Mapping Cognitive Models to Social Semantic Spaces – Collaborative Development of Project Ontologies

Thomas Riechert[1], Steffen Lohmann[2]

[1]University of Leipzig
Department of Computer Science
Johannisgasse 26, 04009 Leipzig, Germany
riechert@informatik.uni-leipzig.de

[2]University of Duisburg-Essen
Dep. of Informatics and Applied Cognitive Science
Lotharstrasse 65, 47057 Duisburg, Germany
lohmann@interactivesystems.info

Abstract: In this paper, we present an approach that applies concepts from the areas of Social Software and Semantic Web to application development. We start with a short introduction into Semantic Based Requirements Engineering. Then, we present an ontology for capturing requirements and related information. Based on this ontology, we describe how stakeholders are enabled to directly participate in requirements elicitation and to collaboratively create a project ontology. Finally, we report about the application of the presented approach in a use case from the e-government domain.

1 Introduction

The goal of software development is to build applications that meet the users' needs. Involving all relevant stakeholders is therefore a crucial part of the development process. However, every stakeholder has its own view and expectations on the application to be developed. The creation of a shared understanding and terminology is often very challenging. Requirements engineering methods, ranging from heavyweight [Roy87] to agile [Bec00], aim to face these problems. Their major task is to model the application on an abstract level and from different points of view. However, these models cannot be directly created by the stakeholders. In general, the stakeholder statements are translated in the vocabulary of the software designers before aggregated in the respective models.

Semantic Web and Social Software open up new opportunities to cope with these difficulties. Within the SoftWiki research project[1] we develop a web based collaborative environment that fosters direct stakeholder participation in early stages of requirements engineering. The SoftWiki philosophy follows the notion of the Social Semantic Web: Participation should be as easy as possible and semantically structured at the same time.

[1] http://softwiki.de

This paper presents two main results of our research: a semantic model for requirements engineering and a method to directly involve stakeholders in the collaborative development of project ontologies.

2 Semantic-based Requirements Engineering

Ontologies attracted significant attention in software engineering recently. Developments such as the W3C's Ontology Driven Architecture (ODA)[2] or the OMG's Ontology Definition Metamodel (ODM)[3] testify the growing interest for ontology-based approaches in software engineering and related disciplines. The advantages of ontology-based approaches compared to conventional ones, however, often remain unclear. To motivate the use of ontologies, one can argue with the vision of the Semantic Web and its need for a comprehensive ontological grounding, but this would presumably not convince many companies to invest in ontology construction. Another argument is the facilitation of automation in subsequent development phases when basing the requirements on an ontological structure.

Besides these arguments, we regard an underlying ontological structure as highly valuable for integrating distributed stakeholders in the requirements engineering process and for building consensus among them. A semantically unambiguous, well-structured means to represent shared views and prevent (costly) misunderstandings is crucial – in this regard, ontologies have their strength. Ontologies may serve as a suitable *Interlingua* between the stakeholders and system designers. Requirements engineering would then consist partly in ontological engineering, but with the particular and challenging constraint that the ontological structure is understandable and usable by various stakeholders with restricted or no backgrounds in ontology construction.

3 SoftWiki Ontology for Requirements Engineering

In order to semantically support the requirements engineering process we developed the SoftWiki Ontology for Requirements Engineering (SWORE) in accordance with standards of the requirements engineering community [Poh07, HPW98, Lam01]. Figure 1 visualizes the core of SWORE. Central to our approach are the classes *Stakeholder* and *Abstract Requirement* as well as the properties *details* and *defines*. Abstract requirements have the subclasses *Goal*, *Scenario* and *Requirement*, each of which are defined by stakeholders and can be detailed by other abstract requirements. This enables the specification of requirements at different levels of granularity. We emphasize the collaborative aspects of requirements engineering by integrating discussions amongst the stakeholders and voting (with the criteria of agreement and importance) in the model. This documentation is often relevant for future decisions in the requirements engineering process.

[2] http://www.w3.org/2001/sw/BestPractices/SE/ODA/060211/

[3] http://www.omg.org/ontology/

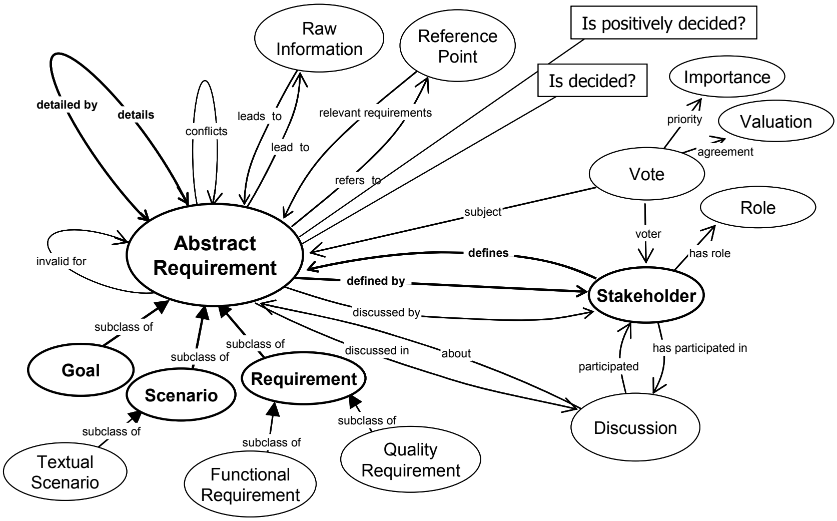

Figure 1: Visualisation of the SWORE core

The use of SWORE is visualised in Figure 2. A requirements engineering knowledge base contains instantiations ("is a") of the SWORE concepts. Furthermore, every abstract requirement refers to one or more concepts of the project ontology via its reference point.

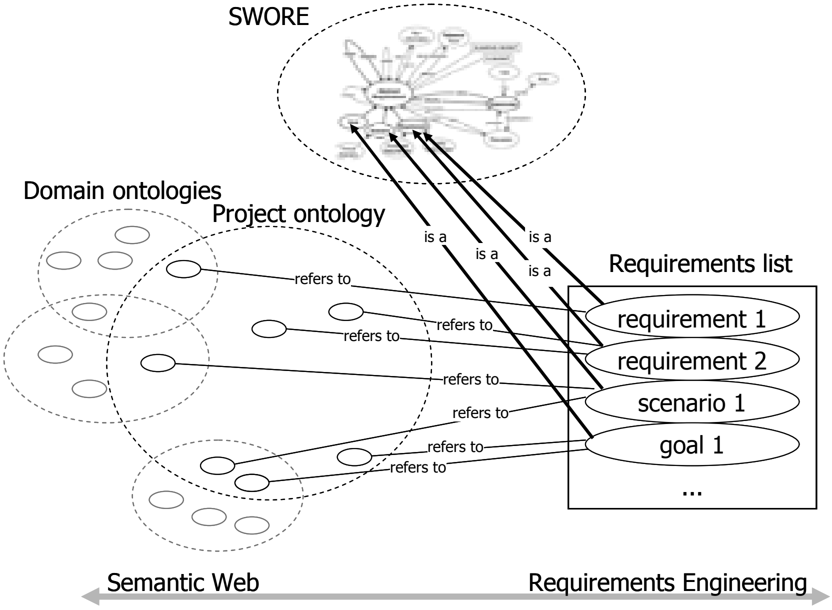

Figure 2: SWORE use and interlinking with the project ontology

The following example of a functional requirement illustrates the interlinking with the project ontology.

"The e-mail application should load messages from several email providers."

The stakeholder that formulates this requirement may assume that the model should contain concepts such as *message* and *email provider*, while another stakeholder may assume the model contains a concept such as *data transfer* (cp. Figure 3).

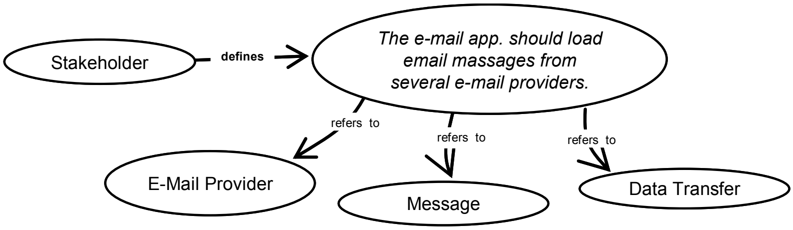

Figure 3: Requirement and links to instances of the class *Reference Point* equal to *owl:Thing*

The referenced concepts form the basis of the project ontology. Some of the concepts may also be parts of general domain knowledge that is already available in the Web[4]. This way, the requirements are enriched with further semantics. The project and domain ontologies provide useful semantic contexts to the stakeholders. This is illustrated by the following example of a requirement:

"The email application should support the most common data transfer protocols."

Stakeholders may refer this requirement to the concepts *data transfer* or *POP3 protocol.* Accessing an ontology that contains knowledge on the topic of email technologies, further concepts may be retrieved that are semantically related to the referenced concepts such as *IMAP protocol.*

4 Collaborative Development of Project Ontologies

The described method of collaboratively assigning reference points to requirements can also be regarded as a semantically enriched form of social tagging. As illustrated in Figure 4, additional relationships emerge in this tagging process (cp. [Sa06, Ma06]). On the one hand, relations between requirements evolve as two or more requirements get tagged with the same reference points (semantic tags). On the other hand, stakeholders become linked when using the same reference points (tag based relations) or tagging the same requirements (requirement based relations).

[4] This domain knowledge could have been created with the help of semantic extraction algorithms such as it is the case in the SoftWiki-related DBpedia project where formal knowledge is extracted from Wikipedia contents [AL07].

The reference points provide a valuable overview on the requirements and enable their exploration from different angles. Following the model of "perspective making and perspective taking" [BT95], the individual is faced with other stakeholders' views on the planned application that are expressed by the reference points they assigned to the requirements. New insights or categorization schemes may result that positively effect the requirements engineering process. Moreover, reference points used by the stakeholders may indicate their expertise with respect to a specific topic („I tag, therefore I know" [JS06]). Consequently, these concepts can serve as a starting point for fruitful discussions about the application in mind and might foster knowledge exchange and requirements refinement. Finally, the collaboratively created reference points provide a consolidated conceptual basis that can facilitate further software development. Particularly with regard to software design, some of the reference points may indicate components, technologies, or features that should be part of the planned application.

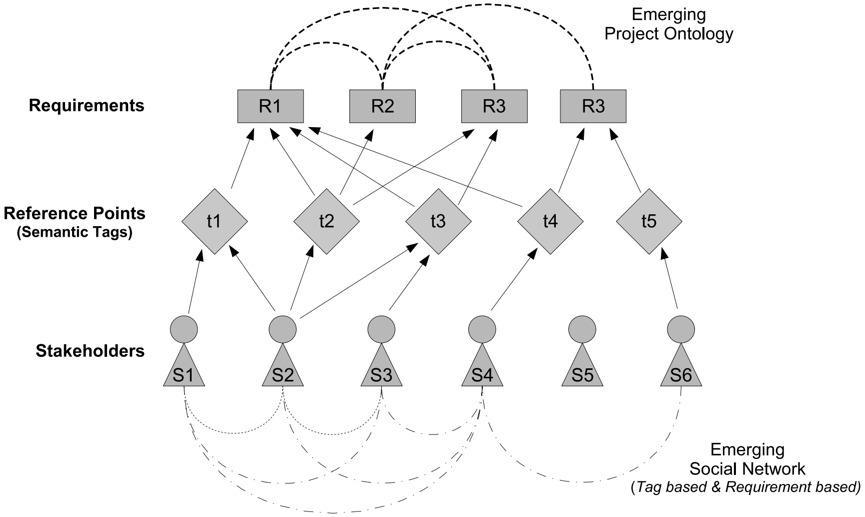

Figure 4: Collaborative semantic tagging of requirements by stakeholders

5 E-Government Use Case

An application and evaluation of the presented approach is currently performed in an e-government use case. The situation under consideration is characterized by a multiplicity of applications that have been developed and are in use by individual administration offices. In order to facilitate automated data exchange between different community administrations and to provide comprehensive web-based citizen services there are endeavours to homogenize and better integrate these individual applications on a local, regional and federal level. This is not just a technical problem, since different community administrations established different processes and information structures and hence have different requirements regarding information integration and exchange.

In order to enable the different stakeholders to collaboratively work on requirements for standardization, information and process integration we employed the described approach of Semantic Based Requirements Engineering. We adopted the semantic collaboration platform OntoWiki[5] [ADR06] for this purpose. OntoWiki enables a distributed user community to develop ontologies and collect instance data adhering to those ontologies. It aims at making browsing of knowledge bases and user contribution as easy as possible.

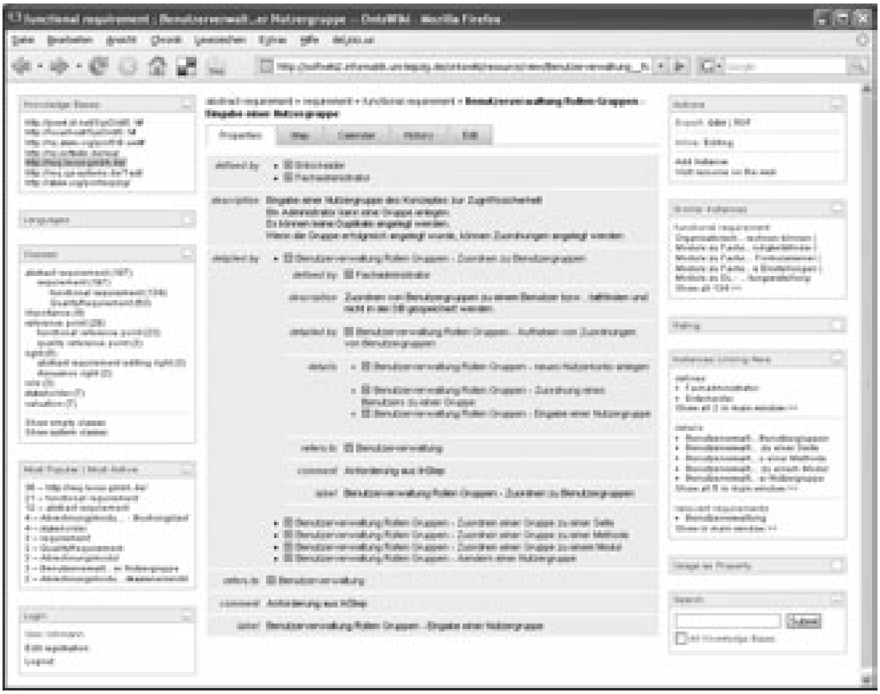

Figure 5: Functional requirement instance related to user roles and access control of the e-government application

In order to use OntoWiki for requirements elicitation we pre-loaded the SWORE ontology into this platform. The existing requirements have been imported directly from the prior used software management tool. Stakeholders are now enabled to create and interlink requirements and scenarios. Furthermore, they can vote, discuss, and comment individual pieces of information. Figure 5 shows a functional requirement instance related to user roles and access control of the e-government application.

The initial project ontology has been created in a combined method: (1) by an analysis of the site map of the existing web application, (2) by the use of concept extraction methods based on statistical text mining procedures [Bö02].

[5] OntoWiki is open source software and available at http://ontowiki.net.

Figure 6 shows a tag cloud visualization [KL07] that has been generated from the reference points of the use case. A reference point's font size represents its usage popularity. Though this visualization type has limited expression capabilities, it provides a good impression of major concepts with respect to the project and the application that is to be developed. It furthermore facilitates the exploration and navigation of the requirements: simply by clicking on a tag (reference point) in the cloud, a stakeholder gets a list of all associated requirements.

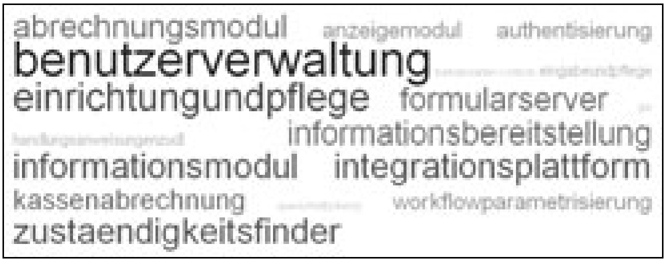

Figure 6: Tag cloud visualization of the reference points

6 Conclusion and Future Work

We consider a combined approach based on Semantic Web and Social Software concepts as beneficial for requirements engineering, since a crucial part consists in the collaborative development of a common terminology and shared understanding between different stakeholders. The presented approach tries to facilitate stakeholder participation and to foster the development of a conceptual basis. Similar to the notion of Wikis and Social Tagging, stakeholders are enabled to explore and expand the project ontology from different angles leading to new insights and a better understanding regarding the planned software application. Our current activities include further investigation of the e-government use case and the preparation of two additional use cases in e-commerce and geographic information system (GIS) domains.

Literaturverzeichnis

[ADR06] Auer, S.; Dietzold, S.; Riechert, T.: OntoWiki – A Tool for Social, Semantic Collaboration. In: Proceedings of the 5th International Semantic Web Conference, Springer, 2006; pp. 736–749.

[AL07] Auer, S.; Lehmann, J.: What have Innsbruck and Leipzig in Common? Extracting Semantics from Wiki Content. In: Proceedings of the European Semantic Web Conference, ESWC2007, Springer, 2007; pp. 503-517

[Bec00] Kent Beck. Extreme Programming Explained – Embrace Change. Addision Wesley, 2000.

[Bö02] Böhm, K.; Heyer, G.; Quasthoff, U.; Wolff, C.: Topic Map Generation Using Text Mining. In: Journal of Universal Computer Science, 8(6), 2002; pp. 623-643

[BT95] Boland J.R., Tenkasi R.V.: Perspective Making and Perspective Taking in Communities of Knowing. In: Organization Science, 6(4), 1995; pp. 350-372

[KL07] Kaser, O.; Lemire, D.: TagCloud Drawing: Algorithms for Cloud Visualization. In: Proceedings of WWW 2007 Workshop on Tagging and Metadata for Social Information Organization, 2007.

[JS06] John, A.; Seligmann, D.: Collaborative tagging and expertise in the enterprise. In: Proceedings of WWW 2006 Workshop on Collaborative Web Tagging, 2006.

[Lam01] Van Lamsweerde, A.: Goal-Oriented Requirements Engineering: A Guided Tour. In: 5th IEEE International Symposium on Requirements Engineering (RE'01), IEEE Computer Society Press, 2001; pp. 249-263

[Ma06] Marlow, C.; Naaman, M.; Boyd, D.; Davis, M.: HT06, tagging paper, taxonomy, Flickr, academic article, to read. In: Proceedings of the 17th Conference on Hypertext and Hypermedia, 2006; pp. 31-40

[HPW98] Haumer, P.; Pohl, K.; Weidenhaupt, K.: Requirements Elicitation and Validation with Real World Scenes. IEEE Transactions on Software Engineering, 24(12), 1998; pp. 1036-1054.

[Poh07] Pohl, K.: Requirements Engineering – Grundlagen, Prinzipien, Techniken. Dpunkt, 2007.

[Roy87] Royce, W.W.: Managing the Development of Large Software Systems: Concepts and Techniques. In: Proceedings of the 9th International Conference on Software Engineering, 1987; pp. 328-338.

[Sa06] Sack, H.: Kollaborative Indexierung und die Emergenz neuer sozialer Netzwerke. In: Workshop Social Software in der Wertschöpfungskette, 2006