

# RuleCNL: A Controlled Natural Language for Business Rule Specifications

Paul Brilliant Feuto Njonko<sup>1</sup>, Sylviane Cardey<sup>1</sup>, Peter Greenfield<sup>1</sup>, and Walid El Abed<sup>2</sup>

<sup>1</sup> Centre Tesnière - Équipe d'Accueil 2283  
Université de Franche-Comté - UFR SLHS  
30, rue Mégevand, 25030 Besançon Cedex, France  
{paul.feuto\_njonko, sylviane.cardey, peter.greenfield}@univ-fcomte.fr  
<sup>2</sup> Global Data Excellence Ltd.  
Geneva, Switzerland  
walid.elabed@globaldataexcellence.com

**Abstract.** Business rules represent the primary means by which companies define their business, perform their actions in order to reach their objectives. Thus, they need to be expressed unambiguously to avoid inconsistencies between business stakeholders and formally in order to be machine-processed. A promising solution is the use of a controlled natural language (CNL) which is a good mediator between natural and formal languages. This paper presents RuleCNL, which is a CNL for defining business rules. Its core feature is the alignment of the business rule definition with the business vocabulary which ensures traceability and consistency with the business domain. The RuleCNL tool provides editors that assist end-users in the writing process and automatic mappings into the Semantics of Business Vocabulary and Business Rules (SBVR) standard. SBVR is grounded in first order logic and includes constructs called semantic formulations that structure the meaning of rules.

**Keywords:** Business Rule, Controlled Natural Language, Automatic Mapping, Semantics of Business Vocabulary and Business Rules

## 1 Introduction

Nowadays, companies are facing much pressure due to competition and growth, which requires frequent adaptation of their business rules. However, for a couple of decades, business rules have been hard-coded in automated business processes, information systems and often inconsistently so. Thus, changing or modifying business rules inevitably requires software engineers' intervention because they are inaccessible to business experts (e.g. healthcare experts, finance experts, etc.) who understand the actual problem domain and are responsible for finding solutions. As a result, companies cannot keep pace with the changing business environment.

The business rule approach (BRA) has evolved over the years in order to solve the deficiency described above [5] [1]. It claims that all business rules should be

collected and explicitly represented in a centralized application called business rule management system (BRMS). Many formal business rule languages have been devised allowing companies to define their business rules explicitly and formally. However, since such rules have to be created and/or verified by domain experts who are mostly not familiar with formal notations, a promising solution is the use of a controlled natural language (CNL) that can serve as a front-end interface and provide automatic mappings into formal notations. Thus, this paper presents RuleCNL for expressing business rules. Its core feature is the alignment of the business rule definition with the business vocabulary which ensures traceability and consistency with the business domain. The underlying natural language (NL) in this paper is English but RuleCNL also works with French. The RuleCNL tool provides editors that assist end-users in the writing process and provides automatic mappings into the Semantics of Business Vocabulary and Business Rules (SBVR) standard. SBVR is grounded in first order logic and includes constructs called semantic formulations that structure the meaning of rules.

The rest of the paper is structured as follows: In Section 2, we introduce the notion of business rules and CNLs. Section 3 presents some related work on CNLs for business rules and in Section 4, we describe the RuleCNL in detail. Section 5 presents the RuleCNL tool and Section 6 the conclusions and future work.

## 2 Business Rules and Controlled Natural Languages

### 2.1 Business Rules

In the literature, we find numerous definitions of business rules. However the most used definition is given by the Business Rule Group (BRG) [2] as follows:

*“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.”*

**E.g.** *A loan must be approved if its value is less than 10,000 Euros.*

One challenge with the BRA is to find the characteristics of a good business rule. Some workers [3] [4] have proposed a set of characteristics for a business rule statement to be deemed as good. Among them, we can cite that a business rule should be atomic, declarative, business related, consistent, unambiguous, etc.

In the context of the BRA [1] [5], many formal languages have been proposed by many vendors for business rules modeling. These languages have a well-defined syntax, an unambiguous semantics and support automated reasoning over rules. [6] provides a state of the art on business rule languages and concludes that most of them are hard to use for business people without training in formal methods, but are rather easy for software engineers. We contend that business rules should be expressed declaratively in NL sentences for the business audience. Thus, CNLs are good solutions for bridging the gap between natural and formal languages [7].

## 2.2 Controlled Natural Languages

CNLs are engineered subsets of natural languages whose grammars and vocabularies have been restricted in a systematic way in order to reduce both the ambiguity and complexity of full NLS (e.g. English, French, etc.) [8].

In general, CNLs fall into two broad categories: human-oriented CNLs and machine-oriented CNLs. Human-oriented CNLs are intended to improve the communication among people for specific purposes and the readability and comprehensibility of technical documentations. They have no formal semantics and are usually defined by informal guidelines [9]. Machine-oriented CNLs are designed to improve the communication between humans and computers. They are completely unambiguous and can be defined by formal grammars with a direct mapping to formal logic [9].

Machine-oriented CNLs can be used in various domains and applications such as CNL for knowledge representation (Attempto Controlled English [10], Processable English [11], Computer Processable Language [12], etc.), CNL for ontologies [13], CNL for semantic web [14], CNL for machine translation [15], CNL for business rules like RuleCNL as presented in this paper, etc.

## 3 Related Work

The idea of verbalizing rules that already exist in a formal representation [16] [17] has led the domain of business rules to become an interesting application of CNLs. Because business rules need to be approved and followed by people with no particular background in formal or logic representation, it is important to have an intuitive representation that CNLs can offer. There are some CNLs that have been defined for this particular problem area.

In the business context, the Object Management Group (OMG)<sup>3</sup> has published a standard called SBVR [20] which provides a means for describing the structure of the meaning of rules expressed in the natural language that business people use. However, SBVR is not itself a CNL so it is up to each SBVR-implementing language or notation to specify its formal mechanisms [19]. SBVR claims to be restricted to semantics leaving apart a key functionality of NLS, which is syntax. Thus, for various reasons, the SBVR standard did not include a normative specification of the language to be used by business people to express their vocabulary and rules.

SBVR-Structured-English (SE) [20] and RuleSpeak [21] are both defined in the SBVR specification as CNLs to express business rules in a restricted version of English. However, these CNLs are not languages per se, but rather a set of best practices for human speakers. They are defined informally by sets of guidelines based on experiences of best practice in rule systems [9]. They are not normative and have no formal grammar but can be mapped to the semantics formulation of the SBVR meta-model. They are not supported by any tooling and cannot be processed in a fully automatic way [18]. The syntax is achieved by

<sup>3</sup> <http://www.omg.org/>

text formatting and coloring, which could be used to aid understanding by the domain expert user. However, a CNL requires a formal definition of its syntax (the language's grammar), which can be used to support business users in the process of entering syntactically correct inputs. This limitation is avoided by our RuleCNL controlled natural language.

## 4 RuleCNL: A Controlled Natural Language for Business Rules Specifications

### 4.1 Introduction

In this section, we present our RuleCNL for expressing business rules. As mentioned in the introduction, its methodology is based on the alignment of the business rule definitions with the business vocabulary. Thus, business rules are semantically connected to the business domain and readily understandable by domain experts. The methodology is derived from the core idea of the BRA advocated by the BRG as follows: *"Rules build on facts, and facts build on concepts as expressed by terms."*[3]. In order to overcome the limitations highlighted in the previous section, we defined a formal grammar and therefore a parser that can be used for the syntax analysis of rules. The writing process of a business rule is fully supported by the consistency check imposed by the methodology. Its semantics is defined by automatic mappings into the SBVR semantic formulations. This enables a language-independent way of describing the semantic structure of rule statements and is grounded on a sound theoretical foundation of formal logic. Fig. 1 shows the general architecture of the RuleCNL.

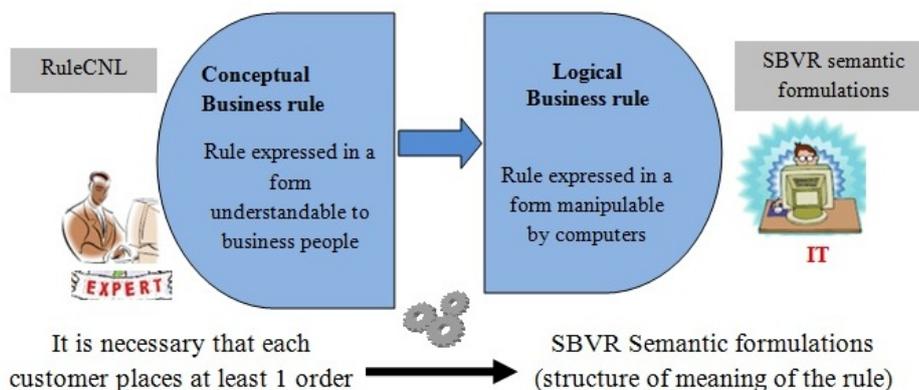


Fig. 1. General architecture of the RuleCNL

For the sake of readability, the SBVR semantic formulations of the example of the Fig. 1 is shown at the Fig. 2

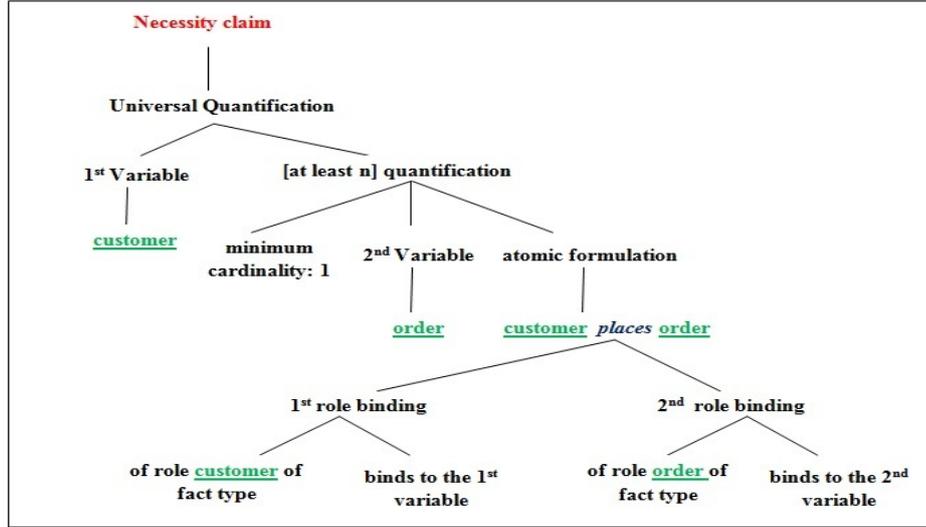


Fig. 2. SBVR semantic formulations of the rule shown in the Fig. 1

## 4.2 RuleCNL Vocabulary

The RuleCNL vocabulary represents the conceptual model of the business domain which defines a cohesive set of interconnected concepts (domain terms and their relations) that a given company uses in its talking or writing in the course of doing business. It is defined in a structured way by a business user and represents the knowledge that the company knows about itself. The RuleCNL is domain-independent and the vocabulary consists of:

- **Domain Term:** designates a significant business entity that can be represented by a common noun or a noun phrase. (e.g. *customer*, *gold customer*, *bank account*, etc.). A *Domain Term* is always represented in a singular form and with no articles or determiners.
- **Domain Name:** designates a significant business entity that represents only one thing. It is usually a proper name. (e.g. *France*, *Euro*, *USA*, etc.)
- **Domain Verb:** designates a relationship, situation, or action involving one or two *Domain Terms/Names*. In order to keep the RuleCNL vocabulary simple and readily manipulable by domain experts, we only consider unary and binary *Domain Verbs*.

The binary *Domain Verb* defines a semantic relationship and has two placeholders filled by *Domain Terms/Names* and its declaration syntax is *Subject + Domain Verb + Object*. The *Domain Verb* per se is only a part of this declaration syntax and has no meaning in isolation, but only within the relationship.

For instance: Let us consider the verb *to run*, it has different meanings within the following relationships:

manager *runs* company; horse *runs* race; computer *runs* program

A *Domain Verb* can be written both in active or passive form. (e.g. *customer places order*; *order is placed by customer*). The *Domain Verb* can be a linguistic verb (in this case, it is conjugated in the third-person singular) or a combination of a verb with some functional words (preposition, etc.).

The unary *Domain Verb* defines a characteristic or a state of a *Domain Term/Name* and its declaration syntax is *Subject + Domain Verb*. Its evaluation leads to a Boolean value. (e.g. *order shipped*, *customer smokes*)

There are no additional words or functional words in the relationship. This leads to a great flexibility and any constraints or restrictions will be added when defining business rules. The RuleCNL vocabulary includes some built-in relationships as comparison verbs (equality/inequality) that are not defined by domain users. Domain users define or import their vocabulary with the help of the vocabulary editor.

### 4.3 RuleCNL Grammar

RuleCNL grammar defines syntax rules and constrains for business rules. It assumes the existence of RuleCNL vocabulary and makes reference to *Domain Terms/Names* and *Domain Verbs* defined in the vocabulary. The general structure of a rule is: *Modality + Statement*. The *Modality* carries the sense of operational or structural rules. (e.g. *It is obligatory that*, *It is necessary that*, etc.). The *Statement* is a declarative sentence that regulates the structure of the rule. Its general pattern is a set of clauses which can go from a simple to a very complex/compound (linked by connectives and conditionals) structure. Each clause is always based on exactly one *Domain Verb* defined in the vocabulary. Subsequently, it combines many linguistic particles (function words, adverbs, etc.) in order to form a grammatically correct sentence. The grammar supports the use of complex noun phrases involving quantifications, instantiations and qualifications of *Domain Terms*. It also supports verb phrases involving verbs and prepositional phrases and can be used to define simple, compound and conditional sentence structures. The following examples show some statement structures. In these examples, *Domain Terms* are underlined; *Domain Verbs* are in italic font and other linguistic particles are in bold font.

**Simple statement:** It is based on only one clause.

**E.g. each** customer *places* **at least one** order

This statement follows the structure: *Subject + BinaryVerb + Object* and is based on the clause (*Domain Verb*) customer *places* order. *Subject* and *Object* are noun phrases with determiners which in this case are quantifiers (each, at least). In general, a determiner can also map to an article (a, an, the) or nothing. *Subject* and *Object* make reference to *Domain Terms* visible in the vocabulary. The *BinaryVerb* also makes reference to a *Domain Verb* of the vocabulary. In this example, *Subject*, *Object* and *Domain Verb* are unqualified, and then the rule will be applied to any instance of the related *Domain Terms*. However, they can be qualified by other descriptive elements,

such as their existence in a particular state, in order to specify the applicability of the rule with enough precision.

**Compound/complex statement:** this is recursively built from simpler statements through coordinators (and, or) and subordinators (if, who, that, which).

**E.g. each order is shipped if the customer who places the order is adult and holds an account that has a outstanding balance that is greater than 0**

This statement is based on the following *Domain Verbs*:

order shipped;

customer adult;

customer places order;

customer holds account;

account has outstanding balance

defined in the vocabulary

and the built-in comparison verb quantity1 is greater than quantity2

#### 4.4 Formalization of the Grammar

In order to build a parser for the RuleCNL, we have formalized our grammar rules using Extended Backus-Naur Form (EBNF) which is a notation for specifying context free grammars. The RuleCNL grammar consists of a set of rewriting rules used to restrict the syntax of rule statements. An excerpt of the general rule pattern is shown at Fig. 3. The vertical bar (|) is the disjunction and the comma character ( , ) is the conjunction. The symbol ( )+ means that at least one occurrence of rule element enclosed in the bracket must appear at that point.

```
Statement = (Base_Statement | Implication_Statement) ;
Implication_Statement = Consequent, "if", Antecedent ;
Consequent = Base_Statement;
Antecedent = Base_Statement;
Base_Statement = (Simple_Statement | Compound_Statement) ;
Compound_Statement = (Connector, Simple_Statement)+;
Connector = ("and" | "or");
Simple_Statement = Clause ;
Clause = (Binary_Clause | Unary_Clause) ;
Binary_Clause = (Subject, Binary_Verb, Object) ;
Unary_Clause = (Subject, Unary_Verb) ;
Subject = Noun_Phase ;
Object = Noun_Phase ;
Binary_Verb = Verb_Phase ;
Unary_Verb = Verb_Phase ;
```

Fig. 3. General rule pattern

#### 4.5 RuleCNL Semantics

The RuleCNL semantics is defined by automatic mappings from RuleCNL rules to the SBVR semantic formulation (SF) description. SF is a part of the SBVR specification [20] that provides a means for describing the structure of the meaning of rules expressed in NL that business people use. SF is an abstract and language independent syntax to represent the meaning of a rule in a set of logic structures so that it can be machine processed. The full SBVR SF is not presented in this paper, but can be found in [20].

Fig. 4 shows the resulting SF in the XML form generated by the RuleCNL tool from the example of Rule 1 below

Rule 1: It is obligatory that the customer "John" places at least one order

```
<sbvr:rule ruleID = "Rule1"/>
<sbvr:guidanceStatement xmi:id="stmt-formal" expression="stmt-formal-t" meaning="stmt-formal-p"/>
<sbvr:obligationFormulation xmi:id="obligation"/>
<sbvr:universalQuantification formulationEmbedsuniversalQuantification "the"/>
<sbvr:QuantificationIntroducesVariable variableRangesOverConcept " ref= xmi:customer"/>
<sbvr:projectionIsOnVariable variable ="ref= xmi:customer"/>
<sbvr:closedProjectionDefinesInstanceofNounConcept "John"/>
<sbvr:QuantificationScopesOverAtomicFormulation/>
<sbvr:atomicFormulation atomicFormulationIsBasedOnBinaryFactType " ref= xmi:places"/>
<sbvr:existentialQuantification formulationEmbedsexistentialQuantification "at_least" cardinality ="1"/>
<sbvr:QuantificationIntroducesVariable variableRangesOverConcept " ref= xmi:order"/>
```

**Fig. 4.** Mapping of a RuleCNL rule to SBVR semantic formulation

*Domain Terms* are mapped to Noun concepts, determiners are mapped to quantifications, *Domain Names* are mapped to Individual concepts, *Domain Verbs* are mapped to Fact types, coordinators are mapped to logical operators, relative clauses are mapped to Projections, etc.

## 5 RuleCNL Tool

### 5.1 Implementation

An important feature of a reliable CNL is its tool support because one of the biggest problems (if not the biggest problem) of CNLs is the usability of a new CNL by end-users [9]. In reality, the limited expressiveness due to the restriction on vocabulary and grammar of CNLs leads to the difficulty in writing statements that comply with the imposed restriction. Writing syntactically correct statements without tool support is much more complicated because the user

needs to learn syntax restrictions, which are in many cases not trivial to explain. Thus, we have developed editors that make the writing process and the usability of the RuleCNL as effortless as possible. There are two editors: a vocabulary editor (Fig. 5) and a rule editor (Fig. 6)

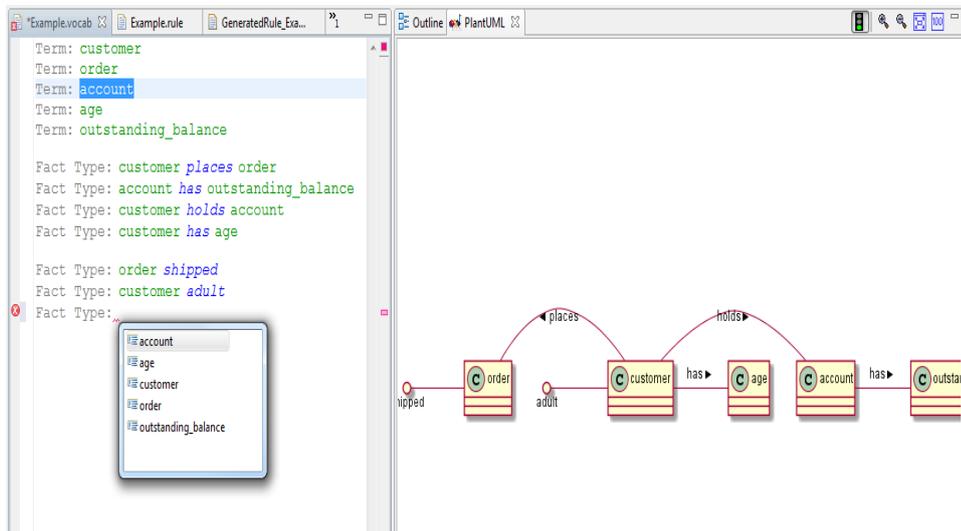
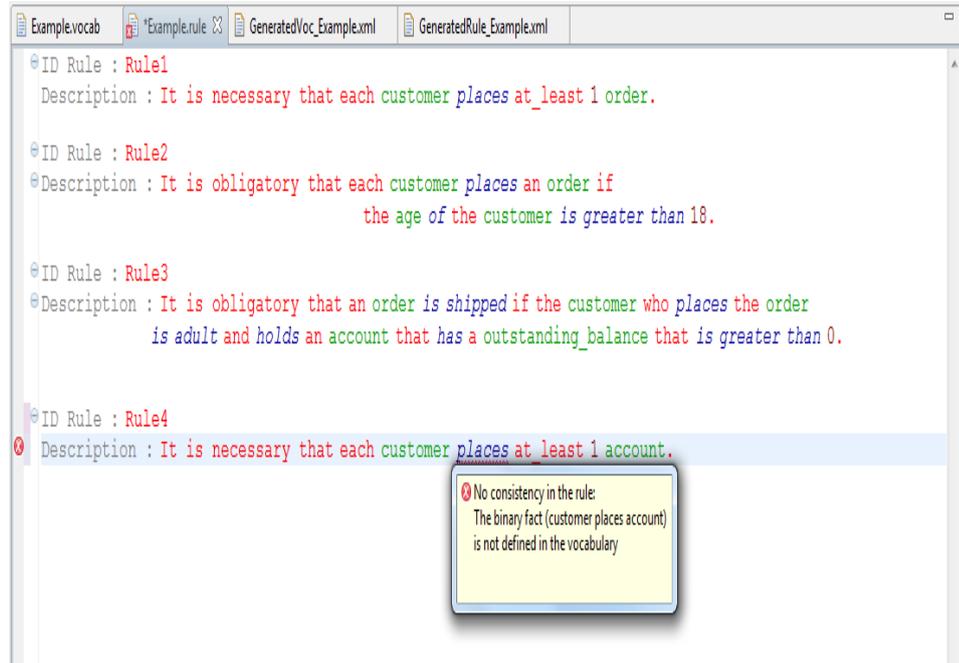


Fig. 5. Vocabulary editor



**Fig. 6.** Rule editor

The vocabulary editor assists the users to specify their vocabulary. It has two customized and dynamic views on the vocabulary in use: an outline view (tree-like) and a graphical view (UML-like notation). The rule editor assists the users to specify the business rule statements.

Both editors offer high level features such as auto-completion, error-handling, automatic highlighting and validation.

## 5.2 Evaluation

An ideal CNL should be effortless to learn and expressive enough to describe the domain problem. Thus, we have evaluated the RuleCNL with respect to its expressivity and comprehensibility. For the experiment, we collected about 50 business rules from real-life case studies of two companies written in the English and French languages. The first company operates in the domain of banking and insurance whereas the second is a parastatal. The evaluation was carried out by four end-users divided into two groups: group 1 is made up of two business experts with no background in formal notations of business rules and group 2 is made up of two business users with a background in information system technology. Our objective was twofold and consisted in finding how many business rules the RuleCNL could formalize in a natural way and how easy the users could understand the formalization. Thus, the evaluation's metrics are the

expressiveness and the users comprehension of the RuleCNL. One could have also added the readability, but as RuleCNL is close to NL, statements written in RuleCNL are read in the same way as in the underlying NL. Table 1 shows the result of the experiment with the agreement of the two users of each group. This result is the same both for English and French.

**Table 1.** Evaluation

Measures/Users	Group 1	Group 2
<i>Expressiveness</i>	84%	84%
<i>Comprehensibility</i>	90%	100%

As we can see in the Table 1, the expressiveness is 84% for both groups. The remaining 16% was because of syntactic and semantic ambiguities in some rules. However, with more training, the users rephrased these rules so that the tool was able to formalize. Regarding the comprehensibility, group 1 confirmed that it understood 90% of the formalized rules in a natural way and group 2 understood all the rules. This result is not surprising because group 1 users do not have a background in formal notations. Thus, the 10% remaining consisted of complex rules, which require much constraints imposed by the grammar.

## 6 Conclusions and Future Work

The ultimate goal to bridge the gap between natural and formal languages has brought interesting research challenges in the area of CNL. In this paper, we have presented the RuleCNL which is a CNL for business rule specifications. The aim of RuleCNL is to help business experts formalize their business rules in a business-friendly way that can be understood by computers. The RuleCNL syntax is defined by a formal grammar and its semantics is defined by automatic mappings of RuleCNL rules to SBVR semantic formulations. The RuleCNL tool provides editors that assist end-users in the writing process. RuleCNL along with its tool have been evaluated with satisfactory results from business experts. We are currently improving the tool and extending evaluation to many other companies. The future work will be to go from the SBVR SF to some production rules for rule engines or software components.

## References

1. Barbara, V. H.: Business Rules Applied: Building Better Systems Using the Business Rules Approach. John Wiley & Sons, Inc., New York, (2001)
2. Business Rule Group, <http://www.businessrulesgroup.org/defnbrg.shtml>
3. Business Rule Approach, <http://www.businessrulesgroup.org/bra.shtml>

4. Tony, M.: *Business Rules and Information Systems: Aligning IT with Business Goals*, Addison-Wesley Professional, (2002)
5. Ross, R. G.: *Principles of the Business Rule Approach*, Addison-Wesley Professional, USA, (2003)
6. Lucie, B., Sophie, R., Bertrand, G., Stefan, L., Michael, S.: *Report on State of the Art and Prospective Evolution of Formal Languages for Business Rules*, Public Research Centre Henri Tudor, Luxembourg, (2006)
7. Lévy, F., Nazarenko, A.: *Formalization of Natural Language Regulations through SBVR Structured English (Tutorial)*. In: Morgenstern, L., Stefaneas, P., Lévy, F., Wyner, A., Paschke, A. (eds.) *RuleML 2013*. LNCS, vol. 8035, pp. 19-33. Springer, Heidelberg (2013)
8. Rolf, S.: *Controlled Natural Languages for Knowledge Representation*. In: *COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics*, pp. 1113-1121 (2010)
9. Kuhn, T.: *Controlled English for Knowledge Representation*. Doctoral thesis, Faculty of Economics, Business Administration and Information Technology of the University of Zurich (2010)
10. Fuchs, N. E., Kaarel, K., Kuhn, T.: *Attempto Controlled English for Knowledge Representation*. In: *Reasoning Web*, LNCS, vol.5224, pp. 1041-1054, (2008).
11. White, C., Rolf S.: *An Update on PENG Light*. In: *Proceedings of ALTA 2009*, pp.80-88, (2009).
12. Clark, P., Harrison, P., Jenkins, T., Thompson, J., Wojcik, R.: *Acquiring and Using World Knowledge using a Restricted Subset of English*. In: *The 18th International FLAIRS Conference (FLAIRS'05)*, (2005)
13. Hart, G., Johnson, M., Dolbear, C.: *Rabbit: Developing a Controlled Natural Language for authoring ontologies*, In: M. Hauswirth, M. Koubarakis, S. Bechhofer (eds.), *Proceedings of the 5th European Semantic Web Conference*, LNCS, Springer Verlag, Berlin, Heidelberg, (2008)
14. Rolf, S.: *Controlled Natural Language as Interface Language to the Semantic Web*. In: *2nd Indian International Conference on Artificial Intelligence (IICAI 2005)*, Pune, India, (2005)
15. Cardey, S.: *Machine Translation of Controlled Languages for more Reliable Human Communication in Safety Critical Applications*. In: *Proceedings of the 12th International Symposium on Social Communication*, pp. 953-958, Cuba, (2011)
16. Terry, Halpin.: *Business Rule Verbalization*. In: *Proceedings of the 3rd International Conference ISTA 2004*, volume P-48 of *Lecture Notes in Informatics*, pp. 39-52, Germany, (2004)
17. Sergey, L., Gerd, W.: *Verbalization of the REWERSE I1 Rule Markup Language*. Deliverable I1-D6, REWERSE, September, (2006)
18. Ruth, R. P.: *An object-oriented approach to the translation between metaschemas application to the translation between UML and SBVR*. Doctoral thesis, Polytechnic University of Catalonia, Barcelona (2009)
19. Anderson, K., Spreeuwenberg, S.: *SBVR's Approach to Controlled Natural Language*. In: *Proceedings of the Workshop on Controlled Natural Language*, pp. 155-169, Italy, (2009)
20. *Semantics of Business Vocabulary and Business Rules (SBVR)*, v1.0. Object Management Group, <http://www.omg.org/spec/SBVR/1.0/>, (2008)
21. *RuleSpeak*, <http://www.rulespeak.com/en/>