Applying EDON Methodology and SBVR2OWL Mappings for Building an Ontology-Aware Software

Cecilia Gaspoz, Valeria Bertossi, Emiliano Reynares, Ma. Laura Caliusco

CIDISI Research Center – UTN – FRSF – Lavaise 610 – Santa Fe – Argentina
ceciliagaspoz@gmail.com, valeriabertossi@live.com.ar,
{reynares, mcaliusc}@frsf.utn.edu.ar

Abstract. In Ontology-Aware Software, ontologies are use at run time to, for example, use their content in operations of information searching or as database substitutes for information storage. In order to integrate the software development and ontology building processes, involved in building ontology-aware information system a methodology called EDON have defined. The main disadvantage of this methodology is that the heuristic to generate an implemented ontology from the requirement elicitation is not complete enough. On the other hand, recently, the Object Management Group (OMG) has standardized a language called Semantics of Business Vocabulary and Rules (SBVR) and different approaches have been proposed to map SBVR expressions into the OWL ontology language. In this paper, we report our experience in developing an ontology-aware information system by using an adaptation of the EDON methodology including the SBVR2OWL mappings.

Keywords: Ontology-Aware Information System, SBVR2OWL Mappings

1 Introduction

Since the latter part of the 20th century there has been a growing interest in applying the ontology in the context of software engineering due to the advent of the Semantic Web and the technologies for its realization. In the software engineering context, an ontology can be used at run time in two different ways: (1) as Architectural Artifacts (Ontology-Driven Software), ontologies are used as central elements of the proposed software architecture, and (2) as Information Resources (Ontology-Aware Software), ontologies are used at run time in order to, for example, use their content in operations of information searching or as database substitutes, for information storage [3].

In the context of ontology-aware software, developers have to face the problem of how to integrate software development and ontology building methodologies assuring the project success. The development of methodological approaches for building an ontology as software artifact is still an open research area. There are many languages, techniques and tools for the representation, design and construction of ontologies [5]. But the great majority of these have been created for and by the knowledge engineering community. Because of this, the use of ontologies by Software Engineering professionals and researchers can be seen as an additional learning experience, and in some cases, of considerably great effort [3]. Moreover, a survey [14] showed that approximately 50 % of its participants did not use any ontology engineering methodology in large-scale projects.

In order to avoid this problem, Reynares et. al. [13] have defined a methodology called EDON to build an ontology-aware system. This methodology proposes to develop an ontology that fulfills the requirements of the development cycle to which it belongs. From requirements, through CQs and LELs, you get the necessary information about the domain which is then captured as objects, relationships and properties in the implemented ontology. With regard to CQs, they can lead to create objects, relations or properties that are not relevant to the system, but they are for the environment in which the system is embedded. This happened to us in our development and is mainly due to those who are not familiar with the development of ontologies think in terms of the system. With regards to LELs, the heuristics used to build the ontology from them is not complete enough [1]. Then, although the ontology conceptualization by using CQ and LELs has proven to be useful to facilitate the communication among the DEs, SEs and KEs, a more powerful formalism will improve the way complex business rules are expressed.

Recently, the Object Management Group (OMG) has standardized another language called Semantics of Business Vocabulary and Rules (SBVR) [10]. SBVR has been conceptualized for business people and designed to be used for business purposes independent of information systems designs. The linguistic approach adopted by the proposal enables the expression of business knowledge through statements rather than diagrams. That is rooted in the insight that diagrams are helpful for depicting structural organization of concepts but they are impractical as a primary means of defining vocabularies and expressing business rules. Different approaches have been proposed to map SBVR expressions into OWL language [12].

The objective of this paper is to report our experience in developing an ontology-aware information system by using an adaptation of the EDON methodology including the SBVR2OWL mappings defined by Reynares et al. [12]. To this aim, the paper is organized as follows. Section 2 defines the concepts necessary to understand the content of this paper. Section 3 describes the development the Ontology-Aware Information System. Finally, Section 4 is devoted to discussion and lessons learned.

2 Conceptual Foundations

2.1 Evolutionary Development of ONtologies (EDON)

EDON [13] is an approach for building from scratch an ontology intended to be used as a structural conceptual model of an information system, encoding business rules in a declarative way. EDON adopts a requirement driven, iterative, and incremental approach and it is composed by the processes described next.

Requirements Selection Process. This process is composed by three activities: (1) identification of the functional requirements that involves business rules in their meeting, (2) identification and prioritization of the domain entities involved in the meeting of the requirements identified before, and (3) requirements grouping and selection according to the importance of the entities involved.

Ontology Development Process. This process involves Development Activities that allows evolving from an abstract model toward a computable ontology, and Support Activities are carried out along the whole development process. The Development Activities are: specification, conceptualization, formalization, refinement, implementation and alignment. The Support Activities are: knowledge elicitation and evaluation. The techniques to carry out them are based on the different methodologies and good practices for building ontologies developed since mid-1990 [5]. EDON considers the performing of the refinement activity with the aim of extending the ontology by focusing on the declarative formulation of business rules.

Ontology Alignment Process. Each application of EDON produces an ontology that supports a disjoint set of functional requirements, i.e., those selected on the specification activity of the iteration. Therefore, the alignment of current and previous version of the ontology is needed as a way to support both set of requirements. Ontology alignment is the process of determining the different types of (interontology) relationships among their terms [11]. As a result, a new ontology composed by sub-ontologies is created.

2.2 SBVR2OWL Mappings.

SBVR. SBVR [10] defines the vocabulary and rules for documenting the semantics of business vocabularies, business facts, and business rules; which allows their verbalization in a controlled vocabulary readily understandable by business people. The fact-oriented approach of SBVR stems from the Business Rules Manifesto [2], stating that rules builds on facts, and facts build on concepts as expressed by terms. Therefore, terms express business concepts, facts make assertions about these concepts, and rules constrain and support these facts. SBVR supports such approach by providing noun concepts and verb concepts respectively corresponding to the notions of terms and facts.

As early stated, SBVR adopts a linguistic approach that allows to define vocabularies and express operative rules. According to this insight, SBVR defines a Controlled Natural Language (CNL) and describes the way to mechanically mapping such CNL expressions to SBVR formal concepts.

OWL. The OWL 2 Web Ontology Language (OWL 2) is the latest version of an ontology language proposed by the World Wide Web Consortium (W3C) [16]. OWL 2 ontologies provide classes, properties, individuals, and data values, and are stored as Semantic Web documents. An OWL 2 ontology is a formal description of a domain of interest rooted in three syntactic categories that are interpreted under a standardized semantics, which allows useful inferences to be drawn.

- Entities, such as classes, properties, and individuals. They are the basic elements of an ontology and are identified by Internationalized Resource Identifiers (IRIs) [7].
- Expressions, representing complex notions in the domain being described.
- Axioms, which are statements asserted to be true in the domain being described.

OWL 2 ontology language defines several concrete syntaxes that can be used to serialize and exchange ontologies. Among them, the functional style syntax is defined in the OWL 2 structural specification [7] with the aim to state the semantics of OWL 2 constructors and allow a compact writing of ontologies.

Mappings. Mappings defined by Reynares et. al. [12] allow the automatable generation of an OWL2 ontology from the SBVR specifications of a business domain. Transformations are rooted on the structural specification of both standards and are depicted in subsections below by grouping and sequencing them according to the inherent logical order of the subject matter itself. In addition to their theoretical expression, the mappings are illustrated by building an ontology that reflects the business knowledge exposed by a case study. Some of these mappings are shown in Table 1.

Table 1. An excerpt of the Mappings defined by Reynares et.al.[12]

1. Each object type ot is mapped to Declaration(Class(a:ot))
2. exactly-n Quantification, where "n" is a non-negative integer:
   - If the logical formulation scopes over a unary fact type, the expression is mapped to DataExactCardinality(n a:DataPropertyOne a:DataRangeOne)
   - If the logical formulation scopes over a binary fact type, the expression is mapped to ObjectExactCardinality(n a:ObjectPropertyOne a:ClassOne)

3 Applying EDON and SBVR to OWL2 Mappings for developing an ontology-based system.

The methodology applied in the development of the fellow recommender system is based on EDON methodology [13], which was adapted, in the experience describe in the following subsections, to include the SBVR to OWL Mappings [12].

3.1 Requirements Selection.

Requirements were classified in two classes: those requirements that will be supported by the ontology and those which will not. Some requirements of the first class were selected to implement in a first iteration of the development process. A storyboard exposing a functional requirement belonging to the selected subset is: *"The system should evaluate the indicators involved in the point assignment process for* *each one of the candidates and the order of those candidates based the general indicator. The ranking should be display on screen."*

Alumno (Student), Materia (Subject), Universidad (University), Facultad, Carrera (Career), PlanDesarrolloAcadémico (AcademicDevelopmentPlan), Beca (Fellowship), SituaciónAcadémica (AcademicSituation), SituaciónEconómica (EconomicSituation) were identified as the core entities involved in the meeting of the requirements identified before.

3.2 OntologyDevelopment.

Specification. Based on the core entities identified before and the general knowledge of the problem, Competency Questions (CQs) were proposed. An excerpt of them is showed in Table 2. From the CQs, a list of the domain entities needed for answering them was identified. Some of these domain entities are: Postulante (Applicant - student enrolled in a fellowship), Candidato (Candidate – applicant who meets every requirement), Becario (Fellow – Candidate to whom the fellowship has been granted).

Table 2. An excerpt of the Competency Questions

- Are every applicant to the university fellowship registered during the registration period?
- Are all candidates regular students?
- Which are the aspects related with the academic situation of the candidate that impact in the ranking process?
- Is the list of the candidates order by decreasingly based in the general indicator?
- Has every candidate approved at least 5 subjects during the last school year? Those who not, ¿are those new students?

Conceptualization. In this activity, the knowledge about the domain entities was collected from the information sources: the university's fellowship regulations and a fellowship management report. The business rules extracted from these resources were written in natural language, in order to represent them independently of the modeling paradigm and the implementation language of the target ontology.

Formalization. The business rules identified were translated from the natural language to SBVR. This activity includes: Recognize the noun concepts, the fact types and keywords; differentiate noun concepts belonging to complex concepts from noun concepts belonging to datatypes, re-elaborate the fact type according to the fact being represented, build the business rules by applying restrictions on the statements.

Then, the business vocabulary was organized by means of vocabulary entries, as shown in Table 3.

Table 3. SBVR specification of "Beca" (Fellowship) concept.

Beca
• Definitions: • General Concept: • Concept Type: Object Type • Necesity: – each beca tiene ciclo exactly one ciclo beca – each beca tiene plazo inscripción exactly one plazo inscripción – each beca tiene valor canasta familiar exactly one canasta familiar • Posibility:
Ref: tiene ciclo: has school year - ciclo beca: school year of the fellowship - plazo de inscripción: registration period - canasta familiar: basic market basket indicator.

Ontology Implementation. In order to create the ontology implementation, the SBVR2OWL transformations, defined by Reynares et. al. [12], were apply to the SBVR model of the business vocabulary created in the previous activity. An example of this process is showed in Table 4. The ontology was implemented using the free ontology editor called Protégé and the Pellet inference engine that provides sound-and-complete OWL-DL reasoning services. The ontology was written in OWL-DL 2.0 ontology language and serialized in OWL/RDF format.

Table 4. OWL specification of "Beca" concept.

```
Declaration(Class(BecaUTN:Beca))
SubClassOf(BecaUTN:Beca
    ObjectMinCardinality(1 BecaUTN:tieneCiclo BecaUTN:CicloBeca)))
SubClassOf(BecaUTN:Beca ObjectMinCardinality(1 BecaUTN:tienePlazoInscripcion
    BecaUTN:PlazoInscripcion)))
SubClassOf(BecaUTN:Beca
    DataExactCardinality(1 BecaUTN:CanastaFamiliar xsd:float))
```

Refinement. The resulting ontology represents the main concepts of the problem domain. The refinement activity consists in further extending the ontology by focusing on the formulation of rules, which are obtained from the knowledge and information sources identified in the specification activity. The rules allow implementing the algorithm for making the fellows' ranking, and several classifications, e.g. each instance of Alumno (Student) can be classified in Postulante (Applicant) and/or Candidato (Candidate); each instance of Examen (Test) is classified in ExamenAprobado (ApprovedTest) and ExamenNoAprobado (FailedTest), etc.

The rules were implemented in the Semantic Web Rule Language (SWRL), which provides the ability to express Horn-like rules in terms of OWL concepts [9]. Table 5 shows some of the rules implemented in the study case.

Table 5. An excerpt of the rules implemented in SWRL

- Examen (?examen), calificacionExamen(?examen,?calificacion), greaterOrEqual(?calificacion, “4”^^UnsignedShort) → ExamenAprobado(?examen)
- Examen (?examen), calificacionExamen(?examen,?calificacion), lessThan(?calificacion, “4”^^UnsignedShort) → ExamenNoAprobado(?examen)
- Alumno(?alumno), esIngresante(?alumno, true) → AlumnoRegular(?Alumno)
- Alumno(?alumno), esIngresante(?alumno, false), rinde(?alumno,?exam1), rinde(?alumno,?exam2), ExamenAprobadoCicloAnterior(?exam1), ExamenAprobadoCicloAnterior(?exam2), DifferentFron(?exam1, ?exam2), → AlumnoRegular(?Alumno)

3.3 Ontology Evaluation.

Quality evaluation task was performed by means of OQuaRE [4], a framework conceived for that purpose and based on the SQuaRE standard for software quality evaluation [6]. OQuaRe defines a quality model which is divided into a series of characteristics organized into subcharacteristics which are evaluated by applying a set of automatable metrics. OQuaRE defines the criteria to transform the quantitative scores of each metric into a 1-5 range and establishes that 1 means not acceptable, 3 is minimally acceptable and 5 exceeds the requirements. After such transformation, score for each subcharacteristic is the mean of its associated metrics while the score of each characteristic is the mean of its sub-characteristics. The set of characteristics scores is the quality assessment result, enabling the identification of strengths and flaws of the ontologies rather than simply pointing out a “best ontology". Dimensions evaluated, shown in Figure 1, are defined as follows:

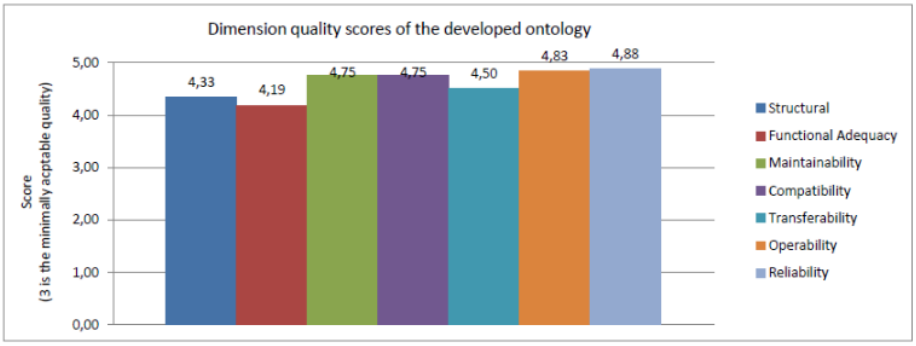

Fig. 1. Characteristics scores of the ontology developed

- Structural dimension involves formal and semantic properties that are important when evaluating ontologies since it accounts for quality factors such as consistency, formalization, redundancy or tangledness.
- Functional adequacy dimension refers to the appropriateness of the ontology for its intended purpose, according to the categories identified by [15].

- Maintainability dimension is related to the capability of the ontologies to be modified for changes in the environment, in requirements or in functional specifications.
- Compatibility dimension refers to the ability of two or more ontologies to exchange information and/or to perform their required functions while sharing the same hardware or software environments. The compatibility dimension can be evaluated over a single ontology - although intuitively it involves properties about more than one ontology - given that it is quantitatively assessed by means of a set of metrics applied to each ontology separately.
- Transferability dimension is the degree to which the ontology can be transferred from one environment (e.g., operating system) to another.
- Operability dimension refers to the effort needed to use the ontology and, in the individual assessment of such use, by a stated or implied set of users.
- Reliability dimension is the capability of the ontology to maintain its level of performance under stated conditions for a given period of time.

A quickly recognizable outcome is the level of quality shown by the ontology: according to the meaning assigned for OQuaRE to the values of the 1-5 ranking system, it largely outperform the minimally acceptable quality in all considered dimensions. Moreover, the global quality score - which is equal to 4.60 and it is calculated as the mean of all the scores - is very close to the maximal quality value.

3.4 Ontology Alignment.

The alignment activity consists in determining the different types of (inter-ontology) relationships among their terms [8] [15]. As a result, a new ontology composed by sub-ontologies is created. The first version of the ontology does not involve the performing of alignment activities. As single iteration of this EDON adaptation was performed, this activity was not required.

3.5 Fellow Recommender System Implementation.

After the ontology evaluation, the Fellow Recommender System was implemented in Java by using the JENA framework. This Software includes inscription, academic plan’s punctuation and fellow’s ranking functions, as shown in Figure 2. With regards the software quality, the functionality, efficiency, reliability and maintainability are closed related with those measured by the ontology since it is the core of the system. The usability was evaluated by a domain expert who gives a useful feedback to improve our system in a further work.

4 Discussion And Lessons Learned

In this paper we have reported our experience and showed the satisfactory results in developing an ontology-aware Fellow Recommender Systems using the EDON

Method adapted to include SBVR language to write business rules, in the formalization activity of the ontology development process, and SBVR to OWL2 Mappings, to be used during implementation activity.

Business Rules are usually embedded in the procedural part of the application. Using Ontologies to encapsulate them, made easier the modification and adaptation processes, allowing to use these system in others environments, such as other universities, without making a lot of changes. This is because, in order to adapt the system to other universities, the set of Business Rules defined in the ontology, is the only thing that have to be modified.

Related with EDON some advantages can be mentioned that we identified form this experience. The use of CQs can lead you to identify objects, relations or properties, that are part of the domain of the problem that is attach by using an ontology-aware system, as well as restrictions, which can guide you in the ontology testing process.

On the other hand EDON proposed to align the ontologies that are developed throughout the history of the system, allowing and providing the system to grow. This is an important feature to get extensible systems, which could adapt to new requirements, e.g. handling a new type of fellowship.

Finally, Adding SBVR and SBVR2OWL Mappings to EDON Methodology, made the ontology development process of this system easier and fluid, making the transition from the business rules to the implemented ontology, natural, simple and intuitive, focusing in the conceptualization and formalization process and without taking great efforts during the implementation process.

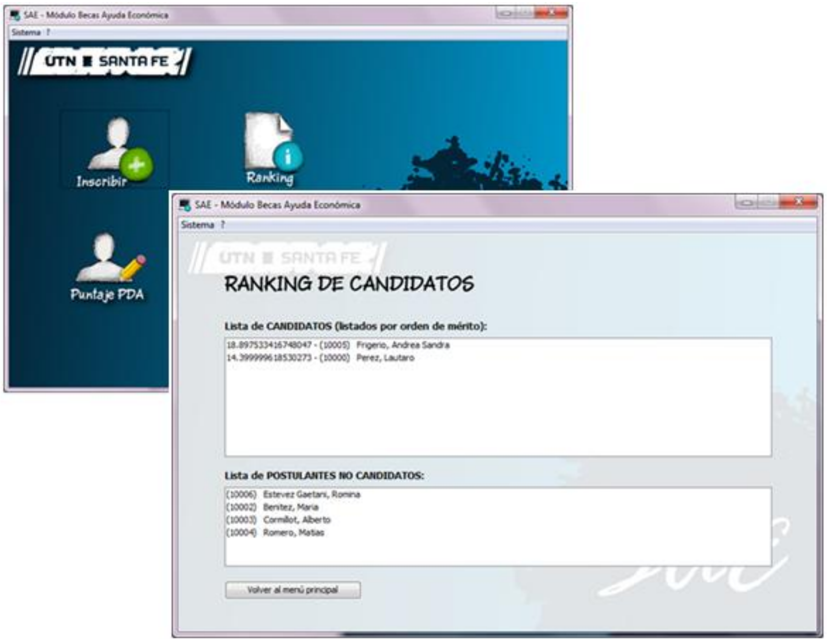

Fig. 2. The ontology – aware system implemented

5 References.

1. Ayub, C.; Cian, A.;Reynares, E., Caliusco, Ma. L. "An Experience Report on using the EDON Method for Building a Team Recommender System". 42° Jornadas Argentinas de Informática - Simposio Argentino de Ingeniería de Software. Córdoba (Córdoba - Argentina) Septiembre 2013. En prensa.
2. Business Rules Group: Business Rules Manifesto. Version 2.0. Accessible in: http://www.businessrulesgroup.org/brmanifesto.htm (2003)
3. Calero, C. and Ruiz, F. and Piattini, M., eds.: Ontologies for Software Engineering and Software Technology. Springer. (2006)
4. Duque-Ramos, A. and Lopez, U. and Fernandez-Breis, J. T. and Stevens, R. and Aussenac-Gilles, N.: OQuaRE: a SQuaRE-based approach for evaluating the quality of ontologies. Journal of Research and Practice in Information Technology, 43, 159- 173.
5. Gómez-Pérez, A. and Fernández-López, M. and Corcho, O.: Ontological Engineering. Springer/Heidelberg. (2004).
6. International Organization for Standardization (ISO) ISO/IEC 25000 2500, Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE)
7. Internet Engineering Task Force (IETF): RFC 3987: Internationalized Resource Identifiers (IRIs). Accessible in: http://www.ietf.org/rfc/rfc3987.txt (2005)
8. Liu, P. and Dew, P.: Using semantic web technologies to improve expertise matching within academia. In Proc.: I-KNOW '04, pages 370–378, (2004)
9. O'Connor, M. Knublauch, H., Tu, S. & Musen M.: Writing rules for the semantic web using SWRL and Jess. In: Proc. 8th Int. Protégé Conference, Protégé with rules, (2005).
10. Object Management Group (OMG): Semantics of Business Vocabulary and Business Rules (SBVR). Version 1.0: Formal Specification. Accessible in: http://www.omg.org/spec/SBVR/1.0/ (2008)
11. Pavel, S and Euzenat, J.: Ontology Matching: State of the Art and Future Challenges. In IEEE Transactions on Knowledge and Data Engineering, PP-99 (2011).
12. Reynares, E. and Caliusco M. A. and Galli, M. R.: An Automatable Approach for SBVR to OWL2 Mappings. In Proc. XVI Ibero-American Conference on Software Engineering (CIbSE 2013) Montevideo, Uruguay. (2013)
13. Reynares, E. and Caliusco M. A. and Galli, M. R.: EDON: A Method for Building an Ontology as Software Artefact. In Proc. 41st Argentine Conference on Informatics - 13th Argentine Symposium on Software Engineering (JAIIO - ASSE 2012) La Plata, Buenos Aires, Argentina. (2012)
14. Simperl, E., Mochol, M, and Bürger, T: Achieving Maturity: the State of Practice in Ontology Engineering in 2009. International Journal of Computer Science and Applications, 7(1):45 – 65, 2010.
15. Stevens, R. and Wroe, C. and Gobel, C. and Lord, P.: Application of ontologies in bioinformatics. In Handbook of Ontologies in Informations Systems. Staab, S. and Studer, R. (eds). pp 635-658. Springer. (2008).
16. World Wide Web Consortium (W3C): OWL 2 Web Ontology Language. Structural Specification and Functional-Style Syntax. Accessible in: http://www.w3.org/TR/2009/REC-owl2-syntax-20091027/ (2009).