Collaborative Modeling with Semantic MediaWiki

Frank Dengler
Institute AIFB, Karlsruhe Institute of Technology
Karlsruhe, Germany
frank.dengler@kit.edu

Hans-Jörg Happel
FZI Research Center for Information Technology
Karlsruhe, Germany
happel@fzi.de

ABSTRACT

Modeling is an important aspect of information system development, allowing for abstract descriptions of systems and processes. Therefore, models are often characterized as communication artifacts between different stakeholders in a development process. However, modeling as such has turned out to be a specialist activity, requiring skills in arcane modeling languages and complex tools.

In this paper, we suggest and present an approach for collaborative, Wiki-based modeling of process models and UML (class-)diagrams. While other web-based "lightweight" modeling tools are available, our approach consequently follows the Wiki-paradigm and allows us to semantically process the modeled information building upon Semantic MediaWiki.

Categories and Subject Descriptors

H.5.3 **[Information Interfaces and Presentation]:** Group and Organization Interfaces - *Computer-supported cooperative work, Theory and models*

General Terms

Documentation, Design, Human Factors

Keywords

Semantic Wikis, Process Models, UML, Modeling

1. INTRODUCTION

During the evolution of information system development, models have become increasingly important artifacts. In the past decades, many modeling languages covering a broad range of domains including data, systems or process modeling have been specified.

In this context, models are often attributed with two major purposes. First, most models provide a formal abstraction, allowing for analysis such as consistency checking, simulation or the automatic generation of lower-level artifacts – such as in the Model-Driven-Development paradigm. Second, modeling languages are often complemented by a visual representation, which makes them popular for documentation and as an artifact facilitating discussion and exchange among different stakeholders.

However, a majority of modeling tools tends to be complex, fully-fledged applications, which require training and are not easily accessible for all stakeholders. This in turn limits their value throughout a development process, since it makes models hard and slow to change and often creates media gaps when transferred to other stakeholders – e.g. as graphics contained in documents.

A prime example is the domain of business process management, which requires specialists to capture process descriptions that are then compiled into extensive documentation. In rapidly changing business environments this often leads to de facto deviations from the "to-be" execution of processes [4], yielding problems in organizational efficiency and service quality.

Summarizing, awkward and heavyweight modeling tool-chains impede the fast implementation of changes and create barriers for collaboration and contribution. This makes modeling artifacts less accurate, reducing the potential benefits of modeling approaches as such.

To address this issue, web-based lightweight modeling tools have been created for various domains. Such tools offer a reduced feature set and focus on early stages of the modeling process, suggesting to use more heavyweight modeling environments in later stages. Notably, web-based modeling tools are independent applications not including typical Wiki functionality. On the other hand, some Wiki engines allow for embedding model markup in Wiki text, rendering e.g. flowcharts or UML diagrams.

2. (SEMANTIC) WIKI-BASED MODELING

In this section, we want to make our case for a collaborative modeling environment based on Wikis and semantic technologies. After introducing these prerequisites, we sketch our implementation of process and UML class diagram modeling using Semantic MediaWiki (SMW) [2].

2.1 Wikis and Modeling

Wikis can be characterized by various design criteria such as the absence of access rights and allowing anyone to openly edit content complemented by a version history allowing to easily recovering from erroneous edits. Another distinct feature is the conceptual integrity between page title and content of a Wiki page. This gives every topic an explicit URL, allowing information to easily accumulate on one dedicated page (avoiding redundancy) and for easily linking Wiki page from one to another.

In terms of modeling, this allows whole models but also individual model elements to be represented by individual Wiki pages. This makes them easy to reference and can thus help collaborators to faster store and retrieve relevant information.

2.2 Semantic Wikis and Modeling

Semantic Wikis such as SMW leverage the fact that Wiki pages – in the ideal case – represent a set of interlinked atomic concepts as just described. This basically materializes in a) interpreting Wiki page categories as conceptual classes, b) allowing to explicitly specify the semantic meaning of links between Wiki pages and c) allowing to attach basic data properties to Wiki pages.

Consolidation of this information enables users to easily build and maintain a structured knowledge base which can then expose powerful query and export mechanisms on the underlying data. In the case of model artifacts, this nicely complements the Wiki-based approach, since it allows for semantically relating model elements and for capturing associated data in a structured way.

2.3 Modeling in Semantic MediaWiki

SMW is an extension to the popular MediaWiki engine, which provides basic semantic annotation features as described in the previous section. Using MediaWiki's template mechanism and another extension called Semantic Forms, SMW offers a user-friendly way to capture semantic properties and relations.

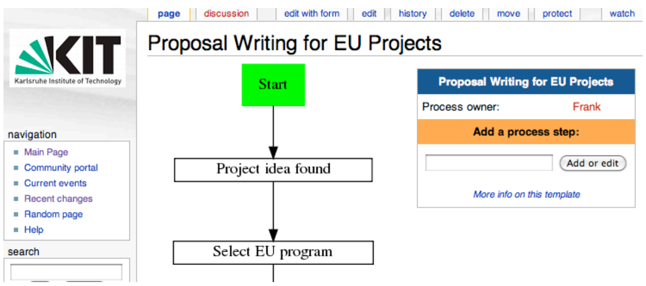

Figure 1: Example Process Graph in SMW

Semantic information annotated in this way is extracted into a knowledge base, which can be queried using a special query language. Queries can be embedded in Wiki pages to produce lists or tables of information. Furthermore, query results can be rendered in arbitrary formats using an extension mechanism. Examples for such "Semantic Result Formats" are data-oriented ones like vCard, RSS-Feeds or iCal or visualization-oriented formats like maps or timelines.

We have combined these simple mechanisms to create a SMW-based modeling approach for business processes [1] (Figure 1) resp. UML class diagrams (Figure 2) First, our approach consists of a set of predefined properties and classes for basic modeling primitives and corresponding structured forms. This includes e.g. a class "ProcessStep" and properties such as "hasSuccessor" for process modeling or "UMLClass" and "hasSuperclass" in the case of UML class diagrams. Using provided input forms, users can easily add and edit model elements such as process steps or UML classes.

Furthermore, we defined a set of templates including semantic queries that aggregate all elements belonging to a specific model. We provide a result printer transforming these related model elements into GraphViz markup language. By calling GraphViz, we can thus generate visual representations of model elements displaying a whole process model or UML class diagram.

3. SUMMARY

Existing modeling approaches are either heavyweight and thus not very responsive in quickly changing scenarios or lightweight and not formal enough to allow for meaningful applications.

With our process and UML class diagram modeling based on SMW, we have described a hybrid approach. Compared to lightweight approaches it allows for maintaining formal model representations. Compared to heavyweight approaches, it offers easy collaborative editing and a flexible, extensible data model which even allows for semantic links to other concepts in SMW.

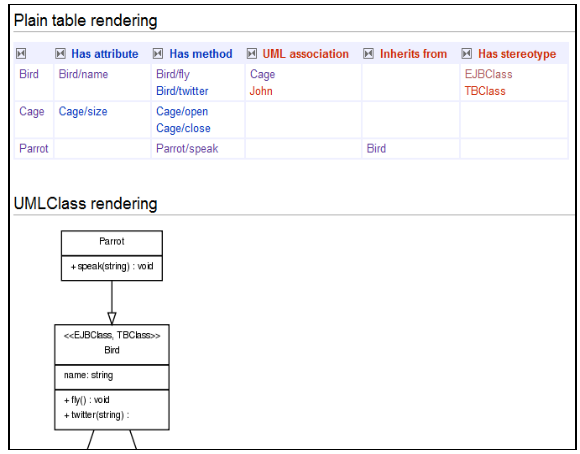

Figure 2: Example UML class diagram

Thus, our approach can provide semantic traceability – e.g. between business documents and process steps or system requirements and UML classes [3]. Due to its deep embedding in the Wiki, model elements in our approach also benefit from Wiki features such as collaborative description, discussion or change notifications.

We think that our approach can be especially beneficial in scenarios that do not require the full expressivity of modeling languages. It is probably more suitable for agile prototyping scenarios or documentation in dynamic, fast-changing environments. Thus, typical application areas could be global and/or inter-organizational collaboration projects, which do not share a common tool infrastructure. We also think that our approach could be helpful for training and teaching modeling, since it does not require individual tool installation.

Current shortcomings of our approach are the interoperability with other tools and dealing with complex models resp. the usability of modeling in general. While we want to address the first issue by import- and export-features for standard formats, we are investigating WYSIWYG editors for improving usability.

4. ACKNOWLEDGEMENT

This work is partially funded by the GlobaliSE project and the IST project ACTIVE (ICT-FP7-215040).

5. REFERENCES

[1] F. Dengler, S. Lamparter, M. Hefke, A. Abecker: Collaborative Process Development using Semantic MediaWiki. Proceedings of the 5th Conference of Professional Knowledge Management, Solothurn, Switzerland, März, 2009

[2] M. Krötzsch, D. Vrandecic, M. Völkel, H. Haller, R. Studer: Semantic Wikipedia. Journal of Web Semantics, 5, p 251-261, September, 2007

[3] W. Maalej, D. Panagiotou, H.-J. Happel: Towards Effective Management of Software Knowledge Exploiting the Semantic Wiki Paradigm. Proceedings of Software Engineering 2008, Munich, 2008

[4] R. Schmidt und S. Nurcan: BPM and Social Software. Business Process Management Workshops, 2009, p 649-658.