An ontological approach for modeling technical standards for compliance checking

Khalil Riad Bouzidi, Catherine Faron Zucker, Bruno Fies, Nhan Le Thanh

An ontological approach for modeling technical standards for compliance checking

Khalil Riad Bouzidi[1,2], Catherine Faron-Zucker[2], Bruno Fies[1] and Nhan Le Than[2]

[1]CSTB, 290 route des Lucioles, BP 209, 06904 Sophia Antipolis,
{khalil-riad.bouzidi, bruno.fies}@cstb.fr
[2]Laboratoire I3S, Université de Nice Sophia Antipolis, CNRS,
930 route des Colles, BP 145, 06903 Sophia Antipolis,
{catherine.faron-zucker, nhan.le-thanh}@unice.fr

Abstract. This paper gives an overview of a formal semantic-based approach of modeling some regulations in the photovoltaic field to help the delivering of technical assessments at the French scientific center on Building Industry (CSTB). Starting from regulatory texts, we first explicit SBVR rules and then formalize them into ontology-based rules in the SPARQL language. These are exploited in the modeling of the compliance checking process required for the delivering of technical assessments.

Keywords: Ontology, Semantic Web, Knowledge Management, Building Industry, E-regulations, E-Government

1 Introduction

The French initiative “Grenelle de l’Environment”, launched in France in 2007 is an open multi-party debate aiming to define the key points of public policy on ecological and sustainable development issues. It will generate additional information of technical regulatory whose analysis is viewed by professionals as an increasing burden. Many rapid changes of regulations and technical documents will intervene to keep the commitment made mainly with regard to optimizing the energy aspects of building.

In this general context, the research communities of Knowledge Engineering and Semantic Web have a key role to play to provide models and techniques to simplify access to technical regulatory information, facilitate its appropriation, support professionals in its implementation, and facilitate the writing of new regulations while taking into account constraints expressed in the existing regulatory corpus.

Our work aims to propose dedicated semantic services to support the process of modeling the compliance checking or regulations. We focus on the process of acquiring expressions in the Semantics of Business Vocabulary and Rules (SBVR) language from textual technical standards and the transformation of these SBVR expressions into interpretable ontology-based rules.

In the next section we present our approach to model technical standards in SBVR. We detail in section 3 the transformation of SBVR rules into ontology-based rules represented in the SPARQL language. In section 4, we present our approach to interpret compliance checks results. We conclude in section 5.

2 Modeling Business Rules

SBVR stands for "Semantics of Business Vocabulary and Business Rules". It is an OMG standard whose ultimate objective is to provide a meta-model that allows establishing data exchange interfaces for tools that create, organize, analyze and use vocabularies and business rules [1], [2]. SBVR rules are based upon the Predicate Logic: they capture their semantics and not the way they must be executed. SBVR is not an executable formalism. It is particularly addressed to business experts. It uses a controlled natural language that all business experts understand. It does not have a specific rule format.

The development of an SBVR base is done in two steps: the development of a business vocabulary and the writing of business rules based on the terms and concepts defined in the vocabulary. SBVR controlled vocabularies consist in hierarchies of concepts specific to some domain, their relationships, definitions and synonyms.

We built an ontology for technical documents. OntoDT is extracted manually from the weft provided by CSTB for these documents. This weft contains all information necessary for a process, a product or a material to be studied by a specialized group at CSTB who will be responsible for delivering a Technical Assessments. This ontology of technical document primarily consisted in a hierarchy of primitive concepts in OWL Lite. We merged OntoDT with the terms of the REEF thesaurus developed by CSTB for the Building Industry. By reusing the REEF we built a controlled vocabulary standard enabling to link technical assessments (ATec) to the REEF. As a result; our ontology of technical document contains 50 classes and 26 properties. 35% of these classes are created from REEF terms; the remaining 65% are concepts more specific than those of the REEF thesaurus which contains general concepts of the building industry. In its current state, it lacks specific terms relative to a particular field (Photovoltaic). However, it remains in constant evolution.

2.1 Transforming standards into SBVR rules

Technical standards can be understood in different ways and this is why the manual intervention of a domain expert is essential. We argue that NLP approaches of knowledge extraction from regulatory texts can significantly alleviate the task of domain experts but cannot replace them. In our work, we do not consider linguistic analysis of texts and focus on the representation of expert knowledge. CSTB experts helped us to identify and classify the constraints expressed in the photovoltaic standards and then the rules which represent them.

The extraction of rules from standards or statutory text is a tedious job, it often requires to structure the information. The descriptions used have been detailed enough to show how the content of standards can be converted into SBVR vocabulary and business rules. However, a clarification of the text was needed before the transformation into SBVR. The steps below are necessary in order to produce an understandable SBVR text:

- **Disambiguation:**

Let us start from the following sentence as regulatory text:

Classe A: Accès général, tension dangereuse, applications de puissance dangereuses : Les modules assignés pour utilisation dans cette classe d'application peuvent être utilisés dans des systèmes fonctionnant à plus de 120 V en courant continu. Les modules requis pour la sécurité dans la présente partie de l'EN 61730 et l'EN 61730-2 dans cette classe d'application sont considérés comme satisfaisant aux exigences de la classe II de sécurité.

This text means that a photovoltaic module is of class A if its system runs over more than120 volt and satisfies the requirements of security class II. The first step is to rewrite this text into SBVR rules by using the ontology of technical document so that the text will be read as follows:

If **a module** has **a system** which runs over more than **120 volt**, and this module is satisfying **security class 2**, then it is of **class A**

The concepts identified in this fragment are Module, System, Class, which belong to the ontology of technical document.

- **Reformulation.**

Let us consider the following regulatory text:

Les dimensions du châssis principal doivent être :

Largeur intérieur : (847 ±5) mm

Hauteur intérieur :(1910 ±5) mm

This standard extract expresses conditions that are difficult to read by non expert readers. It needs a reformulation to be understood:"The maximum width of a main frame must be lower or equal to 853mm and the minimum width higher or equal to 842mm. The maximum height of a main frame must be lower or equal to 1915mm and the minimum height greater than or equal to 1905mm".

If **a frame** has **a minimum** width higher or equal to **842mm** and **a minimum height** higher or equal to **1905mm** and **a maximum width** less than or equal to **853mm** and **a maximum height** less than or equal to **1915mm**, then it is **a main frame**

- **Formulations to avoid.**

Let us consider the following regulatory text:

« Un module doit avoir une isolation assignée pour un minimum de 90 °C, avec un calibre et des caractéristiques de tension acceptables pour l'application... »

This case contains information not only ambiguous but constraints which cannot be interpreted: "with a size and voltage characteristics acceptable for the application" This reformulation is considered non interpretable which avoids its reformulation in SBVR.

3 Transformation of SBVR rules into ontology-based SPARQL rules

SBVR describes business concepts and requirements without addressing their implementation. To validate our model, we developed a rule base for checking the compliance of technical documents to regulatory standards in the photovoltaic field. Our rules are formalized in the SPARQL language and annotated in the RDF language. This enables us to both automatically reason on the representation of regulatory standards and to model the process of compliance checking.

With the help of CSTB experts we have defined orders of processing business rules based on priorities existing among standards of the same domain, among the constraints of a standard and the additional knowledge that specifies the rules of the same constraint. We aim to formalize the experts reasoning like this:

- **In the same domain: S**tandards that describe a device (comprising multiple components) have priority on those that describe unique components. Example: Standard NF[1]61730 which describes “the requirement of PV module” has priority over NF12600 which describes “the classification of glass”
- **In the same standard: C**onstraints that describe the compliance of a product have priority on those relative to its components. Example: in standard NF61730, application class constraints of PV module have priority on electrical cable constraints used in this module.
- **In the same constraint:** The representation of a constraint can involve multiple business rules. In that case, there are priority orders that can be extracted from standards when identified by experts. We represent it in RDF in a so-called annotation of standards requirement (ASR) and we rely on these ASR to schedule rules in the conformance checking process. These annotations are based on OntoDT; the requirement scheduling is mainly based on priorities among components. For instance, in Standard NF 61730 described in Fig. 1, priority requirements are: solar energy, solar cell, etc.

```
<Dt:Standard rdf:ID="NF61730">
     <Dt:hasForRequirement rdf:parseType="Collection">
        <rdf:Description rdf:about="#SolarEnergie"/>
        <rdf:Description rdf:about="#SolarCell"/>
        ...
     </Dt:hasForRequirement >
</Dt:Standard>
```

Fig. 1. Extract of an ASR

We use SPARQL query patterns like that in Fig. 2 in order to identify requirement orders by querying ASR annotations before browsing the explicit dependency rules.

```
SELECT ?requirementList WHERE{
    ?standards rdf:type DTonto:Standards
    ?standards DTonto:hasRequirement ?req
    ?req rdf:rest*/rdf:first ?requirementList
   FILTER( ?standards = DTonto:”StandardsID”)}
```

[1] http://www.afnor.org

Fig. 2. Example of a SPARQL query pattern for ASRs

4 Interpretation of compliance checking results

The implementation of our compliance checking model is based upon the matching of standards representations with those of a technical document, i.e. the matching of SPARQL queries with RDF annotations if there are conditions for applying the standard. For this purpose, we use the CORESE semantic engine [3].

One major problem when automating the compliance checking process is to justify the decisions taken by the system – compliance or noncompliance of the product. Technical standards that validate the products are modeled as rules of compliance checking. The failure of these rules means the noncompliance of some component.

We propose an approach based on reaction rules that set off as soon as noncompliance is identified. This kind of rules helps to identify failing compliance checking rules and the component or property responsible of noncompliance.

We use reaction rules to explain the "why" of the negative decision. These rules are triggered by **Event** occurrences and require satisfaction of **Conditions** to perform **Actions**. They are therefore called ECA rules. According to several works, [4], [5], ECA rules are best adapted to describe the logic of a process by a set of rules. They are extensions of production rules (rules where the event part is missing; noted Condition-Action "CA").We use ECA rules to represent in a declarative manner the logic of a process and explain negative decision. Our model of reaction rule is as follows:

- **Event**: represents the context of implementation of an action. It is defined by the noncompliance of a component caused by the failure of a compliance checking rule.
- **Condition**: is checked during the identification of noncompliant products. CORESE has the notions of Event and Event Listener that enable to catch some predefined events. Events are related to SPARQL query processing. In our case, we will implement an Event Listener to capture the cause of failure, i.e. of non compliance.
- **Action**: A message display to an external user to explain the result of applying the condition. In our case it would represent the "why" of making a negative decision.

The originality of our approach is to use reaction rules to justify the decision taken in case of noncompliant of a photovoltaic product. The triggering event is the failure of one or more compliance checking rules of a specific product "$\mathbf{R}_{\mathbf{noncompliance}}$". In the condition we check the failure rules events in CORESE engine to identify the real cause of noncompliance "$\mathbf{Check}_{\mathbf{Events}}$". The action is a response message displayed that explains the noncompliance of the product "**Response**".

We will use a knowledge base containing a list of explanations produced by experts. Each explanation or response is unique to a single state of noncompliance. The answer will be extracted by using a SPARQL query that takes as input the noncompliant component and displays as output the appropriate response.

Let us consider the example of a noncompliant application class A of a photovoltaic module. First, the process is triggered when a noncompliance of the photovoltaic module to Application Class A is identified. The Application Class rules contain two parameters: the Voltage and the Security Class. The noncompliance of one of these two parameters leads to the failure of the rules. The verification of which parameter are noncompliant is carried out. We identify the failure request by catching the Event which demonstrates it.

5 Conclusion

The main contribution of our research is twofold. First, we propose a method to formalize paper-based regulation texts using SBVR standards and ontology based rules. Second, we propose a model based on ECA rules to explain negative decisions taken by the system.

We aim to use simple and unambiguous standard language conventions to specify regulation. SBVR modeling represents a quality assurance of the technical standards text and decrease later modification needs. It uses an implicit model (SBVR vocabulary) and expresses definitions, concepts and restrictions with a clear associated semantics.

This controlled vocabulary is an ontology called OntoDT formalized in OWL-Lite and coupled with the REEF thesaurus for the building industry. SBVR defines correspondence between SBVR rules and implementation rules which can help to formalize the SBVR concepts in terms of existing rule languages. In our case we use SPARQL as SBVR implementation rule language. Our model integrates expert knowledge in the process of compliance checking.

This knowledge is expressed into annotations representing standards in rules format. Annotations are also used to establish a compliance report that interprets the matching between compliance rules and technical document representations, especially in case of noncompliance. To explain the noncompliance we use reaction rules based on the ECA model. This work is still at a premature phase, experimentation and full evaluation must be completed.

Reference

1. The Object Management Group OMG. Semantics of Business Vocabulary and BusinessRules (SBVR). OMG Speciation (2006).
2. Chapin, D., Baisley, D. E., Hall, H.: Semantics of Business Vocabulary & Business Rules (SBVR). In Rule Languages for Interoperability, W3C Workshop on Rule Languages for Interoperability,Washington, DC, USA (2005)
3. Corby, O. Dieng-Kuntz, R., Faron-Zucker, C.: Querying the SemanticWeb with Corese Search Engine. Proc. of the 16th Eureopean Conference on Artificial Intelligence, ECAI, IOS Press, 705--709 (2004)
4. Bry, F., Eckert, M., Pătrânjan, P. L., Romanenko, I.: Realizing Business Processes with ECA Rules: Benefits, Challenges, Limits. In 4th International Workshop, PPSWR 2006, Budva, Montenegro (2006).
5. Knolmayer, G., Endl, R., and Pfahrer, M.: Modeling Processes and Workflows by Business Rules:. In Business Process Management, Models, Techniques, and Empirical Studies. Aalst, W. M., Desel, J., Oberweis, A. (eds.) Lecture Notes In Computer Science, vol. 1806. Springer- Verlag, London, 16--29. (2000).