ACTIVE
FP7-ICT-2007-1
Contract no.: 215040
www.active-project.eu

ACTIVE

Deliverable 3.1.3

Context-aware knowledge filters
Fully functional prototype and evaluation

Editor:	Benedikt Kämpgen, Karlsruhe Institute of Technology (KIT)
Deliverable nature:	Prototype (P)
Dissemination level: (Confidentiality)	Public (PU)
Contractual delivery date:	1st March 2011
Actual delivery date:	1st March 2011
Suggested readers:	Semantic MediaWiki community
Version:	1.0
Total number of pages:	41
Keywords:	keyword search, visualization, process editing, Semantic MediaWiki

Abstract

For Task 3.1 in year three of the ACTIVE project, we have built on our work on knowledge filters, tools that provide users with familiar means to retrieve, refine, and inspect knowledge. In this final deliverable, we present and evaluate two extensions to Semantic MediaWiki: The first one – Ask The Wiki and its light-weight version AskQ – allows familiar and easy-to-use keyword search to exploit structured data. The second component – Process Editor, an extended version of Process Visualization – is inspired by graphical tools, and combines formal semantics, natural language and visual editing for process management by novice and expert users. Approached business needs are still of high relevance. Yet, when we deployed our solutions at our case study partners, we had to react on issues not anticipated before: For instance, Ask The Wiki was too heavy-weight, Process Visualization not usable by non-expert users. Resolving these issues lead our prototypes evolve into well-suited components for concrete business scenarios and further research.

Disclaimer

Impressum

[Full project title] ACTIVE – Knowledge-Powered Enterprise

[Short project title] ACTIVE

[Number and title of work-package] WP3 – Pro-active Knowledge Processes Support

[Document title] Context-aware Knowledge Filters – Fully functional prototype and evaluation

[Editor: Name, company] Benedikt Kämpgen, Karlsruhe Institute of Technology (KIT)

[Work-package leader: Name, company] Carlos Ruiz Moreno, iSOCO - Intelligent Software Components, S.A.

[Estimation of PM spent on the Deliverable]

Copyright notice

Executive summary

In this document, we describe and evaluate our work for Task 3.1 *Knowledge Filters* during the third year of the ACTIVE project. It is centred around the concept of knowledge filters, tools that provide users with familiar means to retrieve, refine, and inspect knowledge. As results, we present two software components extending Semantic MediaWiki (SMW).

The first one offers knowledge filtering in the form of keyword search familiar to most employees. Ask The Wiki has been introduced before, here, its three step search process is evaluated in regard to effectiveness, efficiency and other aspects. For the effectiveness we analyze log data for the ratio of tasks that were successfully completed by trialists. For simple tasks the success rate is 100%, the more complex tasks result in lower success rates of 71% on average. For the efficiency we measure the number of keyword queries that the user had to issue in order to complete the task, also using log data. The average is surprisingly low, considering the structure and complexity of queries and results. Using a questionnaire, we further analyse specific aspects and user satisfaction of the search process. Although some users had difficulties answering the questions, the analysis of the questionnaire reveals that the users found the representation of interpretations easily comprehensible, and it was easy for them to choose the right interpretation. However, faceted search showed useful only if the users knew how to perform refinements. The overall evaluation results were encouraging, the concept of combining keyword search and semantic queries showed promising for our case studies. However, when we deployed the system in their wikis, we discovered limitations regarding configuration overhead, usability and performance. We present a light-weight version of Ask The Wiki, AskQ that tries to minimise these issues. AskQ is used in the case studies, due to time constraints it is not evaluated.

The second software component extends our Process Visualization which we first presented in an earlier deliverable. Users at our case study partners that are not skilled with modelling processes are not able to use it. Therefore, we present an extended version of Process Visualization, called Process Editor that is inspired by graphical tools such as Microsoft Visio. It not only allows to share, and comment on processes through SMW, but also allows for easy-to-use visual editing. We combine SMW with a graphical process editor, namely Oryx, yielding the following advantages: both novice and experts can capture informal processes; standard wiki features can be used for process modelling, e.g., versioning, watch lists, and both text and media storage; SMW acts as a process repository where processes and their process semantics are stored; process knowledge can be linked and browsed, queried and displayed on any wiki page; having processes described with structured metadata makes it possible to search for similar processes; and more experienced users can introduce their own properties and categories and extend the underlying schema of the metadata. Our evaluation results are limited as they rely on a small set of trialists. Nevertheless, the results indicate that our approach is suitable for collaborative modelling of informal and formal processes.

Both software components strongly build on previous work. Concepts that have been worked out during year one and two of the ACTIVE project have dealt with real business needs that are still of high importance. However, when finally deployed we came upon issues not anticipated before resulting in limitations of our prototypes. For instance, Ask The Wiki showed too heavy-weight, and Process Visualization was not usable by certain users. Only when we resolved these issues our prototypes evolved into well-suited components for concrete business scenarios and further research.

List of authors

Company	Author
Karlsruhe Institute of Technology (KIT)	Benedikt Kämpgen
Karlsruhe Institute of Technology (KIT)	Frank Dengler
Karlsruhe Institute of Technology (KIT)	Daniel Herzig

Contents

Executive summary 3

List of authors 4

Abbreviations 8

Definitions 9

1 Introduction 10

2 Ask The Wiki - evaluation and extension 11
- Three steps search process . . . 11
- Implementation . . . 11
- Evaluation of Ask The Wiki . . . 12
  - Data set and evaluation setting . . . 13
  - Task sets . . . 13
  - Evaluation results . . . 13
  - Overall effectiveness and efficiency . . . 13
  - Articulation of the information needs . . . 15
  - Query interpretation by keyword translation . . . 15
  - Result presentation and query refinement . . . 16
  - Response times . . . 17
- Light-weight version AskQ . . . 17

3 Process Editor extension and evaluation 18
- Previous version Process Visualization . . . 18
- Extended version Process Editor . . . 19

4 Conclusion 19

A Appendix 22
- A.1 Questionnaire . . . 22
- A.2 Research Proposal . . . 26
- A.3 Collaborative Process Development Using a Semantic Wiki-based Process Editor . . . 32

List of Figures

1 Result interpretation 12
2 Architecture of Ask The Wiki 12
3 Experience of the users 15
4 Efficiency and effectiveness of the search per task 15
5 Efficiency and effectiveness of the search per query type 16
6 Number of completed tasks 16
7 User feedback. Participants characterize the system. Multiple answers were possible 17
8 Results of the usability study 18

List of Tables

1 Task set “1” for semanticweb.org 14
2 Task set “2” for semanticweb.org 14

Abbreviations

Abbreviation	Description
SMW	Semantic MediaWiki
BT	British Telecommunications plc.
KEA	Kea-pro
Cadence	Cadence Design Systems
OWL	Web Ontology Language
RDF	Resource Description Framework
SPARQL	Simple Protocol and RDF Query Language
AKWS	ACTIVE Knowledge Work Space

Definitions

Term	Definition
Knowledge filter	By knowledge filters we refer to software components that provide the user with intuitive means for retrieving, refining and inspecting snippets of knowledge maintained within formal knowledge bases.
Enterprise knowledge structures	Enterprise knowledge structures we define as the collection of various information sources in an enterprise that employees use to manage the knowledge for their daily work. Examples include customer descriptions, product specifications, and directories of employees.
Knowledge base	A collection of statements or axioms in a given knowledge representation language.
Inline Queries	Wiki syntax for querying semantic data from Semantic MediaWiki.

1 Introduction

In this document we describe and evaluate two software components that we developed for Task 3.1 *Knowledge Filters* during the third year of the ACTIVE project. Those prototypes improve users' daily knowledge work by applying knowledge filters. With that we mean tools that suggest to the knowledge worker which information is best to use for a specific task, e.g., based on community experience and filtering mechanisms. In general, they provide the user with intuitive means for retrieving, refining and inspecting snippets of knowledge maintained within formal knowledge bases.

We specifically investigate methods that allow for context-aware filtering of relevant knowledge. The core concept behind context-awareness is reusing interaction metaphors that users know from other contexts, for instance from Google web search, GoPubMed web search[1], and Microsoft Office.

In year one of the ACTIVE project we have developed a knowledge filter based on the file system paradigm. *Semantic File System* (SemFS) mapps RDF data to virtual files and directories, and vice versa. On the one hand, content of knowledge bases can be seamlessly integrated with desktop systems, and on the other hand, file and directory structures can be seen as structured data enabling more powerful organization schemes. Also in Deliverable 3.1.1 [Bloehdorn et al., 2009], an early prototype of Ask The Wiki was presented – work we have built on during year three. Another example for user-centric knowledge filtering is the *Office Smarttag Wiki Plugin* which has been introduced in Deliverable 3.1.2 [Bloehdorn et al., 2010]. It allows Microsoft Office to transfer structured information to and from Semantic MediaWiki. Here, information icons – so-called Smart Tags – in the office software are used, which are familiar to most Microsoft Office users. When users type a term that is represented in the wiki, they can query for filtered information about this term directly from the office software. WikiTags has since then been further developed by Vulcan, e.g., stemming and stop-word recognition have been added. In year two of the ACTIVE project we have also pursuit a more formal approach to context-awareness. We enabled semantic search within Semantic MediaWiki that is aware of the actual working context of the user, given by the current context of the ACTIVE Knowledge Work Space (AKWS) [Dolinsek et al., 2010] and formalized through a micro-ontology. *ContextAsk*, the corresponding "contagging and contextualized search" extension has been introduced in Deliverable 3.1.2 [Bloehdorn et al., 2010]. Also in this deliverable we have presented a knowledge filter framework, *OntoNavigator*, for visualising ontologies and helping users to comprehend them.

In the following, as results of research and development for context-aware knowledge filters, we describe two software components and their evaluation. Both are extensions to Semantic MediaWiki (SMW), which in Deliverable 1.3.1 [Krötzsch et al., 2009] has been selected as a promising approach to managing *enterprise knowledge structures*. Those structures comprise various information sources in an enterprise that employees use to manage the knowledge for their daily work. Examples include process descriptions, product specifications, and directories of employees, as well as possible external information sources such as blog entries, news items, and twitter feeds, all of which are not to be replaced but enhanced by the wiki. In Deliverable 1.3.3 [Kämpgen and Ell, 2011] we are looking at how we have prepared SMW to the enterprise context in more general terms; whereas, in this deliverable we will go into details of two specific solutions based on knowledge filtering.

The first software component deals with keyword search and result presentation. When using wiki text, so-called *Inline Queries*, for querying structured data, the users in SMW need to know exactly what they are looking for. They need to know the underlying schema and specify the categories and properties. This is not realistic in an enterprise environment with users having no technical background. *Ask The Wiki* empowers users to use keyword search, e.g., known from Google, to let the system decide what information they are looking for and then to let it present the information accordingly. The concept of semantic-enhanced keyword search, therefore, is a well-suited example of knowledge filtering. It has already been described in Deliverable 3.1.1 [Bloehdorn et al., 2009]. Here, it will be evaluated. However, although evaluation results were encouraging, feedback from potential users within our case studies revealed deployment and usability issues. Therefore, we have developed a new light-weight version of Ask The Wiki, *AskQ*, which we will also cover.

Our second software component is related to processes. Processes are essential in daily tasks of knowledge workers within enterprises. More efficient ways to collaboratively capture, refine, and exploit processes therefore are of high importance. However, depending on their grade of formalization, processes are difficult to

[1] `http://gopubmed.org/web/goweb/`

grasp and knowledge about them difficult to manage. Process handling for the case studies has mainly be done with our ACTIVE *Process Visualization* described in Deliverable 3.2.2 [Tilly et al., 2010]. It allows users to share and improve on processes using Semantic MediaWiki. Although in use in BT and Cadence case studies, we are aware of usability issues, that on the long run might undermine the benefits of this approach. Therefore, we have developed an extended version of Process Visualization, called *Process Editor*, that is described and evaluated here. It applies knowledge filtering to both retrieving and modifying processes: Besides other advantages, it particularly makes processes available for visual editing by both expert and novice users. Also, it allows users to query for existing processes and then displays them in a comprehensible format.

This deliverable is structured as follows: In Section 2 we discuss our first software component; we evaluate Ask The Wiki, and then, describe our new version AskQ. In Section 3 we present and evaluate our extension to Process Visualization – Process Editor, after which, we conclude the outcome of Task 3.1 in year three of the ACTIVE project in Section 4.

2 Ask The Wiki - evaluation and extension

In this section we shortly summarize the user interface and the implementation of Ask The Wiki. This provides the context for the evaluation, thereafter.

Three steps search process

Ask The Wiki is integrated into Semantic MediaWiki as a "Special page extension". This extension provides the user interface and the faceted search methods via SPARQL modification. Here, we shortly summarize the search process. More detailed information we have included in Deliverable 3.1.1 [Bloehdorn et al., 2009]. The search process is divided into three steps described in the following:

Articulation of the information need Users articulate their information needs using simple keyword queries, e.g., "conference country abstract deadline". For searching, the users do not need to know about the query syntax, the schema and even the labels of the data elements. Ideally, they can use their own words to express their information needs.

Query interpretation using keyword translation This step is concerned with the translation of the user queries into system queries, i.e. structured conjunctive queries. The submitted keywords are hereby interpreted as elements of structured queries. The keyword search process contains an additional step, namely the presentation of plausible structured queries given the submitted keywords. According to a query ranking scheme, the computed query graphs are sorted and presented to the user for selection. Here, the queries are not presented using a formal syntax, but using an intuitive, graph-based representation.

Result presentation and refinement After the user has chosen the correct query, the query results, which in the general case are sets of tuples satisfying the conjunctive query, are presented to the user in a structured, tabular form. Further, the user can refine the query following the paradigm of faceted search.

See Figure 1 for an example of the interpretation step with Ask The Wiki.

Implementation

Ask The Wiki consists of two major parts, the Semantic MediaWiki extension and a back-end, which we shortly give a summary of in the following.

Figure 2 shows the architecture of Ask The Wiki. It is implemented in PHP and uses AJAX for the user interaction. Although, we pre-compute the schema, because it is often not fully specified in the wiki, the facets are retrieved on the fly for each query individually via AJAX from the back-end. All user selections, query modifications, and process states are kept on the client side.

The back-end is realized as a Java Servlet running on a Tomcat server. The servlet provides the keyword translation and top-k query construction, as well as the query evaluation. Before the keyword translation and query construction are available online, an offline pre-processing step is required. This step comprises computing the schema graph and indexing the OWL data export of SMW using special schema and keyword

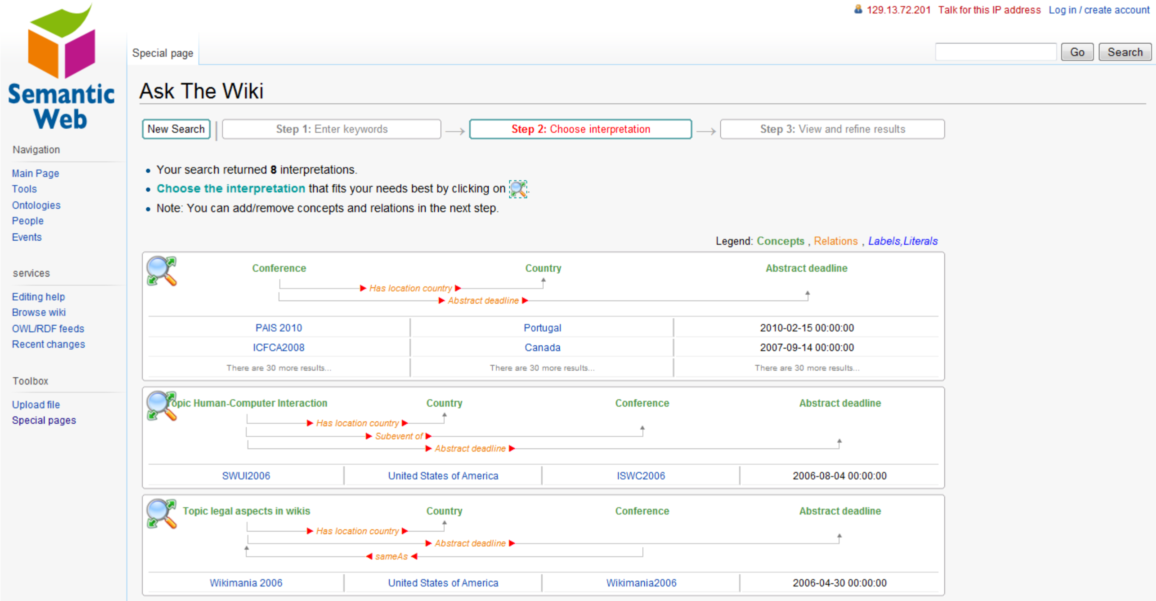


Figure 1: Result interpretation

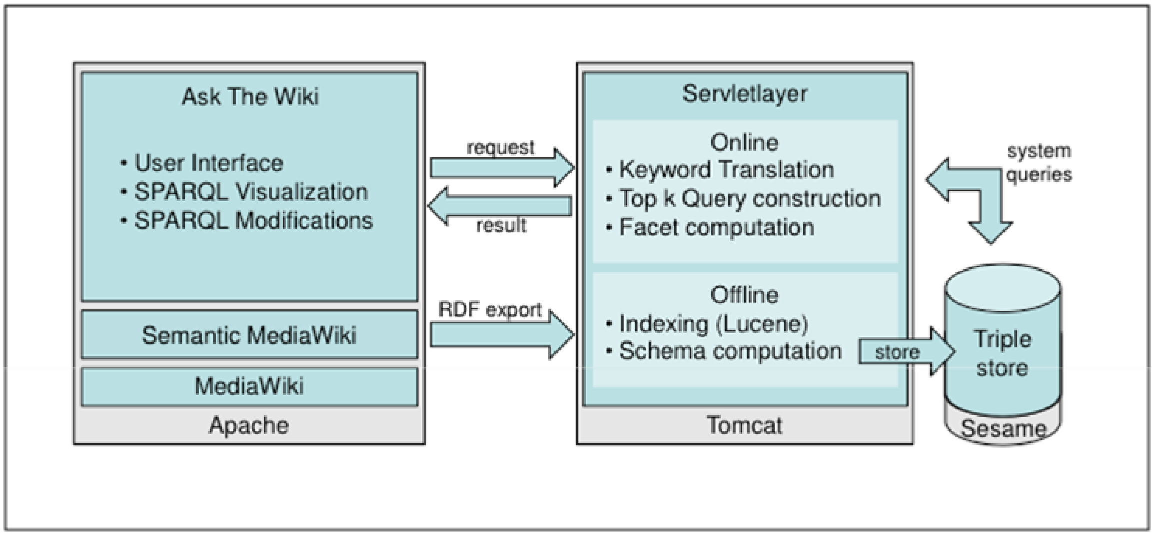


Figure 2: Architecture of Ask The Wiki

indexes, which are realized using Lucene. For storing the data and processing the SPARQL queries, in principle, any triple store exposing a SPARQL endpoint can be used. We use Sesame with its native store and enabled inferencing.

The implementation is compatible with SMW 1.2 (and above) and Tomcat 5.5 (and above) running on Java 6. We have already deployed the system on several instances running Semantic MediaWiki. In the following section, we report on evaluation experiments in one of the installations of Ask The Wiki.

Evaluation of Ask The Wiki

The goal of the evaluation was to assess the potential and ability of our approach to semantic search in a real-life application. Since the search should make the potential of the underlying semantic technologies available to end users, we performed a user study to evaluate the system in terms of effectiveness and efficiency, as well as user satisfaction and usability.

Data set and evaluation setting

We performed the evaluation within the community portal *semanticweb.org*, a wiki-based platform serving the Semantic Web community. The wiki contains information about the Semantic Web, in particular (but not limited to) events, publications, tools, and people. Its OWL data is available in the RDF/XML format and comprises a total of 55,365 triples, with 657 classes, 948 properties, 27,778 property instances, and 11,275 individuals. The data has been created by the users of the wiki over the last three years. Since the nature of a wiki is to provide unconstrained user editing, the data does not follow a predefined vocabulary or strict schema. The participants of the user study were 14 volunteers from different organizations active in the Semantic Web community.

We chose a task based user evaluation with each task representing an information need that could typically occur when using the portal. Afterwards we asked the participants to answer a multiple choice questionnaire about their experience. The questions concerned their technical background (experience with other search engines, familiarity with certain technologies, etc.) and the experience and satisfaction with certain aspects of our search process. Additionally, the participants could give free text comments. The whole questionnaire is in the appendix in Section A.1.

Each participant got five tasks and had up to three minutes to solve each task. The participants could give up before, if they felt that they could not solve the task. The participant received very limited information about the search interface upfront, namely that the search consists of three steps and that the search will not return a list of links like the common web search engines, but an interpretation of their keywords, also that they have to choose an interpretation, if necessary and that they could modify the interpretation in the third step. However, there was no walk-through introduction or information how to perform these steps.

The tasks were constructed so that as many aspects of the systems functionality were covered, with different levels of difficulty. In particular, we created tasks that required queries that fall into the categories introduced before (i.e. entity queries such as *Find the page describing the AIFB institute*, fact queries such as *When is the paper deadline for the ASWC2008*, general conjunctive queries such as *Find the capitals of countries in Europe and the population of these cities*. The tasks are given and discussed in the next section Section 2. All actions taken by the participants and system responses were logged. In particular, we logged the users' steps and keyword inputs and the system responses and measured how often users could solve the task, how much time it took them and how well the system performed in terms of keyword translation, query construction, and query evaluation.

The evaluation was performed based on both the analysis of the log files as well as the questionnaire.

Task sets

We created two task sets - 1e to 5e and 1d to 5d - in order to minimize the risk of unforeseen problems with particular terms. Both sets have the same structure and the same query types. Each set was given to seven participants. See Table 1 and Table 2 for the two task sets. The tasks are crucial for the success of the evaluation. If the participants misunderstand the tasks, the results of the evaluation are misleading or unclear. Fortunately, we did not have any major problems. However, some participants, who had no or little technical background (see Figure 3 for overview of experiences), said that they mistook "wiki page" and "homepage" at first. Thus, we recommend avoiding those terms in the future. Our trialists are mainly of technical background, especially they are familiar with SQL, so that our evaluation might not fully describe the impact our solution would have for non-technical users. Yet, with more than half of the participants having no practical experiences with SPARQL, the underlying concept of our search process, we think that our evaluation can give insights of usage also by more average users.

Evaluation results

We now report on the evaluation results, discussing first the overall effectiveness and efficiency, and then in more detail specific aspects of the individual steps of our search process.

Overall effectiveness and efficiency

To measure the overall effectiveness, we analyze the ratio of tasks that have been successfully completed.

Task No	Task Description	Type
1d	Find the wiki page of AIFB.	Entity query
2d	When is the paper deadline for the ASWC2008?	Fact query
3d	What is the email of Holger Lewen?	Fact query
4d	Find exporter with GPL license and their homepage.	General conjunctive query
5d	Find the capitals of countries in Europe and the population of these cities.	General conjunctive query

Table 1: Task set “1” for semanticweb.org

Task No	Task Description	Type
1e	Find the wiki page of Stanford University.	Entity query
2e	What is the homepage of the ISWC2008 conference?	Fact query
3e	What is the email of Thanh Tran?	Fact query
4e	Find reasoner with GPL license and their homepages.	General conjunctive query
5e	Who was the local chair of the conferences located in Karlsruhe in 2008?	General conjunctive query

Table 2: Task set “2” for semanticweb.org

To assess the efficiency, we measure the number of keyword queries that the user had to issue in order to complete the task. The results for these measures are shown in Figure 4. The results are aggregated over all queries, grouped by the type of query that was needed to obtain the result.

For the simple tasks (entity queries), the success rate was 100%, the more complex tasks result in lower success rates: 79% for the fact queries and 64% for the general conjunctive queries (see Figure 5). Figure 4 shows the success rate for each task individually.

A typical reason why particular tasks were not completed was that the matching of the keywords against the available data was unsuccessful and no interpretations could be generated: The gap between chosen keywords and the underlying data was too large in certain cases. The more complex the queries (both in terms of structure and number of keywords), the larger was the effect (see Section 2 for a discussion of keyword translation).

There is a notable difference between the success rates of task 4e and 4d, as well as between 5e and 5d. For tasks 4e the success rate is comparatively low, because four of the seven participants stopped short to add the property “homepage” as it was asked by the task, although they all had the correct interpretations up to this point. Why the participants stopped here, is not clear. Probably, they forgot to add it or they misunderstood the term “homepage” and thought that they found it already. The fact that the participants did not issue more queries and tried solve the task again, as they did for task 5e (see discussion below) but proceeded to the next task, is backing this assumption. One participant did not add the property “homepage” for task 4d. Task 5e has a lower success rate, because the number “2008” in the task description caused some problems and many participants included it in every keyword query; the number “2008” matched many elements whose constructed graphs resulted in empty sets. Therefore, this task yielded a lower success rate.

For the efficiency, we see that on average the users needed to issue between 1.6 and 2 keyword queries to fulfil a task, depending on the query type. This number is surprisingly low, considering the structure and complexity of the generated queries and results. Expectedly, the value is larger for the more complex, general conjunctive queries (2 keyword queries per task) than for the simple types of queries. The quite large difference between the number of queries issued for task 5d and 5e is directly correlated to the low success rate of task 5e. The participants tried harder to solve the task 5e and thus issued more queries, whereas task 5d was apparently easier to solve. Overall, 6 out of 14 participants were able to fulfil all five tasks, 12 of the 14 were able to fulfil 60% or more (see Figure 6). The other two users quickly gave up after the first or second query stating that they found the system too complicated (see Figure 7).

We find the overall success rate encouraging, considering that the participants used the system without

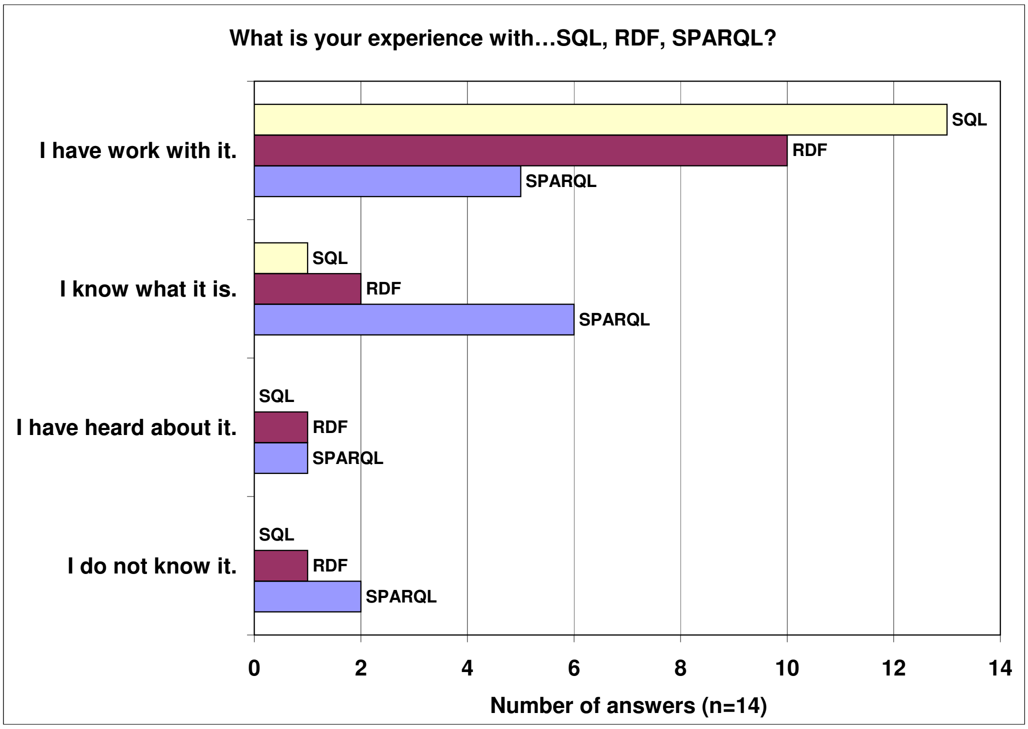


Figure 3: Experience of the users

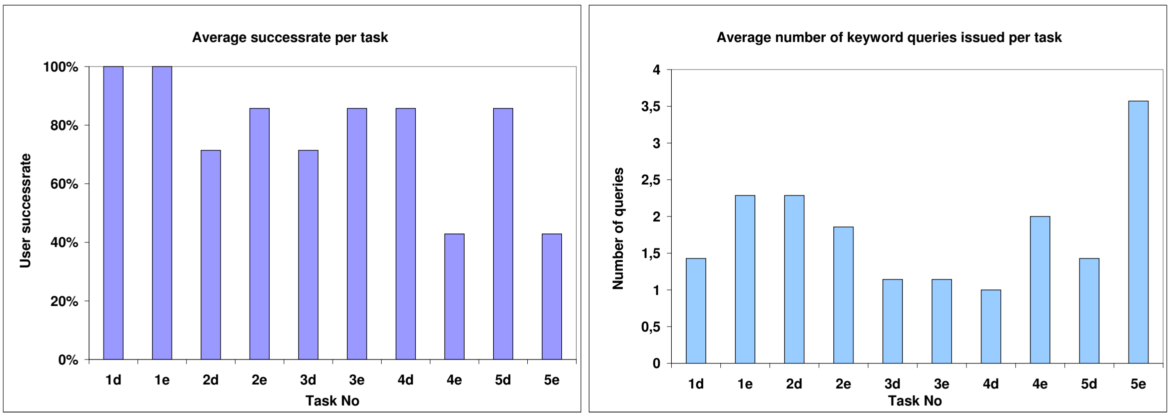


Figure 4: Efficiency and effectiveness of the search per task

detailed usage instructions and without knowing the schema of the underlying data. We now discuss specific aspects of the three steps involved in our search process. In the questionnaire, we ask various questions related to the individual steps. The responses to these questions are shown in Figure 8.

Articulation of the information needs

The first question asks how difficult the users found it to express the information need in keywords. As expected, the users found it rather easy to do so, as all of them were familiar with keyword-based search interfaces.

Query interpretation by keyword translation

The next aspect we analyze is the quality of the translation of the keyword queries to structured queries. First, we analyze the robustness of the keyword matching: 88% of the keyword queries could be translated into structured queries. As mentioned above, the main reason why certain keyword matches failed is the gap between the keywords and the underlying data. As the users were not aware of the underlying schema at all, in some case no matching to the underlying data could be reached. However, after a more detailed analysis of the failures, we were able to improve the keyword matching algorithm after the evaluation.

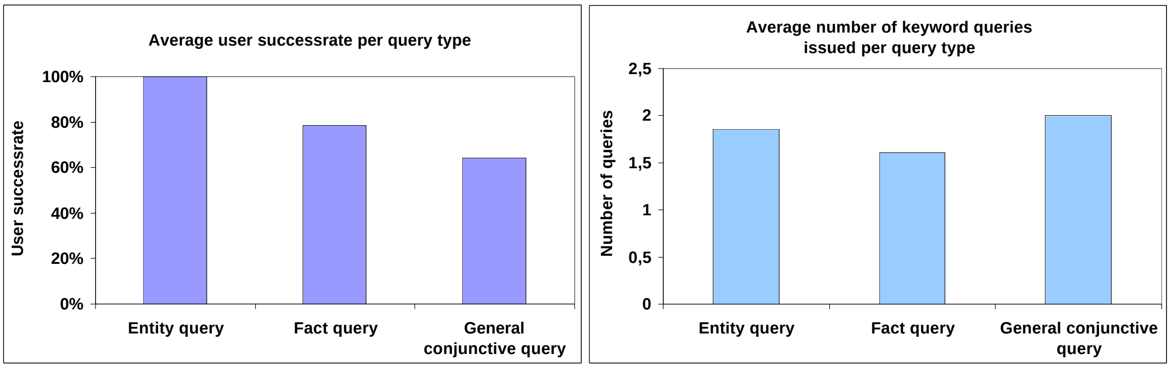


Figure 5: Efficiency and effectiveness of the search per query type

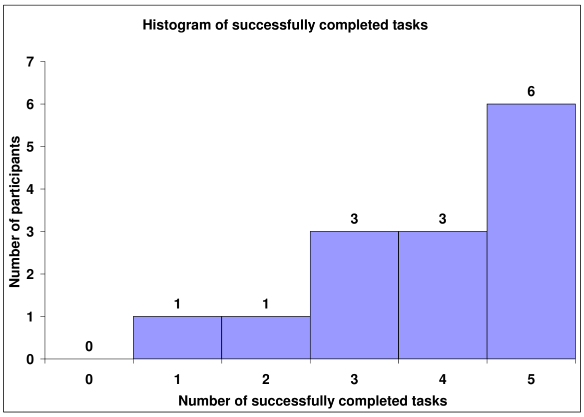


Figure 6: Number of completed tasks

Second, we measure the quality of the rankings of the possible interpretations. For this, we analyze, which interpretation was selected by the user as the correct interpretation. We adopt a standard information retrieval metric called Reciprocal Rank (RR) defined as $RR = \frac{1}{r}$, where r is the rank of the correct query. The mean RR for all queries is 0.84. For successfully completed tasks, the users selected the top-ranked interpretation in 76% of the queries. The results indicate that the intended interpretation in most cases was ranked correctly, i.e. at first position.

Overall, the users found the representation of interpretations easily comprehensible, and it was easy for them to choose the right interpretation. Yet a few users had difficulties (see questions regarding step 2 in Figure 8), one reason being that in some cases the interpretations were so similar that the users could not easily tell the difference.

Result presentation and query refinement

Finally, we analyze the user satisfaction with the result presentation and query refinement. The majority of the users found the presentation of the results understandable. However, only seven users made use of the faceted search to refine a query. This corresponds to the question about how useful the modification was to the users: Seven participants found it very useful or useful to modify the interpretations, whereas three participants stated that they *did not know how to do it* (see Figure 7). Apparently, some of the participants were not familiar to faceted search and query refinement, and had not recognized the faceted search box on the right side as belonging to the search functionality. This suggests that our result presentation is useful only if the users know how to perform refinements. As a consequence, effective use by inexperienced users requires more detailed instructions, which were deliberately not given in our setting.

Interestingly, the use of the faceted search was particularly effective for the more complex tasks. On average, 29.6% of the successfully completed tasks involved a refinement. For the most complex tasks involving general conjunctive queries, 38.9% of the successfully completed tasks involved refinements. We thus have reasons to believe that the overall success rate would have been considerably higher, if all users had known how

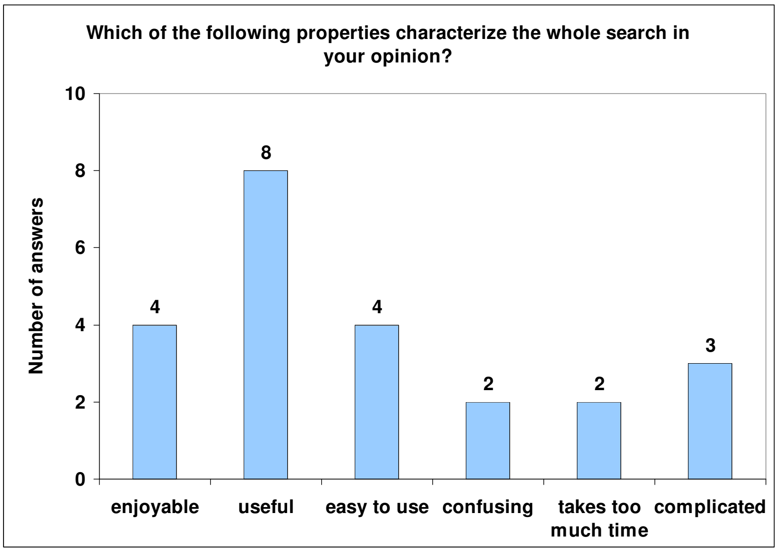


Figure 7: User feedback. Participants characterize the system. Multiple answers were possible

to effectively utilize faceted search.

Response times

The evaluation of the system performance in terms of response times was not a primary goal of our study, as this aspect already has been extensively analyzed in our prior work [Tran et al., 2009]. There, we show that the query translation and query processing can be handled in near real-time even with considerably larger data sets. Still, for completeness reasons we measured the time to translate keyword queries as well as the time for query answering. The keyword translation was performed on average in 132 ms, while the query answering (evaluation of the conjunctive query) on average took 31 ms. As we will discuss in the next section, performance issues rather appeared when it was required to continuously update the underlying triple store.

Our evaluation is limited with respect to comparison to related approaches. As gold standard one can see Google search; a comparison would be possible regarding precision and recall, which, however, was not the the focus of our work. Instead, it was our intention to analyse users behaviour when dealing with our search process to articulate complex queries referring to structured information of the underlying data, which per se is not possible with Google search. For the same reason our evaluation has not required help by our evaluation partner KEA. Despite this limitations, our evaluation provided useful insights into our search process, and revealed actionable results, as described in the next section.

Light-weight version AskQ

Although the evaluation results revealed that our search process is accepted by users, when we tried to introduce Ask The Wiki to the wikis of our case study partners, we have found limitations to our approach:

- Although the interpretation step of the search procedure was comprehensible users did not appreciate the full range of graph queries, but with very few exceptions chose tree-shaped search results.
- For up-to-date information, Ask The Wiki requires to materialise all structured wiki content into RDF and storing it in a triple store. This is slowing down the performance and not feasible in wikis with a high modification rate.
- Ask The Wiki needs a triple store on a Tomcat server. The resulting configuration overhead was a burden for our case study partners.
- Results are only displayed as a table. Semantic MediaWiki and extensions support many more visualization formats such as timelines, graphs, and calendars, as well as export formats such as BibTex, CSV, and MS Project.

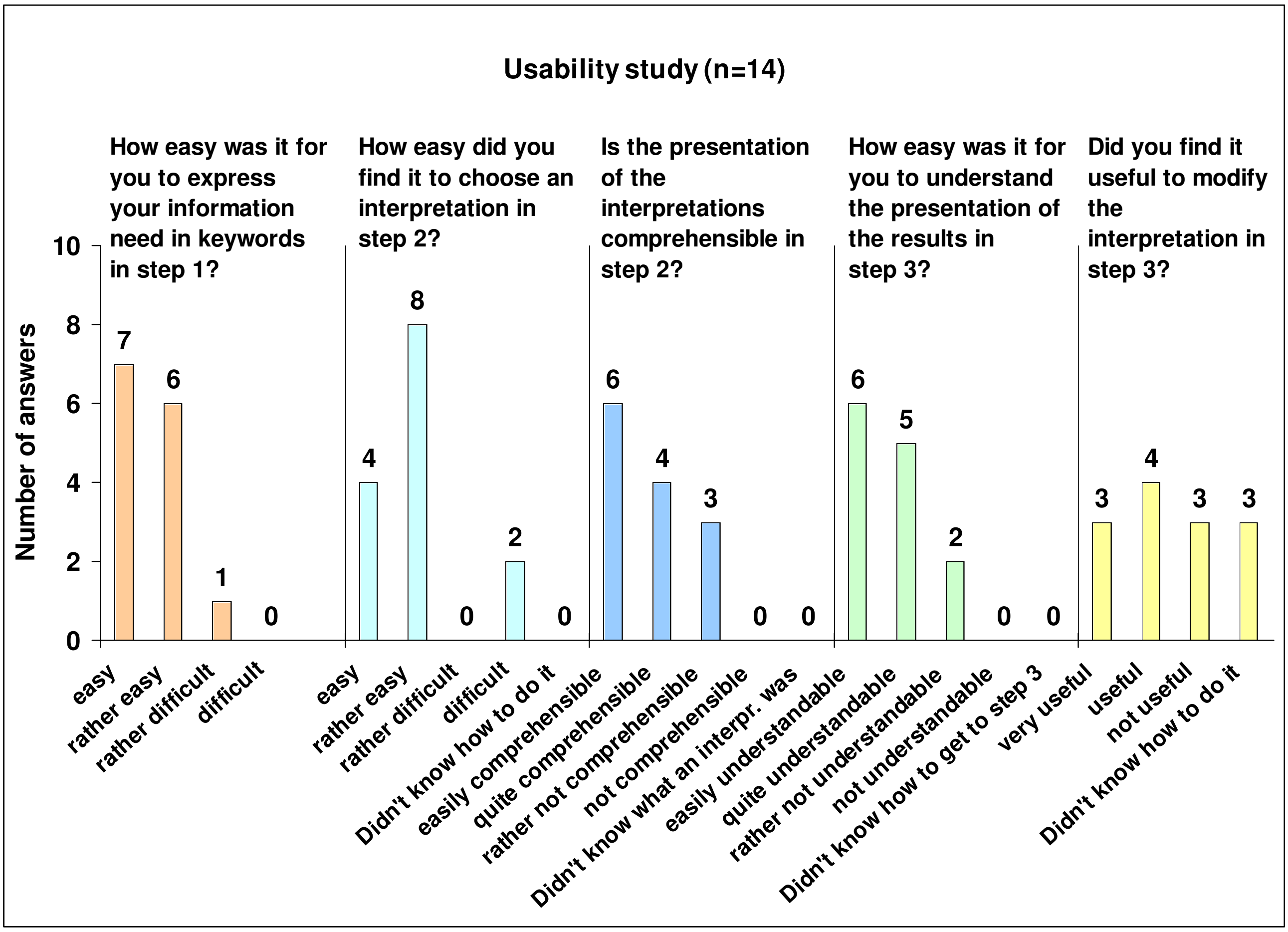


Figure 8: Results of the usability study

We have reacted to these issues and developed a new light-weight version of Ask The Wiki, *AskQ*. There, expressivity of the query processing is reduced to tree-shaped results. Consequently, search can be performed directly in SMW: data does not need to be materialised into a triple store but instead is translated and indexed on-the-fly; also, dependencies to a triple store and Tomcat webserver are removed. In addition, since evaluation revealed that most of the time the system would rank the chosen interpretation first, the interpretation step is now done automatically, less likely interpretations are only shown if explicitly requested. Also, after the first results have been presented, visualization or export formats can be selected.

AskQ has been published[2] to the community, a demo[3] is publicly available, and we intend to further extend it after the ACTIVE project: At the moment, only the interpretation step is done automatically, the results are by default displayed as a table. We plan to automate the visualization selection based on the resulting data, possibly learned with machine learning algorithms and trained on previous user behaviour. AskQ was tested in the case studies, as described in Deliverable 1.3.3 [Kämpgen and Ell, 2011], but due to time constraints, could not be evaluated.

3 Process Editor extension and evaluation

Previous version Process Visualization

In Deliverable 3.2.2 [Tilly et al., 2010] we have presented Process Visualization as a means for sharing and discussing processes in Semantic MediaWiki. This extension is in use at our case study partners BT and Cadence, as described in Deliverable 1.3.3 [Kämpgen and Ell, 2011]. However, Process Visualization revealed limitations for Accenture where querying and editing of processes is done both by experts and novices. In

[2] http://www.mediawiki.org/wiki/Extension:AskQ
[3] http://www.aifb.kit.edu/web/Spezial:KeywordSearch

particular for novices, Process Visualization is cumbersome to use since processes can only be created and modified by creating wiki pages and adding annotations. Instead, a more Microsoft Visio-like interface is needed.

Extended version Process Editor

Therefore, we have developed Process Editor[4], an extension to SMW that provides this functionality of creating and discussing processes by an intuitive visual process editing interface as well as functionality of annotation-based search over existing processes. In order to combine formal semantics, natural language and visual editing, we integrate SMW and Oryx Process Editor [Decker et al., 2008]. For additional information, we have attached two papers: First, a research proposal that has framed our work for Task 3.1 in year three of the ACTIVE project (Appendix A.2). Also, we have attached a paper that has been submitted to K-CAP 2011 (Appendix A.3). It contains a detailed description of our prototype and a usability evaluation. Our evaluation results are limited as they rely on eight participants, only. Nevertheless, the results indicate that our approach is suitable for collaborative modelling of informal and formal processes.

4 Conclusion

In this section, we conclude the outcome of ACTIVE Task 3.1 *Knowledge filters* in year three of the ACTIVE project. We have presented and evaluated two knowledge filters that approach important business needs. The first is a software component for keyword search of structured data, which is familiar to most employees. The three steps search procedure in Ask The Wiki was shortly summarized, then we analyzed its utility regarding effectiveness, efficiency and other aspects. For the effectiveness we analyzed log data for the ratio of tasks that have been successfully completed by trialists. For simple tasks the success rate was 100%, the more complex tasks resulted in lower success rates of 71% on average. For the efficiency we measured the number of keyword queries that users had to issue in order to complete the task, again using log data. The average was surprisingly low, considering the structure and complexity of queries and results. Using a questionnaire, we further analysed specific aspects and the user satisfaction of the search process. Although some users had difficulties answering the questions, the analysis of the questionnaire revealed that the users found the representation of interpretations easily comprehensible, and it was easy for them to choose the right interpretation. However, faceted search showed useful only if the users know how to perform refinements. The overall evaluation results were encouraging, the concept of combining keyword search and semantic queries showed promising for our case studies. However, when we deployed the system in their wikis, we discovered limitations regarding configuration overhead, usability and performance. We presented a light-weight version AskQ that tries to minimise these issues. AskQ is used in the case studies, due to time constraints it could not be evaluated.

Second, we have presented a software component that extends our Process Visualization extension for managing processes with Semantic MediaWiki. It is inspired by graphical tools that are familiar to most employees, e.g., Microsoft Visio. Process Editor not only allows to share, and comment on processes through SMW, but also allows for easy-to-use visual editing. We combined a graphical process editor, namely Oryx, with SMW which yields the following advantages: Both novice and experts can capture informal processes; standard wiki features can be used for process modelling, e.g., versioning, watch lists, and both text and media storage; SMW acts as a process repository where processes and their process semantics are stored; process knowledge can be linked and browsed, queried and displayed on any wiki page; having processes described with structured metadata makes it possible to search for existing processes; users can introduce their own properties and categories and extend the underlying schema of the metadata. Our evaluation results are limited as they rely on a small set of trialists. Nevertheless, the results indicate that informal and formal processes can be collaboratively modelled by using our approach.

Both software components strongly built on previous work. Concepts that have been worked out during year one and two of the ACTIVE project have dealt with real business needs that are still of high importance. However, when finally brought to deployment we discovered limitations of our prototypes not anticipated before. For instance, Ask The Wiki showed too heavy-weight, and Process Visualization was not usable by

[4] Not yet released to the public.

certain users. Only when we resolved these issues our prototypes evolved into well-suited components for concrete business scenarios and further research.

References

[Bloehdorn et al., 2010] Bloehdorn, S., Dengler, F., Vrandečić, D., and Tilly, M. (2010). Knowledge filters - demonstrator. Deliverable, ACTIVE.

[Bloehdorn et al., 2009] Bloehdorn, S., Haase, P., Herzig, D., Kreidler, A., and Vrandečić, D. (2009). Knowledge filters and metaphors early prototype. Deliverable, ACTIVE.

[Decker et al., 2008] Decker, G., Overdick, H., and Weske, M. (2008). Oryx An Open Modeling Platform for the BPM Community. In Dumas, M., Reichert, M., and Shan, M.-C., editors, *Business Process Management*, volume 5240 of *Lecture Notes in Computer Science*, pages 382–385. Springer Berlin / Heidelberg.

[Dolinsek et al., 2010] Dolinsek, I., Stajner, T., Moreno, C. R., and Tilly, M. (2010). Fully functional prototype of active knowledge workspace. Deliverable, ACTIVE.

[Kämpgen and Ell, 2011] Kämpgen, B. and Ell, B. (2011). Collaborative articulation of enterprise knowledge - fully functional prototype and evaluation. Deliverable, ACTIVE.

[Krötzsch et al., 2009] Krötzsch, M., Dengler, F., Siorpaes, K., and Vrandečić, D. (2009). Collaborative articulation of enterprise knowledge (early prototypes). Deliverable, ACTIVE. Available at http://www.active-project.eu/fileadmin/public_documents/d131-final.pdf.

[Tilly et al., 2010] Tilly, M., Ruiz, C., and Dengler, F. (2010). Context-aware knowledge processes fully functional prototype. Deliverable, ACTIVE.

[Tran et al., 2009] Tran, D. T., Wang, H., Rudolph, S., and Cimiano, P. (2009). Top-k exploration of query graph candidates for efficient keyword search on rdf. In *Proceedings of the 25th International Conference on Data Engineering (ICDE'09)*, Shanghai, China. http://people.aifb.kit.edu/dtr/papers/keywordTopk_tr.pdf.

A Appendix

A.1 Questionnaire

Questionary 1/3

Thank you for participating in this evaluation!

It will take you less than 20min to finish it.

Please return your answers by Friday December 5th to
Daniel Herzig <daniel.herzig@student.kit.edu>

What do I need to participate?

You need an internet connection and a Firefox browser.

What do I have to do?

There are a couple of questions and five tasks on the next pages.

You just have to mark your answers for the questions and try to solve the tasks.

What happens with my answers?

Your responses help us to evaluate the search. All answers and responses will be handled confidentially and anonymously at all times.

What is it about? How do I solve the tasks?

This search is different in two ways to the usual common search functions.

1) The result is not a list of links, but the actual data the user is looking for (which might be links, but also phone numbers, email address of certain people etc.)
2) The search process has **three steps** until you find a result.
   In first step you express your information need in keywords and enter those keywords in the search field. In the second step you choose an interpretation of your keywords. Choose the interpretation closest to your information need.
   In the third step, you can modify the interpretation, if necessary.

The scenario of this evaluation:

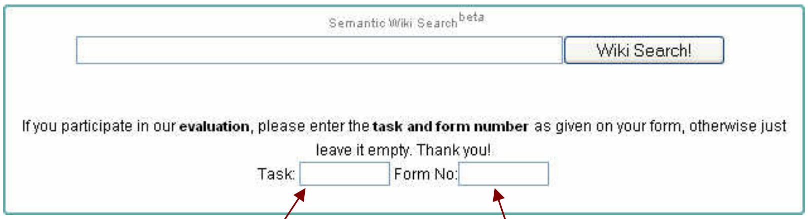
You are a user looking for certain facts and information. What you are looking for is given in the form of tasks. Each task represents an information need. The wiki system contains the information to satisfy the information need. You want to find the answers to the tasks only by using “Ask The Wiki”.

Questionary 2/3

➔ **Please open this website in your Firefox browser:**

http://semanticweb.org/wiki/Special:ATWSpecialSearch [1]

You should see the search field in the center of the page which looks like this:



➔ **Please enter the task number and the form number with each search in the fields below the search field.**

Your form number is: **xyz**

➔ **You can spend up to 3 min per task.**
But you can give up before, if you feel like it. Just proceed to the next task.
If you found an answer, proceed to the next task.

Task no:	Task description:
1e	Find the wiki page of Stanford University.
2e	What is the homepage of the ISWC2008 conference?
3e	What is the email of Thanh Tran?
4e	Find reasoner with GPL license and their homepage.
5e	Who was the local chair of the conferences located in Karlsruhe in 2008?

[1] You can also reach the website by opening www.semanticweb.org
Then click on "Special pages" on the bottom of the left navigation menu and then "Ask the Wiki" on the following overview page.

Questionary 3/3

Question1: How old are you?

☐ < 20 ☐ 20 - 24 ☐ 25 - 34 ☐ 34 - 44 ☐ 45 - 54 ☐ > 55

Question2: What is your experience with…?

SQL
- ☐ I do not know it.
- ☐ I have heard about it.
- ☐ I know what it is.
- ☐ I have worked with it.

RDF
- ☐ I do not know it.
- ☐ I have heard about it.
- ☐ I know what it is.
- ☐ I have worked with it.

SPARQL
- ☐ I do not know it.
- ☐ I have heard about it.
- ☐ I know what it is.
- ☐ I have worked with it.

Question3: Please mark the websites you used before to search information on the web:

☐ Altavista ☐ Ask.com ☐ Cuil ☐ Google ☐ Powerset ☐ Yahoo

Question4: How easy was it for you to express your information need in keywords in step1?

☐ easy ☐ rather easy ☐ rather difficult ☐ difficult

Question5: How easy did you find it to choose an interpretation in step 2?

☐ easy ☐ rather easy ☐ rather difficult ☐ difficult ☐ I did not know how to do it.

Question6: Is the presentation of the interpretations comprehendible in step 2?

☐ easily comprehendible ☐ quite comprehendible ☐ rather not comprehendible ☐ not comprehendible ☐ I did not know what an interpretation was.

Question7: How easy was it for you to understand the presentation of the results in step3?

☐ easily understandable ☐ quite understandable ☐ rather not understandable ☐ not understandable ☐ I did not know how to get to step3.

Question8: Did you find it useful to modify the interpretation in step 3?

☐ Very useful ☐ useful ☐ not useful ☐ I did not know how to do it.

Question9: Which of the following properties characterize the whole search in your opinion?

☐ enjoyable ☐ useful ☐ easy to use ☐ confusing ☐ takes too much time ☐ complicated

Please give some comments: Have you had problems? Would you use this tool later? General feedback and comments:

……….
……….
……….
……….

YOU ARE DONE!! THANK YOU !!!

A.2 Research Proposal

Collaborative Development of Informal Processes

Frank Dengler

Karlsruhe Institute of Technology (KIT)
Institute of Applied Informatics and Formal Description Methods (AIFB)
Karlsruhe, Germany.
frank.dengler@kit.edu

Abstract. Documenting and sharing informal processes in a machine-processable format can increase the efficiency of process management procedures and practices within enterprises, as it is expected that novices can be feasibly guided through these processes without requiring assistance from business process experts. In this context, state-of-the-art methods and tools for process modeling do not provide the most appropriate methods. We address this limitation by combining established process management techniques with semantic wiki technology, thus enabling the collaborative development of machine-understandable process descriptions via graphical and textual representations.

1 Introduction

During their daily task, knowledge workers deal with processes. These can be business processes, formally defined within an organization, but also informal processes, which tend to be scarcely documented, knowledge intensive and dependent on individual preferences, experiences and level of expertise [12].

Informal processes have to be made explicit, in order for them to be reused, shared and improved. This can be done by using either process mining or modeling tools. Via process mining [1] one can automatically identify and analyze activities performed by knowledge workers. Alternatively, information processes can be elicited via manual modeling using editors similar to those known from areas such as business process management. This is of particular importance, as not all activities within an informal process can be reliably detected via process mining techniques. This applies most notably for situations in which process-relevant knowledge is communicated and exchanged orally, during a phone call or during a discussion in the coffee corner.

Knowledge workers do not possess the same level of expertise in process modeling and using semantic annotations. Thus support for novice users is essential. Recker et al. [9] have investigated how novice model business processes in the absence of tool support. Their findings are that design representation forms chosen to conceptualize business processes range from predominantly textual, to hybrid, to predominantly graphical types. They have also discovered that the combined graphical and textual types achieve higher quality. As a consequence, collaboration support as well as manual process modeling and semantic annotation support for novice users are required. Driven by the main research question *how collaborative development of informal processes can be supported with a tool efficiently*, we elaborate the requirements for collaborative process development and provide novice users with means to model processes, which can be further refined by experts. Our approach integrates Semantic MediaWiki [8] with a graphical process editor and thus enables the collaborative creation of semantic process descriptions with a graphical representation, natural language and semantic annotations.

2 Proposed Approach and Methodology

To eliminate the deficiencies of state-of-the-art methods and tools for process modeling, we combine the collaborative creation of knowledge structures (SMW) with graphical process modeling functionality. SMW provides a special syntax to define class hierarchies (Categories and Sub-categories) and semantic properties (Properties and Sub-properties) related to wiki pages. For instance a category *Process* can be added to a wiki page by adding `[[Category:Process]]` on the wiki page. Hence, knowledge workers can develop process knowledge by using graphical representation, natural language and semantic annotations. Processes and their process semantics are stored in a lightweight process ontology within SMW. Therefore we support the *Basic Control-Flow Patterns* introduced in [11]. Every single process step (activity) is represented as a wiki page belonging to category Process Element and linked via the properties *has Type* to the corresponding type (Task) and *Belongs to Process* to the corresponding process summary page in SMW. An activity is the basic element of our process. Depending on the granularity level of the process this can vary from atomic activity, such as open a web page, to activities describing a whole subprocess. The control flow of the process is expressed by using edges in the diagram and the additional property *has Successor* on the corresponding wiki page in SMW. Special predefined process elements (gateways) are used for branching and synchronization.

Our approach has various advantages. The combination of natural language with formal semantics allows collaborative modeling for both novice and experts. Textual and graphical elements can be used interchangeably and complementarily. If the user does not know the graphical representation of a process element, natural language can be used to describe it. An extendible underlying schema is used. Users can introduce their own properties in the wiki by using the SMW property syntax on the process element wiki page. Thus, processes can be linked to existing knowledge (e.g. what input documents are used, can be made available and processable for computers). In addition, standard wiki features can be used for process modeling, like versioning, watch lists, reverting, etc. SMW acts as a process repository. Thus, process knowledge can be linked with semantic properties, queried and displayed on process pages and on other wiki pages. Structured data can improve the search for processes.

Our research will be carried out in several phases. In the first phase, we define informal processes, their occurrence in enterprises and their management. Therefore we describe the characteristics of informal processes, show various examples of informal processes across different companies and illustrate informal process management on the basis of a informal process lifecycle. Afterwards we show how this lifecycle integrates into BPM. The second phase consists of a requirement analysis for modeling informal processes. We study the existing literature as well as existing process descriptions in enterprises and derive requirements to support the modeling of informal processes. In the third phase, we develop a tool based on the previous collected requirements to support knowledge workers in developing informal processes collaboratively. As we support process modeling with graphical representation and natural language in combination with semantic annotations, inconsistencies can appear between graphical and textual descriptions, that we want to detect. In the forth phase the approach is evaluated in two different case studies. The functionality is validated by re-modeling existing scientific workflows from myExperiments [10]. The usability, especially the collaboration, is studied with knowledge workers in an enterprise.

3 Results

By reviewing the literature on process modeling and analyzing existing processes, we derived requirements for supporting collaborative process development. As many best-practice descriptions and how-tos are already stored in wikis, we extended our approach presented in [4] and integrated a graphical process editor with SMW to support collaborative process development. For our implementation we selected the Oryx Process Editor [2], an open source process editor, as the graphical process editor component and combined it with SMW.

For example team members can collaboratively model their processes (e.g. review process for documemts) and link them via semantic properties to existing knowledge stored in the wiki (e.g. knowledge of organizational structure or process-relevant information). Thus process-relevant documents can be filtered out for each activity by using the query language provided by SMW. In addition, process patterns like approval activities can be queried and checked (e.g. if there is always a manager involved).

We have done a pre-evaluation with three students, showing the functionality and intuitive usability of our tool by re-modeling scientific workflows. The results can be found in our demo wiki[1].

4 State of the Art

This work will be based on and contribute to the state of the art in four main areas, namely (1) social software for process management, (2) collaborative process modeling and (3) Semantic Business Process Management (SBPM).

[1] The demo wiki can be accessed via http://oryx.f-dengler.de (Username: ProcessTester – Password: active!)

There are many research activities to support Business Process Management (BPM) with social software [5]. To support workflows, Dello et al. [3] have extended the Makna Semantic Wiki by integrating the workflow engine jBPM to enable the coordination of interactions within a wiki system. However, this approach do not support process development for novice. The Moki enterprise modeling wiki [6], by contrast, allows collaborative development of processes, but the process editor is embedded in the process summary page and does not allow the user to read and edit the wiki pages of the process elements. In addition, process semantics, like successor relations, are not stored on the process element pages. Therefore, queries concerning process semantics, such as show me all process activities resulting in approval activities, are not possible.

Other BPM tools support collaborative process modeling, like IBM BPM Blueprint[2], ARISalign[3], Activiti[4] and processWave[5]. However, they only focus on the collaborative modeling of the process and do not allow users to model unknown constructs with natural language. In addition, only predefined process properties can be used to further describe processes. Since the flow structure is only stored in the process diagram and not as semantic descriptions, the search is rather limited.

The objective of SBPM [7] is to combine Semantic Web technologies with BPM. By using ontologies and semantic web services technologies more automation should be achieved in BPM lifecycle phases, namely process modeling, implementation, execution and monitoring. Wetzstein et al. [13] have specified requirements for SBPM; in the process modeling phase process models are semantically annotated to enable semantic-based discovery of process fragments. However, current SBPM approaches annotate the process elements by referencing existing ontology entities and do not provide users with means to model processes with self-defined semantic annotations in combination with natural language.

5 Conclusions and Future Work

We present an approach to support knowledge workers within enterprises in developing and formalizing processes by using formal semantics in combination with natural language and semantic annotations. We addressed the problem of process modeling by combining a semantic wiki (Semantic MediaWiki) with a process editor (Oryx Process Editor) to allow collaborative process development. By modeling processes with our semantic wiki-based process editor, we automatically get machine-accessible process semantics, which can be used for instance to validate the process model or to improve process search. By providing standardized RDF export, it can easily be integrated into existing approaches and

[2] http://www.lombardisoftware.com/
[3] http://www.arisalign.com/
[4] http://www.activiti.org/
[5] http://www.processwave.org/

enhance their functionality. In the future, we will investigate how automatic annotations of processes and natural language processing approaches can be used to detect inconsistencies between graphical and textual descriptions. In addition, we further evaluate our approach by involving various users form a company which collaboratively model informal processes.

References

1. van der Aalst, W.M.P., Weijters, A.J.M.M.: Process mining: a research agenda. Computers in Industry 53(3), 231–244 (2004)
2. Decker, G., Overdick, H., Weske, M.: Oryx An Open Modeling Platform for the BPM Community. In: Dumas, M., Reichert, M., Shan, M.C. (eds.) Business Process Management, LNCS, vol. 5240, pp. 382–385. Springer Berlin / Heidelberg (2008)
3. Dello, K., Nixon, L.J.B., Tolksdorf, R.: Extending the makna semantic wiki to support workflows. In: Lange, C., Schaffert, S., Molli, H.S., Völkel, M. (eds.) SemWiki. CEUR Workshop Proceedings, vol. 360. CEUR-WS.org (2008)
4. Dengler, F., Lamparter, S., Hefke, M., Abecker, A.: Collaborative Process Development using Semantic MediaWiki. In: Hinkelmann, K., Wache, H. (eds.) Wissensmanagement. LNI, vol. 145, pp. 97–107. GI (2009)
5. Erol, S., Granitzer, M., Happ, S., Jantunen, S., Jennings, B., Johannesson, P., Koschmider, A., Nurcan, S., Rossi, D., Schmidt, R.: Combining BPM and social software: contradiction or chance? J of Software Maintenance and Evolution: Research and Practice 22, 449–476 (2010)
6. Ghidini, C., Kump, B., Lindstaedt, S., Mahbub, N., Pammer, V., Rospocher, M., Serafini, L.: Moki: The enterprise modelling wiki. In: Aroyo, L., Traverso, P., Ciravegna, F., Cimiano, P., Heath, T., Hyvnen, E., Mizoguchi, R., Oren, E., Sabou, M., Simperl, E. (eds.) The Semantic Web: Research and Applications, LNCS, vol. 5554, pp. 831–835. Springer Berlin / Heidelberg (2009)
7. Hepp, M., Leymann, F., Domingue, J., Wahler, A., Fensel, D.: Semantic business process management: A vision towards using semantic web services for business process management. Proceedings of the 2005 IEEE International Conference on e-Business Engineering (ICEBE05) (Jan 2005)
8. Krötzsch, M., Vrandecic, D., Völkel, M., Haller, H., Studer, R.: Semantic wikipedia. J of Web Semantics 5, 251–261 (SEP 2007)
9. Recker, J., Safrudin, N., Rosemann, M.: How Novices Model Business Processes. In: Hull, R., Mendling, J., Tai, S. (eds.) Business Process Management, LNCS, vol. 6336, pp. 29–44. Springer Berlin / Heidelberg (2010)
10. Roure, D.D., Goble, C., Stevens, R.: The design and realisation of the virtual research environment for social sharing of workflows. Future Generation Computer Systems 25(5), 561 – 567 (2009)
11. Russell, N., Hofstede, A.H.M.T., van der Aalst, W., Mulyar, N.: Workflow Control-Flow Patterns: A Revised View. Tech. Rep. BPM Center Report BPM-06-22, BPM-center.org (2006)
12. Warren, P., Kings, N., Thurlow, I., Davies, J., Bürger, T., Simperl, E., Moreno, C.R., Gomez-Perez, J., Ermolayev, V., Ghani, R., Tilly, M., Bösser, T., Imtiaz, A.: Improving knowledge worker productivity - the ACTIVE integrated approach. BT Technology J 26(2) (Feb 2009)
13. Wetzstein, B., Ma, Z., Filipowska, A., Kaczmarek, M., Bhiri, S., Losada, S., Lopez-Cobo, J.M., Cicurel, L.: Semantic business process management: A lifecycle based requirements analysis. Proceedings of the Workshop SBPM 2007 (2007)

A.3 Collaborative Process Development Using a Semantic Wiki-based Process Editor

Collaborative Process Development Using a Semantic Wiki-based Process Editor

Frank Dengler
Karlsruhe Institute of Technology
Englerstr. 11
76131 Karlsruhe, Germany
frank.dengler@kit.edu

Denny Vrandečić
Karlsruhe Institute of Technology
Englerstr. 11
76131 Karlsruhe, Germany
denny.vrandecic@kit.edu

Elena Simperl
Karlsruhe Institute of Technology
Englerstr. 11
76131 Karlsruhe, Germany
elena.simperl@kit.edu

ABSTRACT

During their daily task, knowledge workers deal with processes which are often rather informal and mostly knowledge-intensive. They are rarely documented and subject to frequent changes. Nevertheless, it can be beneficial for the efficient organizational knowledge management to document and share such processes. Current state-of-the-art methods and tools for process development are found lacking for informal processes. We address this issue by combining graphical process modeling techniques with a wiki-based light-weight knowledge capturing approach and a background semantic knowledge base. By automatically translating the graphical process representation into formal semantics, the graphical process knowledge is made explicit and can be further processed. Our approach enables the collaborative creation of process descriptions with a graphical representation, formal semantics, and natural language. As a result, we provide a system that enables both novice users and experts to collaboratively work on the process descriptions: novice users can use graphical or textual descriptions, which then can be refined and formalized by experts.

Categories and Subject Descriptors

H.5.3 [**Information Systems**]: INFORMATION INTERFACES AND PRESENTATION—*Group and Organization Interfaces*

General Terms

Collaborative Process Development, Light-weight Knowledge Capturing, Web 2.0

1. INTRODUCTION

Knowledge workers deal with processes during their daily task. These can be business processes, formally defined within an organization, but also processes which are rather informal, rarely documented and mostly knowledge-intensive [22]. These processes can vary from person to person even when those involved are pursuing the same objective.

For instance a proposal of a consulting company can be seen as a result of knowledge-intensive, informal processes, collaboratively performed by a proposal team. Adding to the complexity and under the time constraints of a proposal many people with different expertise and roles have to be involved in the proposal development. Typically, there is one proposal manager who is responsible for the proposal and initializes the proposal development process by selecting the proposal team. The proposal team consists of knowledge workers with various experience, skills, and knowledge relevant for the proposal. Since no proposal is a copy of another one, the development processes can also deviate from each other. Core activities can be identified like *selecting proposal team* or *getting approval for pricing* but most activities, in which the content of the proposal is created, are distinct to a certain extend. Sections from previous proposals can be reused, others have to be adjusted to the customers and their requirements, or created from scratch. Much in the way the underlying activities are carried out depends, however, upon the proposal team member's expertise and previous experience, on tacit knowledge which is not recorded in formal procedures, but exists in the individuals' head, and in undocumented social communication and collaboration processes [4]. Therefore, the knowledge workers can be supported in creating, reusing, sharing, and also improving these informal, knowledge-intensive processes, typically realized through weakly-structured workflows [20].

We present an approach for the collaborative development of these rather informal processes. Nevertheless we use the term *process development* instead of *pro-*

cess modeling in order to emphasize their collaborative nature. Our methodology for collaboratively creating processes is based on the Knowledge Maturing Process Model [19]. The model structures five phases for the maturation of knowledge building upon each other: *expression of ideas*, *distribution in communities*, *formalization*, *ad-hoc learning* and *standardization*. In addition to being used for the maturation of content objects or ontologies for knowledge organization, the model can also be applied to other knowledge representations such as process models. Regarding the maturation of process models, the conceptual model proposes to start with individual task lists and routines. Task patterns can be derived from recurring tasks and shared between individuals. In a next step, a wider community of people is allowed to discuss, refine, enrich, and even formalize these procedures in a collaborative manner.

As a solution to support such a collaborative, distributed, and iterative process development, we combined Semantic MediaWiki (SMW) [11] with a graphical process editor, allowing the joint use of graphical representation, formal semantics, and natural language to describe processes. The created graphical representation of the process is automatically translated into a machine-processable format. Thus, the user can create semantic descriptions of processes by drawing the process diagram. External resources can easily be linked to the process by adding additional properties within the text on the wiki page of a process element. Whereas the graphical representation is also represented formally, textual descriptions can be made with or without the use of formal semantics. This characteristic makes our tool very powerful and unique compared to other approaches.

The remainder of this paper is structured as follows. Section 2 describes the requirements for collaborative process development derived from previous findings and an analysis of existing processes in industry. As a result of our requirement analysis, we present our approach to support collaborative process development and its implementation in Section 3. After applying our approach to collaborative proposal development in Section 4, we evaluate our approach by modeling existing processes in Section 5. Section 6 relates the elaborations of this paper to other research. Eventually, Section 7 concludes the paper and gives an outlook on future research.

2. REQUIREMENT ANALYSIS

To allow users modeling processes, we derive requirements from academia as well as real processes used in industry.

2.1 Requirements from research findings

In order to reuse, share, and collaboratively improve processes, they have to be externalized. Often, process mining techniques and tools [21] are used to capture and analyze activities performed by knowledge workers. But many tasks of knowledge workers are outside the grasp of mining algorithms: knowledge workers spend 14% of their work time in phone calls or meetings (as shown by a study of task switching and interruptions of knowledge workers [3]), or information required for a process activity can be exchanged between knowledge workers during a discussion at the water cooler.

Complementary to mining the activities of the knowledge worker, process modeling tools can be used to explicitly capture the processes. Each knowledge worker can model their own processes. The aggregate knowledge of a large group is supposed to improve the knowledge of one or a few experts. Therefore, it is equally important to provide knowledge workers and Web users with means to develop processes collaboratively. People with different levels of expertise are modeling processes. There are usually users involved, who are novice in process modeling. Recker et al. [17] have investigated how novices model business processes in the absence of tool support. Design representation forms chosen to conceptualize business processes range from predominantly textual, to hybrid, to predominantly graphical types. Process descriptions that combine graphical and textual types achieve a higher quality. Another survey analyzing the used modeling constructs of Business Process Modeling Notation[1] (BPMN), shows that in most BPMN diagrams, less than 20% of the BPMN vocabulary are regularly used and the most occurring subset is the combination of tasks and sequence flows [15].

2.2 Requirements from industry

For our analysis of processes used in industry we looked at 20 different processes from a large consultancy company. The processes are methodologies and reusable assets describing procedures to guide consultants (knowledge workers) in their daily work. They are defined by experts. The older versions are described in MS Word documents, which can be accessed from the company Intranet. The newer ones are directly stored as Flash and HTML files on the Intranet. A process can have subprocesses which are stored in a separate file and interrelated to each other via links. Each process documentation contains a short description, inputs and outputs, a flow diagram, and extensive textual descriptions of each process step. The process flow is expressed by a process picture created with a graphical software. It does not have a formal behavioral semantics. The textual descriptions are composed of detailed action instructions, links to other resources, and the party responsible for each step. All (sub)processes contain less than 10 steps. Only a few modeling constructs are used within the flow model, namely activities, sequence flows, conditions, and pools. The expressivity of the flow model is complemented by the textual descriptions, e.g. excep-

[1] `http://www.bpmn.org/`

tion handling is described in detail within the action instructions. The process search is rather limited to simple textual search as the process knowledge is not machine-processable. The browsing is also confined to the few links provided by the experts.

2.3 Requirements for collaborative process development

Based on the previous two sections, we derive the following requirements for collaborative process development:

- *Manual Modeling Support for Novice Users.* Novices in process modeling need manual modeling support, so that they can create and extend the processes without the assistance of an expert. If the user does not know the graphical representation of a process element, natural language can be used to describe it. In addition, the tool requires a rich user interface providing the user with means for interacting with processes in a highly intuitive manner. As a result, this leads to a trade-off between the expressivity offered to develop the formal process model and the usability of the tool.
- *Natural Language Support.* Novices are not familiar with all graphical representations of process elements. Thus, it can be beneficial to have natural language descriptions, that novices are able to follow processes.
- *Collaboration Support.* Users must be able to discuss process models asynchronously. Changes of the process model have to be tracked and users should be enabled to access the version history and to revert to previous versions. In addition, design rationales should be documented.
- *Structured Process Documentation Support.* The process models must be stored in a machine-processable documentation format, including additional properties linking to external resources. Users must be able to interlink between process descriptions and external resources to enable more sophisticated retrieval, browsing and navigation.

3. SMW-BASED PROCESS EDITOR

To address these requirements for collaborative process development, our approach combines wiki-based light-weight knowledge capturing with graphical process modeling functionality. Hence, users can develop process knowledge by using graphical descriptions, natural language, and formal semantics.

We extended our approach presented in [7] and integrated a graphical process editor with SMW. SMW extends the MediaWiki[1] software that runs the popular Wikipedia site. The extension combines Semantic Web technology with the collaborative aspects of wikis [12] to enable large-scale and inter-departmental collaboration on knowledge structures. Knowledge can be expressed by using natural language in combination with formal annotations allowing machines to process this knowledge. A special syntax is provided by SMW to define class hierarchies (categories and sub-categories) and semantic properties (properties and sub-properties) related to wiki pages. For instance a category *Process* can be added to a wiki page by adding [[**Category:Process**]] on the wiki page. For a property *has Successor*, expressing a successor relation between two wiki pages, the following syntax is used: [[**has Successor::<Name of wiki page>**]]. To access the formalized knowledge within wiki pages, SMW offers an inline query language (ASK syntax). The syntax for a query asking for all instances belonging to the category *Process* and their property *Short Description* is **{{#ask: [[Category:Process]] |?Short Description}}**. Without stating a specific output format, the query result will be displayed in a table on the corresponding wiki page. To make the formalized knowledge also available for other applications, SMW provides export functionality in RDF [2]

For our implementation we selected the Oryx Process Editor [5], an open source process editor, as the graphical process editor component. It can currently support various modeling languages such as BPMN, Event-driven Process Chain (EPC), Petri Nets, Workflow Nets, as well as Unified Modeling Language (UML), and can easily be extended to handle own process modeling languages.

SMW was extended to be compatible with the Oryx graphical editor, so that data can be exchanged between both. In addition, the graphical editor was extended to display and edit wiki pages from within its interface; as a consequence, users can directly access the corresponding wiki page within the process editor. The entered wiki text is rendered by using the parse method provided by SMW. Thus, the whole SMW syntax can be used including categories and properties. SMW ASK queries are executed and the results are displayed as well. Both the original entered text as well as the parsed wiki text are temporarily stored within the data model of the process editor as additional hidden properties. As illustrated in Figure 1, the process editor interface consists of different regions. As already mentioned the corresponding wiki page is displayed in the bottom of the editor. For our approach we only use a small subset of BPMN constructs. The available process elements are presented in the left region of the editor, namely tasks, sequence flow, parallel gateway, and data-based exclusive gateway. Users can easily add process elements to the process by dragging a process element from the left region and dropping it on the process diagram in the middle. Once the process is saved (by clicking on

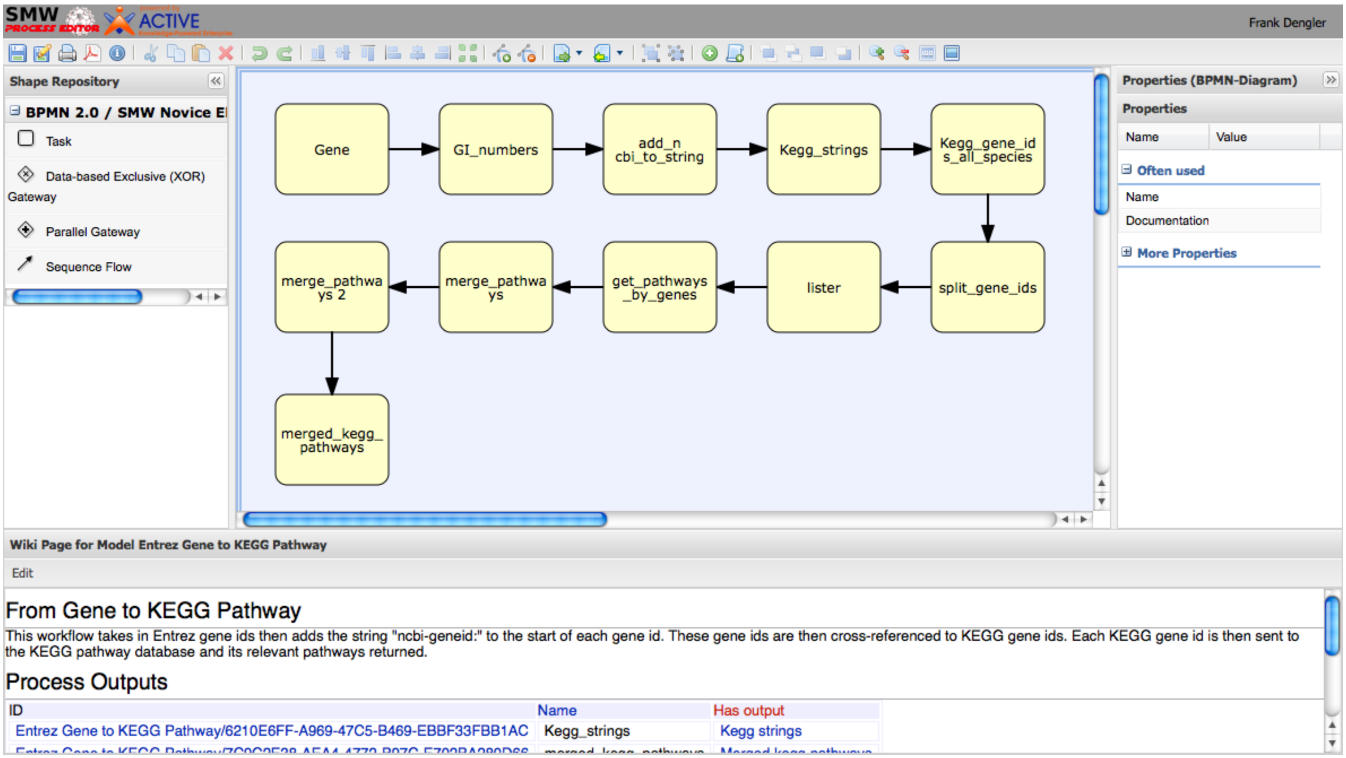


Figure 1: SMW Process Editor Screen Shot

the save button in the process editor), the process data and wiki pages belonging to the process are created or updated in SMW. The process elements are saved as subpages to the process summary page within the wiki. The process element wiki pages contain the textual descriptions and a fact box with all the stored properties. On the process summary page, the process diagram in SVG and a fact box are displayed (see Figure 2).

We support most of the *Basic Control-Flow Patterns* introduced in [18]. Every single process step (activity) is represented as a wiki page belonging to the category *Process Element* and linked via the properties *has Type* to the corresponding type (*Task*) and *Belongs to Process* to the corresponding process, represented as wiki pages themselves (process summary pages). An activity is the basic element of our process. Depending on the granularity level of the process this can vary from atomic activities, such as *open a web page*, to activities describing a whole subprocess. To express the control flow of the process, we use edges in the diagram and special predefined process elements (gateways). If an element has a successor, we draw an edge from the activity to the successor activity in the diagram and store this with the additional property *has Successor* on the corresponding wiki page in SMW. For more successors executed in parallel (parallel-split pattern), a Parallel Gateway is used in between the activities. An activity can have several successors, but only one has to be selected and executed (multi-choice pattern). Therefore we use the Data-based Exclusive Gateway without conditions. The Data-based Exclusive Gateway with conditions is used to split based on a condition (exclusive-choice pattern). A condition is stored as a many-valued property.[2] The distinction between the synchronization pattern and the simple-merge pattern is realized by using the Parallel Gateway and the Data-based Exclusive Gateway the other way round to merge different branches of a process.

All properties of the process elements are also available in SMW. They are stored as SMW properties with their corresponding value. Thus all the process properties can be accessed within SMW and queried. For example, these properties can be displayed in a fact box on the corresponding wiki page as shown in Figure 2. Links to the corresponding wiki pages are automatically added to the SVG graphic, which enable the user to navigate through the process in the wiki.

A new tab *edit with editor* has been added to the process wiki for editing an existing process. The tab automatically appears on pages belonging to the categories *Process* and *Process Element*. The tab contains the graphical process editor with the process model. Even

[2]Many-valued properties in SMW are implemented as records, see `http://semantic-mediawiki.org/wiki/Type:Record`

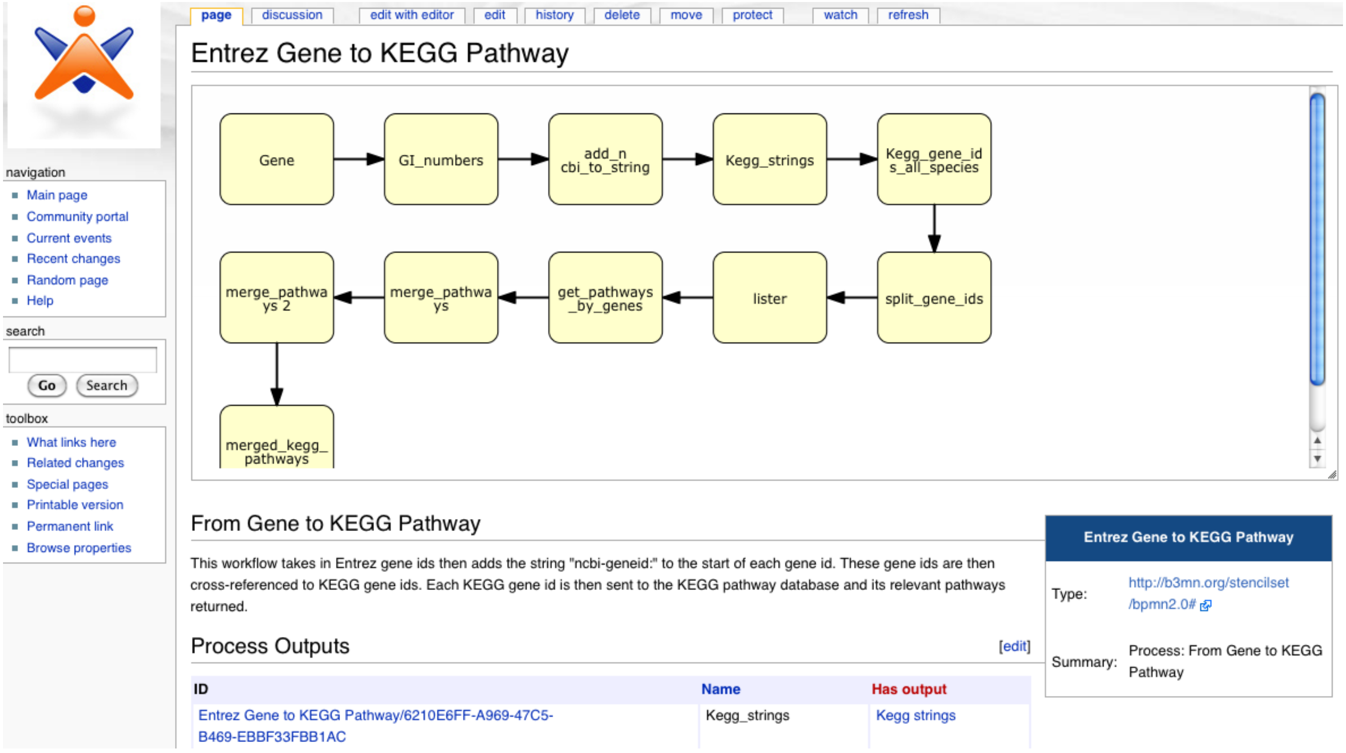


Figure 2: Example Process in SMW (process summary page with fact box)

though technically the process editor runs on a separate Tomcat server, the SMW authentication is also used to control the access on the process model within the process editor, providing a seamless experience between the two different components.

4. APPLICATION OF OUR APPROACH

Our approach supports the collaborative proposal development scenario in different ways. Since many people with different expertise and roles are involved, everybody can contribute in acquiring the proposal development processes. Depending on the expertise and the available time, proposal team members can thereby alter the process to take advantage of it (e.g. contributing only to parts of the proposal document for which they have the necessary knowledge). Additionally, previous proposal documents can be linked to specific process activities using semantic properties. Thus relevant proposals can be filtered out by using SMW inline queries. Also less experienced proposal team members can profit from the process wiki, because they can look up and follow the developed processes. The formalized processes can also be used as a basis for the input in a process execution engine, e.g. accessing the RDF export interface of SMW via the process execution tool. The advantages of our approach are:

- The combination of graphical representation, natural language, and formal semantics allows collaborative modeling for both novices and experts. Textual and graphical elements can be used interchangeably and complementarily. If the user does not know the graphical representation of a process element, natural language can be used to describe it on the corresponding wiki page.
- This approach uses an extendible underlying schema. Users can introduce their own properties in the wiki by using the SMW property syntax on the process element wiki page. Thus, processes can be linked to existing knowledge structures (e.g. what input documents are used, can be made available and processable for computers).
- Standard wiki features can be used for process modeling, like versioning, watch lists, reverting, etc.
- SMW acts as a process repository where processes and their process semantics are stored. Process knowledge can be linked, queried and displayed on process pages and on other wiki pages. Structured data can improve the search for processes (e.g. [14]).

5. EVALUATION

We conducted a usability test with the process wiki. We used four textual process descriptions specifying internal How-tos of the university institute AIFB and five service process descriptions (GR01, GR02, GR03, GR05

Question	N	AVG	DEV
Overall, I am satisfied with how easy it is to use this system	8	5,1250	1,5526
It was simple to use this system	8	5,2500	1,5811
I can effectively complete my work using this system	8	5,3750	1,4079
I am able to complete my work quickly using this system	8	4,5000	1,6036
I am able to efficiently complete my work using this system	8	5,3750	0,9161
I feel comfortable using this system	8	4,8750	1,5526
It was easy to learn to use this system	8	5,1250	1,9594
I believe I became productive quickly using this system	8	5,5000	1,5119
The system gives error messages that clearly tell me how to fix problems	5	2,4000	1,6733
Whenever I make a mistake using the system, I recover easily and quickly	8	5,2500	1,4880
The information (such as online help, on-screen messages, and other documentation) provided with this system is clear	6	4,3333	2,2509
It is easy to find the information I needed	7	4,7143	1,7995
The information provided for the system is easy to understand	7	5,1429	2,2678
The information is effective in helping me complete the tasks and scenarios	7	5,1429	1,9518
The organization of information on the system screens is clear	8	5,0000	1,8516
The interface of this system is pleasant	8	5,8750	1,8851
I like using the interface of this system	8	5,7500	1,0351
This system has all the functions and capabilities I expect it to have	5	5,2000	1,4832
Overall, I am satisfied with this system	8	5,2500	1,1650

Table 1: Evaluation Results

and IT04) from the COCKPIT Project [10]. The processes were modeled by eight students with different experience levels in using Semantic MediaWiki and in modeling processes. Only half of the students have ever used a process modeling tool. After a brief introduction of the basic functionality of our tool, each student was asked to model three assigned processes with it without any hint. The results can be found in our demo wiki[3].

The students modeled the processes in different ways. While all students used task and sequence flows, only four students used the gateway elements. Conditions were sometimes expressed in the textual description, in the graphical representation, or in both representations. Additional semantic properties were introduced by half of the students. The generated graphical representations of processes including a conditional split were not modeled semantically correct according to BPMN due to the lack of gateways.

At the end, each student had to fill out a Computer System Usability Questionnaire (CSUQ) [13], by rating 19 statements from 1 to 7 with respect to our tool, where 7 is *strongly agree* and 1 is *strongly disagree*. In addition, the questionnaire was extended with questions about previous process modeling experiences and free text questions about most positive and negative aspects. The results of the CSUQ can be found in Table 1. The overall assessment of the students about the usability of the tool was very positive. Only the quality of the error messages was ranked negative in average. As a consequence, we will improve them in further releases.

[3]The demo wiki can be accessed via `http://oryx.f-dengler.de` (Username: ProcessTester – Password: active!)

The students criticized that all information entered as wiki text was deleted when they clicked on another task while the wiki text editor was not closed using the *End Edit Mode* button. In addition they recommended that the system should provide duplication of pages within the modeler, in order to enable the faster modeling of similar processes. Some students needed more time in the beginning to get familiar with the wiki syntax. As positive aspects most of the students explicitly mentioned the intuitive usability. It was mentioned that it was easy to handle, because the UI was perceived as simple and clear. It was not overloaded with unnecessary features yet it provided the functions needed to complete the task.

The results of our evaluation shows that processes can be modeled by using our approach, but semantic correctness cannot be assured. Also users having no experiences in process modeling were able to model processes with the tool.

6. RELATED WORK

The work presented in this paper is related to the following streams of research (1) social software for process management, and (2) collaborative process modeling.

There exist other approaches to manage processes with social software [8, 16], but none of them supports process knowledge capturing with graphical representation, formal semantics, and natural language. To support workflows, Dello et al. [6] have extended the Makna Semantic Wiki by integrating the workflow engine jBPM. It enables the coordination of interactions within a wiki system, but does not support the collaborative creation of the workflow.

The Moki enterprise modeling wiki [9], by contrast, allows the collaborative development of processes. However, the tool does not translate the collaboratively created graphical process descriptions into formal semantics. Therefore, queries concerning process semantics, such as *"show me all process activities resulting in an approval activity"* are not possible. The graphical interface does not allow the user to read and edit the wiki pages of the single process elements.

Other BPM tools and Web communities allow collaborative business process modeling, like Activiti[4] and BPMN-Community,[5] both based on Oryx [5], or processWave[6] based on Google Wave. There are also commercial tools, like IBM BPM Blueprint[7] and ARISalign[8]. However, they only focus on the collaborative development of the process model and require modeling exper-

[4]`http://www.activiti.org/`
[5]http://www.bpmn-community.org/
[6]http://www.processwave.org/
[7]`http://www.lombardisoftware.com/`
[8]`http://www.arisalign.com/`

tise. Novice users cannot model unknown constructs with natural language. Only predefined process properties can be used to further formalize processes. If a property is not included in the process modeling language, it cannot be used. Since the flow structure is only stored in the process diagram and not as semantic descriptions, the search is rather limited compared to our approach.

7. CONCLUSION

In this paper, we present an approach to support knowledge workers within enterprises in developing and formalizing processes by using graphical representation, formal semantics, and natural language. We addressed the problem of process modeling by combining a wiki-based light-weight knowledge capturing approach (Semantic MediaWiki) with a process editor (Oryx Process Editor) to allow collaborative process development. In order to store processes developed in the graphical process editor, we have connected Semantic MediaWiki and the Oryx Process Editor. One advantage of this approach is the support of users with different levels of expertise in process modeling. If a user does not know a required process element, natural language can be used to describe it. The combination of natural language with formal semantics allows collaborative modeling for both novices and experts. Further process knowledge can easily be connected to process descriptions by semantic annotations within wiki pages. Since all process knowledge is stored within SMW and is also annotated with semantic properties, semantic queries can be used to filter relevant process information for the users. In addition, standard wiki features can be used for process modeling, like versioning, watch lists, reverting, export, etc.

We acknowledge that our study bears certain limitations. First, only eight students were involved in our evaluation, but since all of them had only few experiences with modeling processes, we can assume based on our results and their feedback, that novices can intuitively use our tool. Second, the semantic correctness of the models cannot be assured. As our approach is designed for collaborative process modeling, normally also process modeling experts are involved. They can correct the errors introduced by novices in order to assure semantic correctness. Third, we did not evaluate how people work collaboratively on developing processes, because we did not expect significant results with our eight test persons. Thus we focused our evaluation on the functionality and the usability of our approach.

All in all, we have shown that our approach for collaborative processes development can be used intuitively. The overall usability was approved by the test persons. It provides enough process modeling capability that existing processes can be modeled with it. By modeling processes with our semantic wiki-based process editor we automatically get machine-accessible process semantics, which can be used, e.g., to validate the process model. By providing standard exchange formats (like RDF) it can easily be integrated into existing semi-automatic process acquisition approaches and enhance their functionality.

In the future we will further validate our approach by involving various users collaboratively constructing processes. We will also investigate the appropriate expressivity of the process language for different modeling skills.

8. ACKNOWLEDGMENTS

Research reported in this paper was funded by the European Commission through the IST projects ACTIVE (ICT-FP7-215040).

9. REFERENCES

[1] D. J. Barrett. *MediaWiki*. O'Reilly Media, Inc., 1 edition, 2008.

[2] D. Beckett. RDF/XML syntax specification (Revised). Technical report, W3C, 2004.

[3] M. Czerwinski, E. Horvitz, and S. Wilhite. A diary study of task switching and interruptions. In *CHI '04: Proceedings of the SIGCHI conference on human factors in computing systems*, pages 175–182, New York, NY, USA, 2004. ACM.

[4] T. H. Davenport. *Thinking for a Living: How to Get Better Performance and Results from Knowledge Workers*. Harvard Business School Press, Boston, Mass., illustrate edition, 2005.

[5] G. Decker, H. Overdick, and M. Weske. Oryx An Open Modeling Platform for the BPM Community. In *Business Process Management*, volume 5240 of *Lecture Notes in Computer Science*, pages 382–385. Springer Berlin / Heidelberg, 2008.

[6] K. Dello, L. J. B. Nixon, and R. Tolksdorf. Extending the makna semantic wiki to support workflows. In *SemWiki*, volume 360 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2008.

[7] F. Dengler, S. Lamparter, M. Hefke, and A. Abecker. Collaborative Process Development using Semantic MediaWiki. In *Proceedings of the 5th Conference on Professional Knowledge Management*, pages 97–107, 2009.

[8] S. Erol, M. Granitzer, S. Happ, S. Jantunen, B. Jennings, P. Johannesson, A. Koschmider, S. Nurcan, D. Rossi, and R. Schmidt. Combining BPM and social software: contradiction or chance? *J of Software Maintenance and Evolution: Research and Practice*, 22:449–476, 2010.

[9] C. Ghidini, B. Kump, S. Lindstaedt, N. Mahbub, V. Pammer, M. Rospocher, and L. Serafini. Moki: The enterprise modelling wiki. In *The Semantic Web: Research and Applications*, volume 5554 of *Lecture Notes in Computer Science*, pages 831–835. Springer Berlin / Heidelberg, 2009.

[10] C. Koutras, S. Koussouris, P. Kokkinakos, and D. Panopoulos. Cockpit public services scenarios. Technical Report D1.1, COCKPIT Project, June 2010.

[11] M. Krötzsch, D. Vrandecic, M. Völkel, H. Haller, and R. Studer. Semantic wikipedia. *J of Web Semantics*, 5:251–261, SEP 2007.

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

[13] J. R. Lewis. Ibm computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7:57–78, 1995.

[14] Z. Liu, Q. Shao, and Y. Chen. Searching workflows with hierarchical views. *PVLDB*, 3(1):918–927, 2010.

[15] M. z. Muehlen and J. Recker. How Much Language Is Enough? Theoretical and Practical Use of the Business Process Modeling Notation. In *Advanced Information Systems Engineering*, volume 5074 of *Lecture Notes in Computer Science*, pages 465–479. Springer Berlin / Heidelberg, 2008.

[16] S. Nurcan and R. Schmidt. The Third Workshop on Business Process Management and Social Software (BPMS'10) as part of BPM 2010, 2010.

[17] J. Recker, N. Safrudin, and M. Rosemann. How Novices Model Business Processes. In *Business Process Management*, volume 6336 of *Lecture Notes in Computer Science*, pages 29–44. Springer Berlin / Heidelberg, 2010.

[18] N. Russell, A. H. M. T. Hofstede, W. van der Aalst, and N. Mulyar. Workflow Control-Flow Patterns: A Revised View. Technical Report BPM Center Report BPM-06-22, BPMcenter.org, 2006.

[19] A. Schmidt, K. Hinkelmann, T. Ley, S. Lindstaedt, R. Maier, and U. Riss. Conceptual Foundations for a Service-oriented Knowledge and Learning Architecture: Supporting Content, Process and Ontology Maturing. In *Networked Knowledge - Networked Media: Integrating Knowledge Management, New Media Technologies and Semantic Systems*. Springer, 2009.

[20] S. Schwarz, A. Abecker, H. Maus, and M. Sintek. Anforderungen an die Workflow-Unterstützung für wissensintensive Geschäftsprozesse. In *CEUR-WS Proceedings Geschäftsprozessorientiertes Wissensmanagement – Von der Strategie zum Content*, 2001.

[21] W. M. P. van der Aalst and A. J. M. M. Weijters. Process mining: a research agenda. *Computers in Industry*, 53(3):231–244, 2004.

[22] P. Warren, N. Kings, I. Thurlow, J. Davies, T. Bürger, E. Simperl, C. R. Moreno, J. Gomez-Perez, V. Ermolayev, R. Ghani, M. Tilly, T. Bösser, and A. Imtiaz. Improving knowledge worker productivity - the ACTIVE integrated approach. *BT Technology J*, 26(2), Feb. 2009.

[end of document]