Collaborative Development of Knowledge Bases in Distributed Requirements Elicitation

Steffen Lohmann[1], Thomas Riechert[2], Sören Auer[2], Jürgen Ziegler[1]

[1]University of Duisburg-Essen
Department of Informatics and Applied Cognitive Science
Lotharstrasse 65, 47057 Duisburg, Germany
{lohmann, ziegler}@interactivesystems.info

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

Abstract: One of the main challenges in distributed software development is the elicitation and management of knowledge regarding system requirements. Due to spatial distribution of involved parties, many limitations concerning interaction, communication, and conceptualization have to be faced. The *SoftWiki* project aims to provide an agile, wiki-based methodology to overcome these limitations in part. This paper introduces the *SoftWiki* approach and presents some of the tools that are developed to support knowledge sharing in distributed requirements engineering.

1 Motivation

Knowledge-intensive activities in distributed software development such as requirements, configuration, and change management are not sufficiently covered by existing knowledge management solutions thus far [KT05]. Particularly in settings where knowledge is spread over diverse participants, established methodologies and tools mostly fail to adequately support collaboration. However, a consolidated conceptualization of the domain in focus and a shared understanding of the planned software product are crucial to the success of the project [HL01].

The aim of the cooperative research project *SoftWiki*[1] is to support these activities, in particular with respect to system requirements. Potentially very large and spatially distributed stakeholders shall be enabled to collect, semantically enrich, classify and aggregate knowledge regarding requirements. Attempts from the domains of software engineering, knowledge management, and the semantic web are combined in a comprehensive framework. In the following, we describe preliminary results of the project, particularly concerning the developed solutions to support agile knowledge sharing.

[1] Research project funded by the German Federal Ministry of Education and Research, see: http://softwiki.de

2 Collaborative Development of a Requirements Knowledge Base

Figure 1 illustrates the overall SoftWiki architecture. The interaction between system users and developers is at the heart of the approach. Users are enabled to directly participate in the development of knowledge bases for system requirements, whereas developers support the users and moderate the collaborative activities. The effort and formal overhead for stating or modifying some piece of knowledge is minimized due to the adoption of the wiki paradigm [LC01] and its underlying philosophy of 'making it easy to fix mistakes, rather than making it hard to make them'. Stakeholders can collaboratively externalize their knowledge in natural language without any formal restrictions. Additional semantics are added by linking the contents with ontological classes and instances.

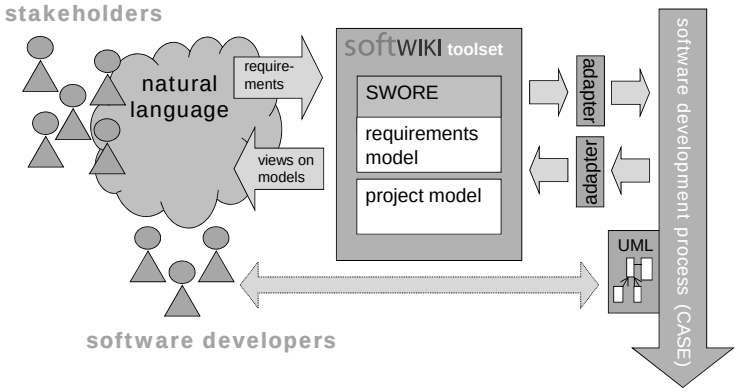

Figure 1: General SoftWiki architecture

The SoftWiki knowledge repository consists of different modeling perspectives. Essentially, these are the requirements and the project model, with the latter containing knowledge about the domain in focus, the product itself, and its context. In order to facilitate participation of various stakeholder groups, the knowledge repository can be accessed in several ways. Further knowledge sources for requirements such as relevant documents or system descriptions can be included via appropriate adaptors.

It is not the goal of the SoftWiki project to replace established CASE tools with a wiki-based approach. It rather aims to support agile collaboration in early stages of distributed requirements engineering that are not sufficiently covered by available tools thus far. Hence, the collaboratively created knowledge base can be made available to further development activities via specific adaptors, at best resulting in a 'knowledge flow' that constantly accompanies the software development cycle.

The underlying semantic structure is provided by the *SoftWiki Ontology for Requirements Engineering* (*SWORE*) that has been developed in accordance with established practices of the requirements engineering community (see [Ri07] and Figure 2 for an illustration of the SWORE core). It provides an ontological schema for

instances of the types *goal*, *scenario* and *requirement*. These instances have relations of the type *defined by* to *stakeholders* and are interconnected by the relations *details* and *detailed by*. In addition, SWORE offers the opportunity to link requirements with domain concepts or parts of already existing system models by using relations of the type *refers to* (see [RL07]).

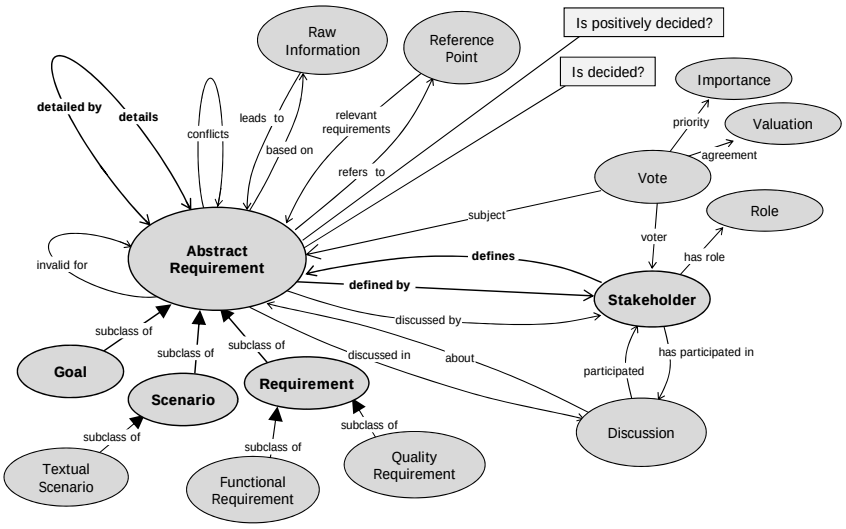

Figure 2: SoftWiki Ontology for Requirements Engineering

3 Applying a Semantic Wiki to Requirements Engineering

Within the SoftWiki project we adopt and extend the semantic wiki *OntoWiki* [ADR06] enabling it to support knowledge management regarding system requirements[2]. As OntoWiki is completely web based, it can be accessed from any location without installation simply by using a web browser.

Figure 2 presents a screenshot of OntoWiki as it is currently used to manage requirements. The user interface is divided into three areas. The left area provides features for selection of knowledge bases and ontological classes. In this case, the knowledge base of an e-government use case (“http://req.lecos-gmbh.de/fabius/”) has been selected that corresponds to the SWORE ontology schema (“http://ns.softwiki.de/req/”). Accordingly, the corresponding class structure is given showing in brackets the number of instances for each class in this knowledge base. Classes without instances are hidden from the user. For instance, the presented use case does not have any instances of the type *goal* or *scenario*. The middle area provides several views on the instances and

[2] *OntoWiki* is based on the ontology authoring and management framework *Powl*. It is open source and available for download at: http://powl.sf.net. A demo installation including some knowledge bases can be accessed at: http://demo.ontowiki.net/.

attributes as well as inline editing possibilities. In the screenshot, a functional requirement instance related to user roles and access control concerning the e-government application is presented. Additionally, its attributes *description*, *comment* and *label* as well as its relations *defined by*, *detailed by* and *refers to* are displayed. Instances that are related to the selected one are shown in more detail by clicking on the plus icons. Other views can be selected via tabbed navigation, such as a map, a calendar, or a view on the change history. The right area provides additional, context-sensitive features including search, similar instances, and rating options.

Figure 3: A knowledge base for requirements of an e-government use case managed in the *OntoWiki* tool

By using this environment, stakeholders are enabled to create and interlink knowledge concepts, requirements, goals, and scenarios. Furthermore, they can vote, discuss, and comment pieces of content. The integrated versioning and change history features facilitate traceability when collaboratively working on a knowledge base. In order to enrich the requirements with semantics, externally available domain knowledge can easily be referenced, too. One source for structured knowledge might be domain ontologies that are extracted from *Wikipedia*[3] – as it is subject of investigation in the SoftWiki-related *DBpedia* project [AL07].

Although the user interface is designed according to the wiki principle, early user studies showed that it is not immediately understandable and too generic in some regards. Therefore, we are currently working on a redesign of the interface and general appearance of the application to ease interaction and better meet the users' needs in requirements elicitation. Not all users can, however, be reached with a wiki-based approach, no matter how well the user interface is designed. This is particularly true for end-users that are not actively participating in requirements engineering for the most part. Nonetheless, as end-users normally are directly affected by the implementation of the requirements, their feedback can be crucial for the success of the project [Ku05, HL01]. This is one reason why we are currently extending the wiki-based environment in various directions in order to provide additional support for specific requirements elicitation activities and stakeholder groups.

4 Extensions of the Wiki-based Approach

Due to the continuous application of semantic web technology, the wiki-based approach can be seamlessly extended by decentralized communication and participation channels. The left screen in Figure 4 shows a part of a tool called *Softfox* that can be easily integrated into the web browsers of users. It has been realized according to the paradigm of social tagging [Ma06] and allows gathering user feedback along with information regarding the usage context. Users are encouraged to add freely chosen tags to their input that can later assist in analyzing the user feedback and developing a conceptualization of the requirements domain.

Another extension called *SW-Analytics* supports the integration of knowledge that has been elicited decentrally (see right screen in Figure 4). It facilitates multidimensional visual exploration and preselection of relevant contents in accordance with concepts that have been defined in the SWORE ontology. The current implementation considers geospatial and tagging data enabling developers to browse stakeholder contributions based on the location where they have been stated or the tags that have been added. This way, requirements can be explored with respect to a specific purpose – for instance, a developer may want to view only those requirements that have been stated at a specific company site.

[3] http://wikipedia.org

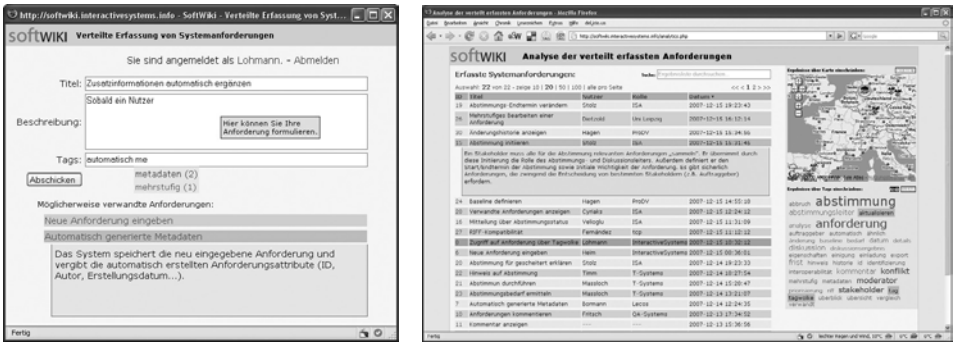

Figure 4: Extensions for decentralized participation and visual exploration

In order to facilitate semantic interoperability with other tools, the collaboratively developed knowledge base can be exported in RDF-format according to the schema of the SWORE ontology. Alternatively, parts of the knowledge base can be accessed via a SPARQL endpoint. Moreover, we are currently working on an adapter that enables export in RIF[4]-format to integrate the SoftWiki approach with established requirements and project management tools (in our case: *IRqA*[5]). However, we do not intend to fully embed the wiki-based environment itself, but rather try to link the collaboratively developed knowledge base with other development activities (see above).

Further extensions that are currently under development include the integration of semi-automatic knowledge extraction and classification mechanisms as well as additional visualizations such as network graphs of the knowledge and requirements structure.

5 Conclusions and Further Work

First experiences with use cases of the project indicate that an agile, wiki-based methodology can provide valuable assistance for distributed requirements engineering. Related work (e.g. [De07, Lo06]) supports this general conclusion. Furthermore, the application of a semantic wiki in combination with an ontological schema seems to foster terminological accuracy and shared understanding among the participants. In addition, it allows for semantic interoperability and automated utilization of the collaboratively developed knowledge base. In order to improve development assistance and meet the needs of various stakeholder groups, we addressed some extensions of the wiki-based approach that we are currently working on.

Furthermore, we consider integrating the SoftWiki approach with on-demand software development. As we experienced with COfundOS[6], a web platform for on-demand development and funding of open source software, sophisticated and usable means to

[4] Requirements Interchange Format: http://www.automotive-his.de/rif/doku.php
[5] http://www.qa-systems.de/html/deutsch/produkte/irqa
[6] http://cofundos.org

handle requirements are crucial in order to make this type of software development possible. Semantic wiki based management of requirements knowledge seems promising in this regard.

Generally, the design of any solution for web based collaboration of stakeholders needs to be easy to use in order to gain high participation. At the same time, it has to be powerful enough to enable structured access as well as efficient analysis and post-processing of requirements knowledge. Furthermore, the solution should be as generic as possible on the one hand and provide sufficient support on the other. Balancing these conflicting demands is a key issue to any approach that aims to assist in distributed requirements engineering.

Literaturverzeichnis

[ADR06] Auer, S.; Dietzold, S.; Riechert, T.: OntoWiki – A Tool for Social, Semantic Collaboration. In: Proceedings of the 5th International Semantic Web Conference, 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 4th European Semantic Web Conference, 2007; pp. 503-517.

[De07] Decker, B.; Ras, E.; Rech, J.; Jaubert, P.; Rieth, M.: Wiki-Based Stakeholder Participation in Requirements Engineering. In: IEEE Software, 24(2), 2007; pp. 28-35.

[HL01] Hofmann, H. F.; Lehner, F.: Requirements Engineering as a Success Factor in Software Projects. In: IEEE Software, 18(4), 2001; pp. 58-66.

[KT05] Komi-Sirviö, S.; Tihinen, M.: Lessons Learned by Participants of Distributed Software Development. In: Knowledge and Process Management, 12(2), 2005; pp. 108-122.

[Ku05] Kujala, S.; Kauppinen, M.; Lehtola, L.; Kojo, T.: The Role of User Involvement in Requirements Quality and Project Success. In: Proceedings of the 13th IEEE International Conference on Requirements Engineering, 2005; pp. 75-84.

[LC01] Leuf, B.; Cunningham, W.: The Wiki Way: Collaboration and Sharing on the Internet. Addison-Wesley, Reading, 2001.

[Lo06] Louridas, P.: Using Wikis in Software Development. In: IEEE Software, 23(3), 2006; S. 88-91.

[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.

[RL07] Riechert, T.; Lohmann, S.: Mapping Cognitive Models to Social Semantic Spaces – Collaborative Development of Project Ontologies. In: The Social Semantic Web 2007 – Proceedings of the 1st Conference on Social Semantic Web, 2007; pp. 91-98.

[Ri07] Riechert, T.; Lauenroth, K.; Lehmann, J.; Auer, S.: Towards Semantic based Requirements Engineering. In: Proceedings of the 7th International Conference on Knowledge Management, 2007.