Extending the Makna Semantic Wiki to support Workflows

Karsten Dello, Lyndon Nixon, and Robert Tolksdorf

Freie Universität Berlin, Institut für Informatik,
AG Netzbasierte Informationssysteme, Takustr. 9, D-14195 Berlin, Germany
{dello,nixon,tolk}@inf.fu-berlin.de,
http://www.ag-nbi.de/

Abstract. Semantic wikis combine the advantages introduced by the wiki principle with the potential of Semantic Web technologies. However, there is still a very limited support for coordination, collaboration and integration in current semantic wikis. In this paper, we present a solution for this through the integration of our Makna semantic wiki with a workflow system. The resulting implementation is presented and an example given how this integration leads to better coordination, collaboration and integration support.

1 Introduction

Wikis have become popular tools for collaborative information management rooted in the principles of editability and collaboration : each member of the community can provide his knowledge, revise incorrect information and of course benefit from access to everyone else's knowledge. Semantic wikis are a natural extension of this, allowing for knowledge to be provided in some formal manner, e.g. using RDF, to improve the search for and retrieval of knowledge in the wiki system.

In this paper, we respond to a significant shortcoming in current semantic wiki systems in that they provide limited support for coordination, collaboration and integration despite the presence of semantics which can be used to improve such aspects. Particularly for corporate semantic wikis, it can be important to follow specific workflows in the collaborative editing of some wiki article, e.g. that a certain department must provide its input before another department takes over working in the wiki.

We present our proposal for the limited coordination, collaboration and integration of semantic wikis: combining a semantic wiki and a workflow management system. We have designed a model integrating the semantic wiki's data model with the WfMC process definition reference model (*http://www.wfmc.org/standards/referencemodel.htm*). Following an introduction to the semantic wiki used in this work, Makna, we describe its integration with the workflow engine jBPM. We evaluate the implementation towards a workflow example and conclude by considering the contribution of this work.

2 Integrating a semantic wiki with a workflow engine

Makna was conceived as a Wiki-based tool for distributed knowledge engineering (*http://makna.ag-nbi.de*). It extends an existing Wiki engine with generic, easy-to-use

ontology-driven components for collaboratively authoring, querying and browsing Semantic Web information. The architecture of Makna consists of the Wiki engine JSPWiki (*http://www.jspwiki.org/*), extended with several components for the manipulation of semantic data, and the underlying persistent storage mechanisms. A more detailed description of Makna and the user interfaces it supports can be found in [DPT06].

For this work we integrated the semantic wiki Makna and the workflow engine jBPM . The architecture of the distributed system with jBPM and Makna consists of a J2EE server, J2SE server(s), and a workstation and database servers that serve as persistence stores. In this paper we briefly consider some of the aspects of this integration which were implemented.

2.1 Semantic Workflow Annotation

It is necessary to reflect the workflow instances (e.g. tasks and processes) in the semantic model in order to support search for and enhanced presentation of tasks and processes. The insertion of workflow concepts into the semantic model at runtime has two prerequisites: first, a mechanism must be provided which assigns URIs to workflow instances; second, these URIs must be made accessible in jPDL process definitions.

With these prerequisites met, semantic decoration of workflow execution can be performed with standard jBPM procedures. While traversing a process graph the engine fires events – e.g. *process-start*, *process-end*, *task-start*, *task-end* and *task-assign* – which can be associated with custom actions. By associating these events with actions that perform SPARUL update queries based on URIs for workflow instances, the progress of a workflow can be reflected in the semantic model.

2.2 Semantic Flow Conditions

Facilitating semantically enhanced flow conditions in the process execution phase is desirable, because inference allows for transitions and control flow, based on the description of a workflow in the semantic model itself.

jPDL supports the association of transitions with boolean expressions which are based on workflow relevant data. The process execution continues via the first transition whose associated expression resolves to true, or via the default transition if none of the expressions resolves to true, though this behavior can be customized. Anyway, a straightforward approach is to query the semantic model with a SPARQL-ASK-Query, and store the result in a process variable which determines an expression that is associated with a transition. This procedure facilitates simple control of the process based on semantic model state.

2.3 Semantic Assignments

Strategies for the assignment of tasks to wiki users that build on semantic user and task descriptions should be supported by the system. Semantic user descriptions can be based on formalizations such as DOAC. Task descriptions are a bit more complicated because not only the general description of a task, which is valid for all task instances,

but also details of the current execution of a particular instance might be relevant for assignments. The description of users and tasks can use common concepts, e.g. by sharing domain ontologies. The actual assignment of tasks to users is realized through a jBPM *AssignmentHandler* implementation which can be configured with a SPARQL query that references these concepts.

2.4 Semantic Search and Presentation of Workflow Individuals

The structured presentation of semantic resources has been a lacking feature in Makna. To deal with this problem we have added support for formating SPARQL XML responses with XSLT. This functionality is encapsulated as another JSPWiki-Plugin, which additionally has limited support for expressions that are resolved at rendering time (e.g. logged in user and page URL). Via an endpoint parameter remote triplestores are also supported. The plugin enables the structured presentation of workflow individuals such as tasks and process instances in the wiki. It can be called from the JSPWiki template level (invoked from a JSP) or from the wiki page level (invoked from wiki syntax). Another application of this plugin in the generation of lists such as semantically enhanced task lists. The tasks that have been assigned to the logged in user can be arranged by domain specific structures, thus enabling semantically enhanced task lists.

3 Evaluation

In this section we present an example workflow to illustrate improvement of coordination, collaboration and integration support in our system.

3.1 Example Workflow

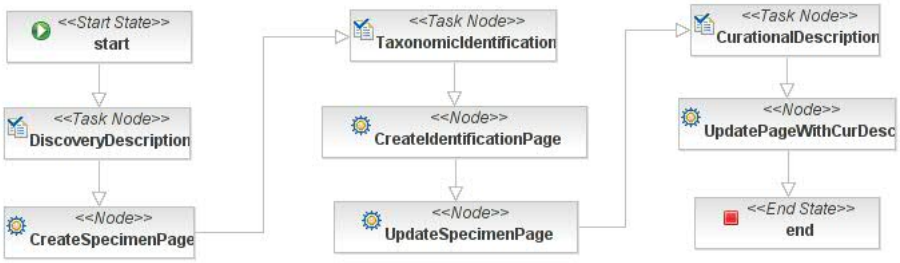

Fig. 1. Sample workflow for the processing of a new specimen. The three columns are associated with the process swimlanes *FieldWorkParticipant*, *Taxonomist* und *Curator*.

We have derived a couple of example workflows from a business model within the EDIT (European Distributed Institute of Taxonomy) project (*http://www.e-taxonomy.eu/*). Figure 1 shows a typical workflow: the processing of a specimen. It involves three

actors: a field trip participant who describes the finding of the specimen, a taxonomist who performs the taxonomical identification and a curator who is responsible for the curational administration.

First, somebody in the role of a *field work participant* triggers the start of the process [1] and describes the discovery of a new specimen through an associated task form in the userconsole, whereby a new wiki page is created from a template. This template contains enhanced wiki syntax with placeholders for subjects of statements, which are replaced with corresponding user input. User input consists of literals (e.g. from textarea and input fields) as well as URIs of concepts and instances (e.g. from JSF *SelectItem* elements). Predicates used in the template are taken from existing ontologies. [2]

Notification by email is sent to somebody in the *taxonomist* swimlane to inform him/her that he/she is due to perform a taxonomic identification of this specimen. Again, the results are committed to the workflow engine through a task form. Upon completion a new page with the results of the taxonomic identification is created, and a typed link to it is inserted on the discovery page. A last action must be taken by somebody in the role of a *curator*. The selection of the actor is formalized as a SPARQL query and executed upon creation of the task instance. It is based on semantic description of curators' responsibilities and the results of the taxonomist's identification.

3.2 Improved Coordination, Collaboration and Integration

Even though this short sample workflow makes only use of one workflow pattern – the sequence – it can be used to show that coordination, collaboration and integration support has been improved. The combination with a workflow system has enabled the coordination of interactions within a wiki system. The creation of a new wiki page in the node *CreateIdentificationPage* has been combined with the modification of another page in the node *UpdateSpecimenPage* which adds a link to the newly created page, thus realizing a coordinated modification of two wiki resources.

The collaboration of the participating actors is described in the workflow, thus enabling a controlled yet versatile modification of a semantic wiki resource. The chronological dependencies between the activities of the actors in the process roles *FieldWorkParticipant*, *Taxonomist* und *Curator* is reflected in the sequential flow of the process. Through task assignment notifications the taxonomist and the curator are informed by emails about their tasks which arise from other user's previous activities. Further on, the workflow assures that information which is required in a later step is present before the execution continues, thus facilitating an efficient collaboration. In this example this has been realized at user interface level by the use of JSF validators in the associated task forms, though it could also has been realized at process definition level by means of a jBPM *task controller*.

Because the execution of processes and tasks is reflected in the wiki's semantic model it is also possible to use hypertext navigation facilities between task, process

[1] Workflow related functionality (e.g. listing and starting of processes) is provided by JSPWiki plugins in Makna which use RMI to interact with the jBPM engine.

[2] In our example we reuse the FungalWeb ontology [SNBHB05] for mycological classifications and TDWG's LSID ontologies for taxonomic data.

and user resources. Further on, Makna's click-searches which are provided with every resource in the wiki can also be helpful for collaboration. Examples include a list of all users with tasks in a certain process and a list of all specimen that have been identified by a certain taxonomist.

Another aspect of the improvement of collaboration support is the selection of an actor in the curator swimlane based on the results of the taxonomic identification and the semantic responsibilities descriptions in the wiki. In this example the taxonomic identification requires the selection of a *FungalWeb* concept and responsibility descriptions of actors refer to the same ontology. Because of the hierarchical structure of the *FungalWeb* ontology it is possible to infer the responsible actor, thus realizing a dynamic assignment strategy.

4 Conclusion

We have considered in this paper how the limited coordination, collaboration and integration support of semantic wikis could be improved and have presented a solution based on integration with a workflow system. Initial evaluation of our implementation has demonstrated improvements in these aspects which, in the authors' view, can prove to be of importance to the future uptake of semantic wiki systems, particularly in the corporate environment.

We plan to continue the development of Makna, and particularly to explore in a practical manner the deployability of semantic wikis in enterprise environments through our Corporate Semantic Web research group (*http://www.corporate-semantic-web.de*), which has as one of its goals research in corporate semantic collaboration. Makna and its workflow system integration is a first product of our work on corporate semantic wikis, and through industrial partnerships we plan to test the usability of Makna in real business scenarios where the described coordination, collaboration and integration support is of great importance.

5 Acknowledgments

This work has been partially supported by the "InnoProfile-Corporate Semantic Web" project funded by the German Federal Ministry of Education and Research (BMBF) and the BMBF Innovation Initiative for the New German Länder - Unternehmen Region.

References

[DPT06] Karsten Dello, Elena Bontas Simperl Paslaru, and Robert Tolksdorf. Creating and using semantic web information with makna. In Max Völkel and Sebastian Schaffert, editors, *Proceedings of the First Workshop on Semantic Wikis - From Wiki To Semantics*, volume 206 of *Workshop on Semantic Wikis*, pages S.43–57, Budva, Montenegro, June 2006. ESWC2006.

[SNBHB05] Arash Shaban-Nejad, Christopher Baker, Volker Haarslev, and Greg Butler. The fungalweb ontology: Semantic web challenges in bioinformatics and genomics. In *Proceedings of the International Semantic Web Conference 2005*, volume 3729 of *Lecture Notes in Computer Science*, pages 1063–1066. Springer, 2005.