Christophe Debruyne, Quentin Reul, and Robert Meersman. GOSPL: Grounding Ontologies with Social Processes and Natural Language. In S. Latifi, editor, ITNG, pages 1255-1256. IEEE Computer Society, 2010

GOSPL: Grounding Ontologies with Social Processes and Natural Language

Christophe Debruyne, Quentin Reul and Robert Meersman
Semantics Technology and Applications Research Laboratory - Department of Computer Science
Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussels, Belgium
Email: {chrdebru,quenreul,meersman}@vub.ac.be

***Abstract*—In this paper, we present the GOSPL application that supports communities during the ontology engineering process by exploiting Social Web technologies and natural language. The resulting knowledge can then be transformed into RDF(S).**

***Index Terms*—Collaborative Ontology Engineering, Social Web Applications, Wiki Technology**

I. Introduction

The Linked Open Data (LOD) initiative[1] aims at linking data between existing resources using a scalable and standardized URI mechanism. This initiative will be a significant contribution to unlocking (legacy) data by publishing it with added semantic annotations on the World Wide Web. The annotation of existing conceptual models and content is a social process, which requires every member of a community to reach an agreement on how these should be linked. RDF(S) [3] is favored for its reusability due to its simple URI mechanism and limited semantics. The problem, however, is that IT laymen prefer to use *natural language* to describe their world.

The Social Web is characterized by large communities of human agents who participate and contribute through an online network to a collaborative activity, such as ontology engineering. An ontology is commonly defined as: *"a [formal,] explicit specification of a [shared] conceptualization"* [6]. In the past decade, Social Web technologies (e.g. wikis) have been developed to enable communities to reach an agreement and share knowledge about different subjects. Wikis can also be used to negotiate the subtle differences in the interpretation of a domain during the collaborative ontology engineering process.

As a result, we propose the GOSPL (Grounding Ontologies with Social Processes and natural Language) application enabling communities to develop and maintain ontologies by combining the DOGMA framework [8] with Social Web technologies. An important characteristic that makes the DOGMA framework different from traditional ontology approaches is that it is grounded on natural language. Thus, GOSPL provides a tool enabling communities (i) to contribute to the knowledge in an ontology based on natural language, (ii) to exploit Social Web technologies to reach a common agreement on the knowledge, and (iii) to support provenance by tracking changes to the ontologies. We have used GOSPL in the TAS[3] project[2] to allow end users (e.g. security and privacy experts) to easily develop conceptual models on the domain of security and privacy. The ontology developed with the GOSPL framework are being used to provide security policy interoperability within the TAS[3] project [9].

II. Related Work

Wiki technology has been adapted in the field of ontology engineering to enable non-technical users to create, visualise and maintain ontologies (OntoWiki [2], WikiDB[3]) or to semantically annotate the content (Semantic MediaWiki [7], DBPedia [1]).

III. GOSPL

Objectives (1) and (2) are covered by adapting a MediaWiki to fit the DOGMA Approach to ontology engineering with its grounding in linguistics. While objective (3) is implemented and integrated within the MediaWiki's history page.

A. Groundings in Natural Language

In RDF(S), one uses URIs to identify concepts, whereas humans use terms (in a certain context). RDF(S) have been built to provide a machine readable format, even though `rdfs:label` and `rdfs:comment` exist to provide human readable terms and descriptions for humans. The major problem of RDF(S) is the difficulty to think in terms of URIs or to read existing URIs.

In GOSPL, knowledge is stored as lexons, which are 5-tuples declaring a bi-directional relationship in some context γ, e.g., (γ, `Person, currently at, has, Location`). With GOSPL as a basis for modelling RDF(S) for LOD, we not only have a linguistic grounding in which users can more easily render their conceptualization in terms of senctences, but we also have access to the inverse relations between to concepts. These can be used to define backlinks in the RDF(S), which are redundant, as these triples can be "derived", but it allows browsers and crawlers to traverse links in either direction. Note that even though these inverse triples can be derived, their groundings in linguistic are not always explicit; whereas in GOSPL the co-role must be defined whenever a concept plays a role with another concept.

[1]http://linkeddata.org

[2]The research leading to these results has received funding from the EC's FP7 programme under grant agreement number 216287 (TAS[3] - Trusted Architecture for Securely Shared Services).

[3]http://www.mediawiki.org/wiki/Extension:WikiDB

B. GUI and Visualization

Compared to existing wiki-based ontology engineering tools, GOSPL provides a graphical modeling language to represent concepts and the relations between them. In GOSPL we use NORM-Trees [10], which are undirected rooted trees constructed by concatenating lexons and are also used in the GOSPL wiki. We make a distinction between the formal model and free text for both viewing and editing the model. We have integrated a GUI for managing the lexons for the latter.

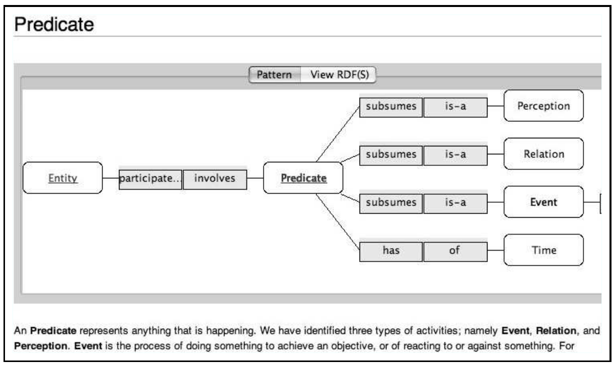

Figure 1. Viewing the model and content of a semantic object.

C. Provenance

In GOSPL, we keep a record of changes on two levels: one for the model and a second for the textual content. Separating the evolution of the formal and the non-formal content has as an advantage that one can apply heuristics to the history of the formal model (e.g., set-operations on the facts of two different versions). We use the timestamp to analyze the relationship between the two evolutionary processes of a semantic object.

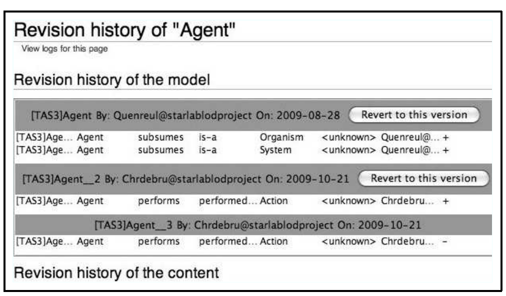

Figure 2. Revision history of model and textual content of a semantic object.

IV. Transforming Lexons into RDF(S)

Algorithm 1 shows how the models created by the user can be easily transformed into RDF(S) by generating classes, properties and their respective hierarchies from the lexons. The generated RDF(S) can then be reused by other users to publish their data on the Web, which is encouraged by LOD.

V. Semantic Object Interlinking

In our approach we opted to capture the semantics in a separate object linked to a wiki page rather than storing them within a page, i.e., the lexons are stored on the DOGMA server and manipulated through the wiki using the page's URI. These lexons can then be consulted though other tools, such as DOGMA Studio Workbench [4]. The reason for this is twofold: to avoid creating a page for each term, as this would be often overkill for some lexical concepts such as `Name`, `E-Mail`, etc and to enable less formal savvy persons to engage in the ontology engineering process with free text.

Algorithm 1 Transforming a commitment to RDF(S)

for all term in commitment **do**
  if term is a value-type **then**
    Create a new class that extends from rdfs:Literal
  else
    Create a new class that extends from rdfs:Class
  end if
  Add term as label for that new class
end for
for all lexon in commitment **do**
  Create property $p_1 : Term_1 \rightarrow Term_2$ for the role
  Create property $p_2 : Term_2 \rightarrow Term_1$ for the co-role
end for

VI. Future Work

GOSPL is still in a preliminary stage and work still needs to be done on the reuse of existing fact types by integrating the work in [5] and provide import functionality for existing ontologies. We will also investigate means to aid users in reusing the semantic objects in their commitments as well as providing functionality to publish one's data on the Web using those commitments, even go as far as retrieving data from several databases at the same time.

References

[1] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. Dbpedia: A nucleus for a web of open data. In *ISWC+ASWC 2007*, pages 722–735. Springer, 2008.

[2] S. Auer, S. Dietzold, and T. Riechert. Ontowiki - a tool for social, semantic collaboration. In *ISWC 2006*, pages 736–749. Springer, 2006.

[3] D. Brickley, R. Guha and B. McBride. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, 2004. http://www.w3.org/TR/rdf-schema/

[4] P. De Leenheer and C. Debruyne. DOGMA-MESS: A tool for factoriented collaborative ontology evolution. In *OTM 2008 (ORM 2008)*, pages 797–806, Springer 2008

[5] C. Debruyne, P. De Leenheer, and R. Meersman. A method and tool for fact type reuse in the dogma ontology framework. In *OTM 2009 (ODBASE 2009)*, pages 1147–1164, Springer 2009.

[6] T. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 6(2):199–221, 1993.

[7] M. Krötzsch, D. Vrandecic, M. Völkel, H. Haller, and R. Studer. Semantic wikipedia. *J. Web Sem.*, 5(4):251–261, 2007.

[8] R. Meersman. Semantics ontology tools in information system design. In *ISMIS 1999*, volume 1609, Springer, 1999.

[9] Q. Reul, G. Zhao, and R. Meersman. Ontology-based Access Control Policy Interoperability. In Proceedings of the 1st *International Conference on Mobility, Individualisation, Socialisation, and Connectivity* (MISC 2010).

[10] D. Trog, J. Vereecken, S. Christiaens, P. De Leenheer, and R. Meersman. T-lex: A role-based ontology engineering tool. In *OTM 2006 (ORM 2006)*, pages 1191–1200. Springer, 2006.