, http://jordicabot.com/research/SBVR/index.html
23. SBVR/MDT, http://wiki.eclipse.org/MDT-SBVR
24. Kleiner, M., Albert, P., Bézivin, J.: Parsing SBVR-Based Controlled Languages. In: Schürr, A., Selic, B. (eds.) MODELS 2009. LNCS, vol. 5795, pp. 122–136. Springer, Heidelberg (2009)
25. Marinos, A., Moschoyiannis, S., Krause, P.: An SBVR to SQL Compiler. RuleML Challenge (2010)
26. Morpheus morphological analyzer, http://sgjp.pl/morfeusz/morfeusz.html
27. Saloni, Z., Grzuszczyński, W., Woliński, M., Wołosz, R.: Słownik gramatyczny języka polskiego. Wydanie I, Wiedza Powszechna, Warszawa (2007), http://sgjp.pl/
28. TaKIPI Tagger for Polish language, http://nlp.pwr.wroc.pl/takipi/
29. PlWordNet version 1.6, http://plwordnet.pwr.wroc.pl/wordnet/
30. Linehan, M.H.: SBVR Use Cases. Information Systems Journal, 182–196 (2008)
31. http://www.bbc.co.uk/languages/polish/soap/background.shtml
32. Linehan, M.H.: Ontologies and Rules in Business Models Rules in SBVR Vocabularies in SBVR. Constraints, 149–156 (2008)
33. Piasecki, M., Radziszewski, A.: Polish Morphological Guesser Based on a Statistical *A Tergo* Index. In: Proceedings of the International Multiconference on Computer Science and Information Technology, AAIA 2007, pp. 247–256 (2007)