

SUPPORT KNOWLEDGE INTENSIVE WORK WITH SEMANTIC TECHNOLOGIES

Roman Brun, School of Business, Institute for Information Systems, University of Applied Sciences Northwestern Switzerland FHNW, Riggengbachstrasse 16, CH-4600 Olten, Switzerland, roman.brun@fhnw.ch

Andreas Martin, School of Business, Institute for Information Systems, University of Applied Sciences Northwestern Switzerland FHNW, Riggengbachstrasse 16, CH-4600 Olten, Switzerland, andreas.martin@fhnw.ch

Abstract

This paper introduces an approach to support knowledge intensive work with semantic technologies by semantically enriching processes and then applying reasoning techniques on them. Further, the proposed use cases show the abilities of semantically enriched processes. Finally the paper introduces a possible architecture of a system which facilitates the knowledge intensive work with semantic technologies.

Keywords: Process Reasoning, Demonstrator, NEPOMUK, Knowledge Intensive Work.

1 INTRODUCTION

Nowadays business process management and workflow management systems become standard to support users in their daily work. The systems and technologies users work with are normally not integrated. But many activities in these processes require additionally to the explicit process and functional knowledge also explicit knowledge stored in other systems as well as implicit knowledge which is often only in heads of people. To address the problem of isolation, our approach is to combine processes (the organizational environment) with a semantic desktop (the personal environment). A semantic desktop gives the users the possibility to store their digital information like, documents, contacts, etc. semantically. This means, the data is accessible and queryable as RDF graph. (Sauermaun et. al, 2005) Further on, the elicitation of implicit knowledge is supported by using a semantic desktop through sharing, tagging and retrieval functionalities. In other words: "The Semantic Desktop is an enlarged supplement to the user's memory" (Sauermaun et. al, 2005).

Section 2 describes the research in the state of the art. How to integrate a process ontology into a social semantic desktop (e.g. NEPOMUK) is mentioned in section 3. Then, a use case is introduced in section 4 with whose reasoning possibilities are depicted. Challenges of integrating the scenario into the chosen environment are explained in section 5 which also indicates the future work in section 6, before concluding in the last section.

2 RESEARCH IN STATE OF THE ART

GNEWSIS (Sauermaun, 2003) is one of the first reference implementation of a semantic desktop. This approach provides the main functionalities of a semantic desktop. The semantic desktop is defined by (Sauermaun et. al 2005) as follows: "A Semantic Desktop is a device in which an individual stores all his/her digital information like documents, multimedia and messages. These are interpreted as Semantic Web resources, each is identified by a Uniform Resource Identifier (URI) and all data is accessible and can be queried as RDF graph. Resources from the web can be stored and

authored content can be shared with others. Ontologies allow the user to express personal mental models and form the semantic glue interconnecting information and systems. Applications respect this and store, read and communicate via ontologies and Semantic Web protocols. (Sauermann et. al 2005)”

The NEPOMUK (Tudor et al., 2007) project enhances the semantic desktop approach with social aspects. The aim of the NEPOMUK project was to come up with social semantic desktop including Web 2.0 elements.

NEPOMUK, whose goal is to "empower individual knowledge workers to better exploit their personal information space and to maintain fruitful communication and exchange within social networks across organizational boundaries." (NEPOMUK, 2008) offers a framework which can be accessed, used and enhanced. The major objective of the SUPER (Hepp et al., 2005) project is "to raise Business Process Management (BPM) to the business level, where it belongs, from the IT level where it mostly resides now [...] which requires that PM is accessible at the level of semantics of business experts." (SUPER, 2009). An outcome of this project is a semantic description of the Business Process Modeling Notation (BPMN) (OMG, 2008), sBPMN.

For proof of concept of the approach, the NEPOMUK framework is utilized to integrate processes, similarly as it has been done with KASIMIR (Grebner, 2008) for tasks. To describe processes, the sBPMN ontology of the SUPER project is used as reference. This work is also related to the MATURE approach (Schmidt, 2005). The approach of this paper can be used to trigger and push the mature process by providing refined and user observed processes.

3 INTEGRATION OF SEMANTICALLY DESCRIBED PROCESSES INTO THE SOCIAL SEMANTIC DESKTOP

In a process there exist different activities with respect to the degree of automation and complexity. As there are already various approaches and technologies available to support automated activities with few complexity (as e.g. workflow management systems or (BPEL, 2003)), the approach supports the non-automated and knowledge intensive activities (Feldkamp et al., 2007) by using the semantic technology and applying standard reasoning functionalities. As knowledge intensive activities often involve human interaction, they are also already partly support by (BPEL4People, 2007) or (WS-HumanTask, 2007).

The state of the art shows that considerable work has already been done in semantically enriching processes (Hepp et al., 2005) and also in the development of a semantic desktop (Tudor et al., 2007). The approach is to combine these two research fields through integrating semantically described processes into a semantic desktop by using already existing technologies.

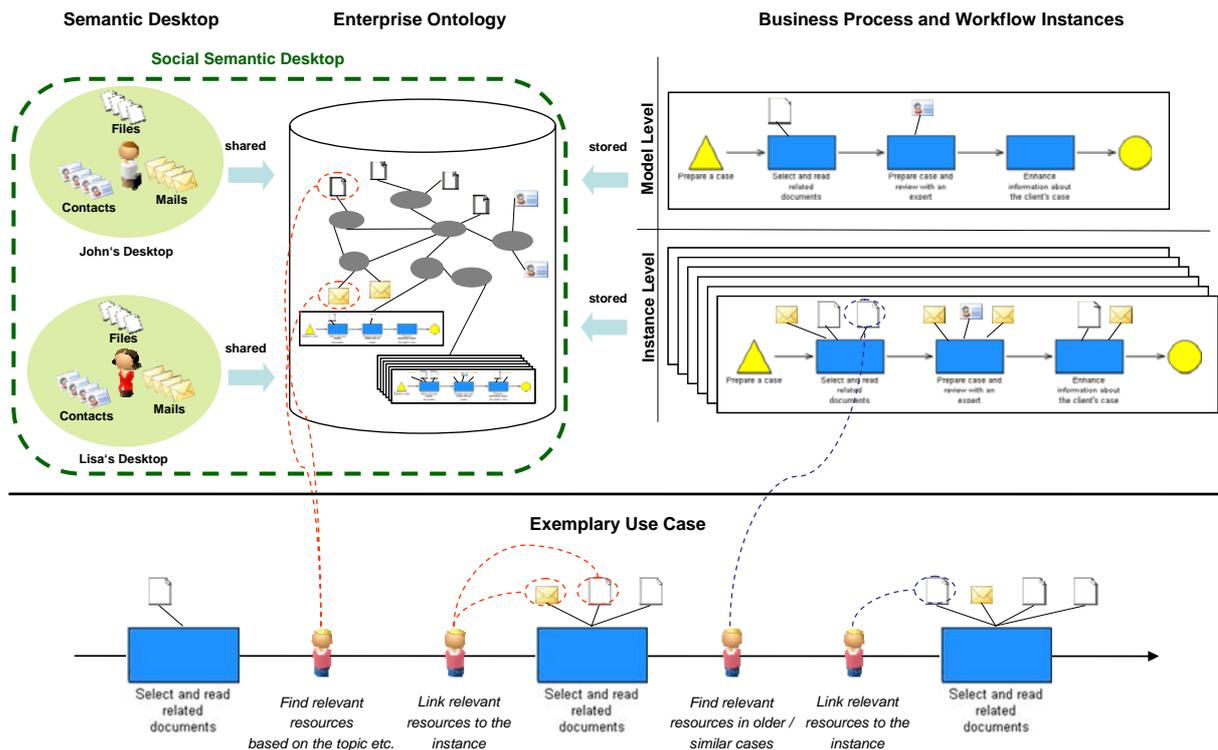


Figure 1: Architecture overview and exemplary use case

Figure 1 gives an overview over the architecture of the integration approach. The heart of this approach is the enterprise ontology which contains the accessible knowledge of an organisation. This enterprise ontology consists of resources (files, contacts, mails, etc.) shared by the employees. They use the possibility of the (personal) semantic desktop for annotating and sharing files. The enterprise ontology further consists of business process models and instances. This requires the semantic description of these processes and instances and consequently the enhancement of the NEPOMUK ontology as we used it to test our approach.

The exemplary use case (see Figure 1) shows the usage of the proposed approach. Let's assume that Michael has to perform the "Select and read related documents" task (see details in chapter 4 The Use Case (process & instances)). Michael is searching for relevant documents (and other resources) in the enterprise ontology by using the semantic desktop query tools (see chapter 4.3 Reasoning). An example for such a query could be: find documents which are related to a certain topic. If Michael has found some useful and relevant documents, he can decide to link these resources to the actual instance. This gives the possibility to use these resources in further tasks or instances. Though the integration of process and instances into an enterprise ontology, it is now possible for Michael to retrieve relevant resources in historical process instances which are similar (e.g. have the same topic) to the current case.

The following chapter shows the implementation and usage of semantically described processes and instances in the NEPOMUK framework.

3.1 The process ontology and its integration

For the approach a simple ontology to describe processes (see Figure 2) has been designed. The ontology was designed based on the use case needs and is kept very general and not intended to be exhaustive. Also the predicates do not exist in the indicated ontologies but have rather been named in order to better understand the relationships between the classes. In the following, the used classes and

relationships are explained. They have been chosen based on the importance for the work and the reasoning that has been conducted.

A **process** has relations to tasks (`processHasTask`), persons (`processIsRelatedToPerson`), topics (`processHasTopic`) and artefacts (`processHasArtefact`); to keep a clear structure there is a distinction between `hasProcessArtefact` and `hasInstanceArtefact` in the implementation. A **task** is related to persons (`taskIsRelatedToPerson`), artefacts (`taskHasArtefact`) and to other tasks (`hasPredecessorTask`, `hasSuccessorTask`) to describe the order of them. An **artefact** can be a document, website, person, etc. For the approach documents (`artefactIsaDocument`) have been considered.

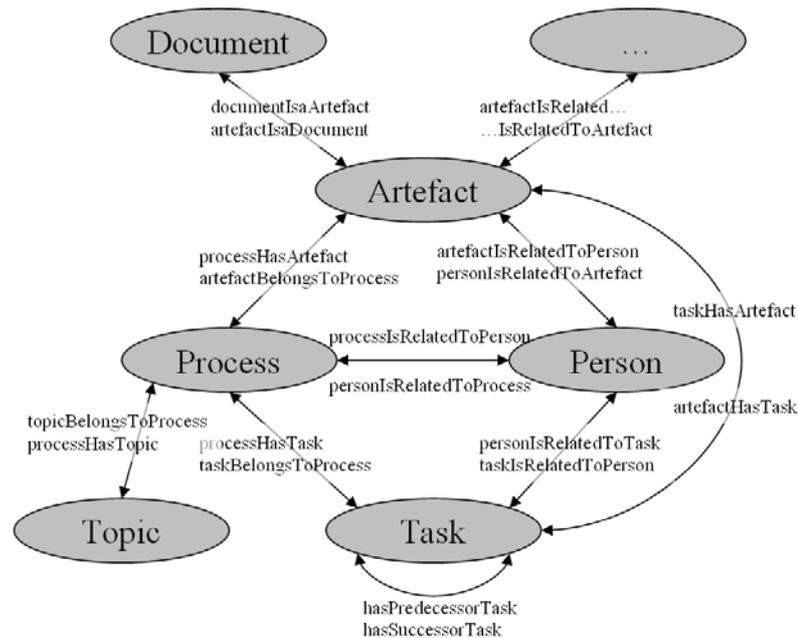


Figure 2: The process meta-model ontology

3.2 Combination of already existing ontologies

The goal was to link existing ontologies together in a way they support the approach and extending it with new classes where needed, but not to create new ones if possible. Figure 3 shows that the class related to processes is taken from the sBPMN ontology (<http://www.ip-super.org/ontologies/process/sbpnm/v2.0>), the task from the TMO ontology (<http://www.semanticdesktop.org/ontologies/2008/05/20/tmo>), the class artefact did not exist yet, and any further information element can be taken from the NIE ontology (<http://www.semanticdesktop.org/ontologies/2007/01/19/nie>).

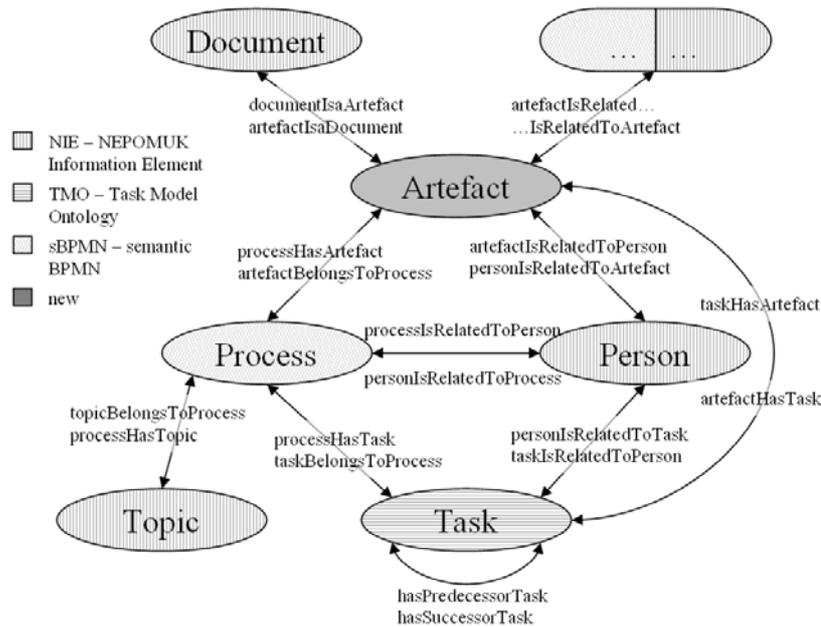


Figure 3: Ontologies combination

4 THE USE CASE (PROCESS & INSTANCES)

The use case represents a process of career guidance organization. Specially trained Personal Advisers (P.A.s) who are based in schools, colleges, at special access points and so on help (young) people with all sorts of personal issues, including employment and training. The following figure 4 represents a first process in the whole career guidance workflow. It is kept very general and consists of three activities to prepare career guidance. There are two documents related to the first activity. During runtime further documents, topics and further relations might be related to process instances.

For a useful use of NEPOMUK the example test data has been integrated as described in <http://dev.nepomuk.semanticdesktop.org/wiki/TestData>.

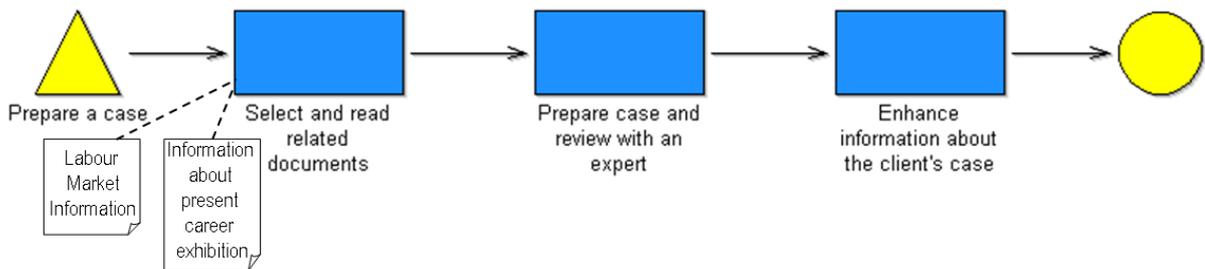


Figure 4: The process "Prepare Career Guidance Case" (PCGC)

- First, the Personal Advisor (P.A.) has to select and read documents (Labour Market Information & Information about present career exhibition) which are related to the request. The P.A. might also read additional documents to the given one's and add them if they seem to be relevant for him/her.
- The P.A. prepares himself/herself for the case and reviews the preparation with an expert, selected by himself/herself.
- As a last step, further information which is relevant for the case is added in order to be optimally prepared for the meeting with the client.

4.1 The instances

To go on with the use case, we assume that the following two instances have been executed. During run-time, additional documents have been added to activities and experts for reviewing have been selected as described below.

4.1.1 Instance PCGC1

Topics which are related to the instance:

- Plumber
- Young woman

Activity: "Select and read related documents"

- Additionally added documents in this activity:
 - Information about qualification needs for getting a plumber
 - Statistical reports about young women's profession

Activity: "Prepare case and review with an expert"

- The following expert has been selected:
 - Lars Bender

4.1.2 Instance PCGC2

Topics which are related to the instance:

- Gardener
- Apprenticeship

Activity: "Select and read related documents"

- Additionally added documents in this activity:
 - Average earnings of gardener in education
 - Article "A day in the life of Beni Oudo, gardener"

Activity: "Prepare case and review with an expert"

- The following expert has been selected:
 - Bernadette Hulgy

4.2 Adding the process and instances to NEPOMUK

In order to apply the use case, the process model and the instances must first be added to NEPOMUK. As aforementioned, the existing ontology has been enhanced by creating an appropriate N3-ontology (the process meta-model) and integrating it. To create process models, subclasses of the meta-model and instances of the subclasses have to be created in NEPOMUK.

Unfortunately, in the present NEPOMUK version we used, there is no inheritance possible when using the subClassOf relationship. Therefore the instances have to be built from scratch according to the subclass (the process) they belong to. However, this has no direct negative influence on the reasoning as any information is available.

4.3 Reasoning

Let us assume that an instance of another process "Prepare an information brochure for young woman in technical jobs" is running. For an employee mainly working on the creation of information brochures for customers it might be interesting to know who is an expert in the field of young woman working in such typical "male jobs" and contacting him/her. Therefore the employee enters the search

terms "young woman male jobs". The system searches for any process instance with such a topic or a similar one. It realizes that an instance of a process with the topic "Plumber" exists. As the topic "Plumber" is a subclass of the topic "male job", this relation can be inferred. In such a way, the system finds the related expert of the instance and suggests him/her to the employee.

This shows that through ontological stored data, related information can be found which cannot be retrieved in conventional systems. Therefore, employees working in different departments on different processes are easily able to find each other and exchange information. Consider the idea of the semantic desktop where beside process information any other artefact (emails, contacts, bookmarks, etc.) can be integrated. This offers an overall information retrieval by using reasoning techniques. Furthermore, the sharing functionalities of the social semantic desktop framework (NEPOMUK) could give the possibility to access or retrieve the organisational knowledge at a central point (this can be seen as an enterprise ontology storage).

NEPOMUK already offers the possibility to search through ontologically stored information. Let us search for any process instance with the topic "Male job". In the Structured Query Builder the needed classes (subjects and objects) and relationships (predicates) can be chosen and also free text can be entered, including wildcard searches (see Figure 5). These search terms are translated into SPARQL queries which then are executed against the RDF Repository to return the result.



Figure 5: Structured query for a process with the topic Male Job in NEPOMUK

The result (see Figure 6) is then given and can also be browsed in the side results (see the right hand side of the screenshot).

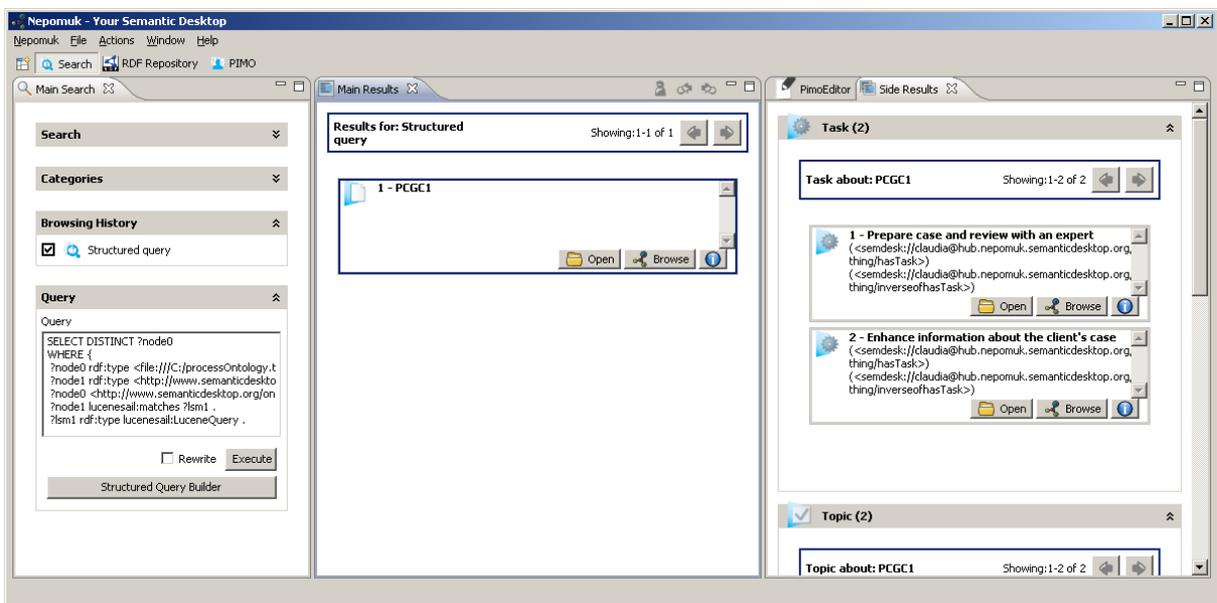


Figure 6: Result of search for a process with the topic Male Job

Beside this example, further reasoning and uses cases can be thought as for example: Reasoning with much more artefacts like website, person, bookmarks, etc. besides documents), comparing historical cases, reasoning on tags, and so on.

5 CHALLENGES TO OVERCOME

5.1 Level comparison

Regarding (process) modelling, there normally exist the four levels meta²-model, meta-model, model and instance (Geisler et al., 1998), whereas the later three are relevant for the approach. Figure 7 shows these levels on the left-hand side. In the centre the ontology and on the right-hand side its class- and instance-level is shown. The figure illustrates that it is not possible to distinguish between the metamodel- and the model-level as only the class-level exists.

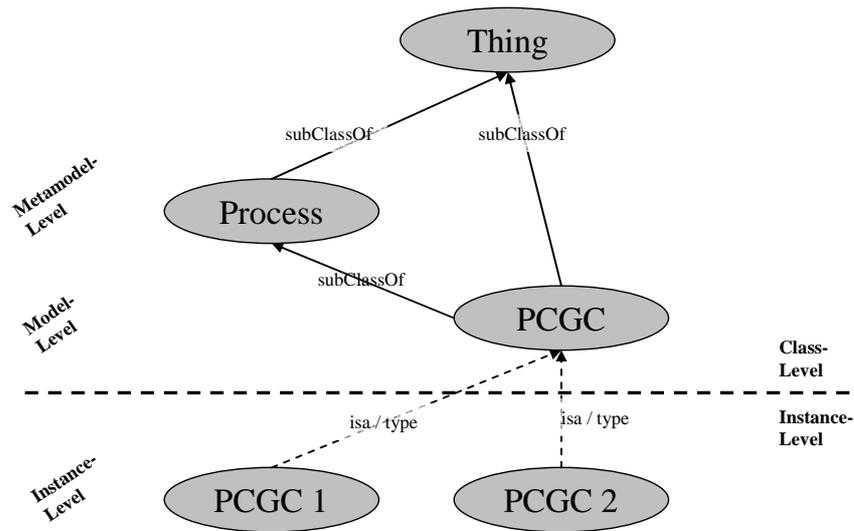


Figure 7: Comparison of standard modelling- and ontology- levels

5.2 Process specific needs

A further challenge that has to be mentioned is related to several process specific mannerisms. In a process, in general a task follows after the other. How can that be indicated? Our approach handled this by introducing the predicates `hasPredecessorTask` and `hasSuccessorTask` (see Figure 3). Next would be to handle branching or parallelism in a process. This means that it has to be ensured for parallelism that a task can have several predecessors but not all of them must have been performed to continue with the task; for parallelism it has to be checked at a certain point that all predecessor tasks have been executed.

However, there might be more challenges regarding process specific needs when going into further detail of process specific mannerisms, which have not been discovered yet. But it shows that NEPOMUK seems not to offer the best framework for integrating processes. One approach could be to use a semantic desktop and a workflow system in combination (see 6.1 Vision).

6 FUTURE WORK

To increase the reasoning possibilities, a next step would be to enhance and complete the process ontology. This would require the acceptance in the community by passing the standard review scenario. The modelling of processes in a tool like WSMO-Studio (Dimitrov et al. 2007) or ATHENE (Hinkelmann et al., 2007) and directly integrating the models into NEPOMUK or any other framework would be a further step.

However, the challenges to overcome (see chapter 5) and the according explanation why NEPOMUK might not be appropriate for our purposes forced not to go on with the approach in NEPOMUK at this time and looking for another framework to further develop the idea of our approach.

6.1 Vision

The following screenshot (Figure 8) describes the vision how the interfaces could look like. The "Semantic Workflow Workbench" is divided in two main parts, the "Workbench" and the "Reasoning Widgets" on the right-hand side.

The **Workbench** itself contains the elements process overview, task description, task documents, and further the case element with its sub elements. All these elements are generic and their composition can be defined in model time as well as adapted during runtime.

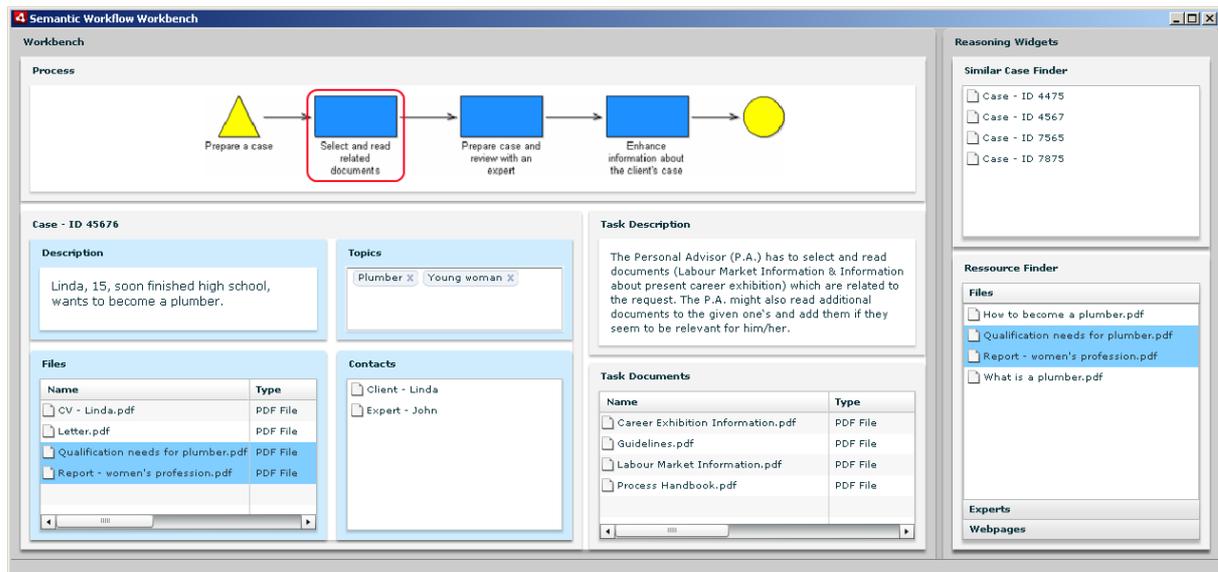


Figure 8: Semantic Workflow Workbench

The *process element* shows the actual position of the task in the workflow and the *task description* gives instructions about the task. The user has also access to relevant *task documents* like guidelines or a process handbook etc. The system offers the possibility to add additional task documents.

The *case element* consists of several sub elements which will be filled with information at runtime (given by the instance the user is currently working on). The *description* shows the actual case information. The *topic field* contains the relevant topics for the case. When entering a new topic, the field suggest already known topics. The user can find files related to the actual case in the *files field*. It is also possible to add additional files to the case. The *contacts field* offers information about all relevant contacts for the running instance and has the same features as the files field.

Whereas the information of the case element is filled at runtime, the information for all other elements in the "Workbench" part will be defined at modelling time and can be adapted during runtime.

An interesting feature is the possibility to use **Reasoning Widgets**. As seen in chapter 4.3 (Reasoning), the query creation is not easy. Therefore the paper introduces the usage of reasoning widgets which offer an end-user friendly interface. On widget basically offers one functionality, e.g. similar case finder, similar resource finder, etc. These widgets could be developed by software-engineers (develop the widget GUI, access web services, etc.), knowledge specialists (write SPARQL queries) or power users (use query builder as in described in chapter 4.3) who know the ontology and then added as additional elements. Figure 8 shows two exemplary widgets. The *similar case finder* widget provides a selection of similar cases. It is conceivable to refine the similar case finder for example with a rating. The other exemplary widget *resource finder* shows possible relevant files, experts or web pages related to the topics of the actual case. It is even imaginable, that these widgets could be exchanged over a marketplace or a widget gallery. Such a flexible approach would offer the possibility to have a great range of functionalities.

Figure 9 shows schematically the possible architecture of a system which supports the knowledge intensive work with semantic technologies.

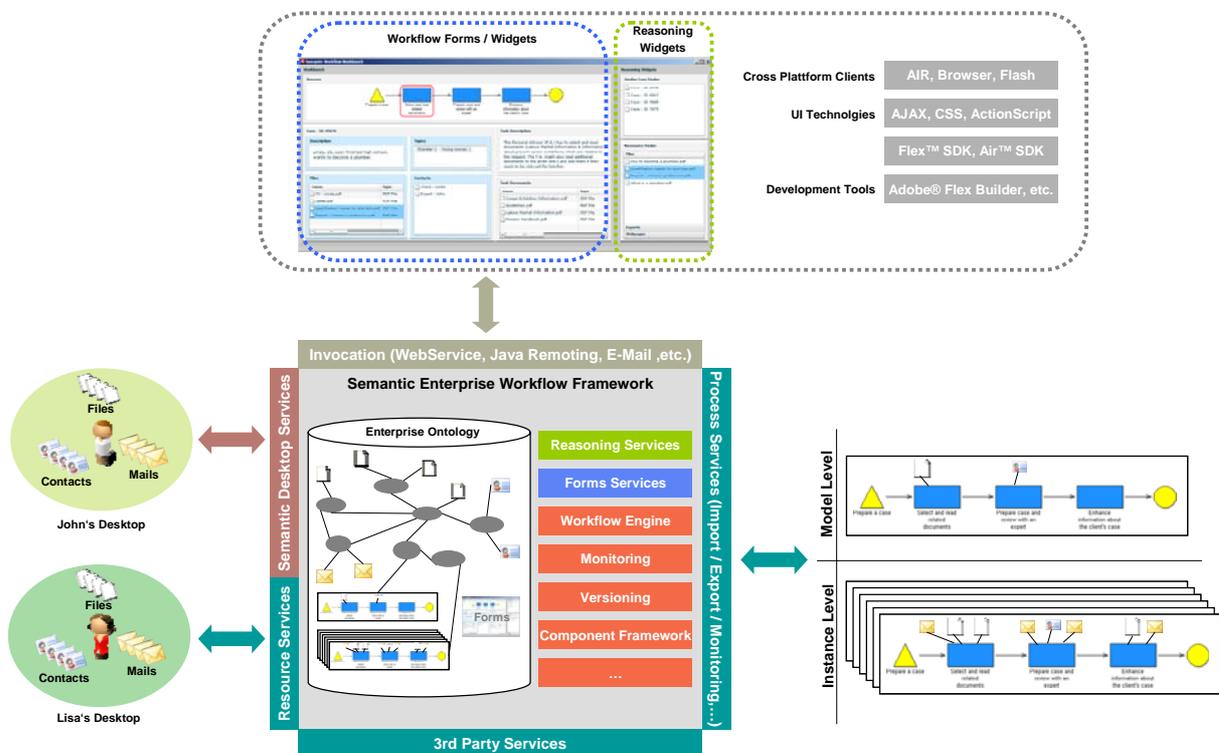


Figure 9: Possible architecture of enterprise ontology and semantic workflow

7 CONCLUSION

The approach shows how processes can be integrated into a semantic desktop framework. This means to integrate processes into the overall context of the organisation but also into the personal environment of a person. It allows the direct and easy relation of processes and its activities with any needed artefact as documents emails, contacts, bookmarks, etc. to perform knowledge intensive work.

Additionally, through the semantic description of processes it is possible to apply reasoning in a new way and therefore enhance the support of workers performing a task by offering needed information in an easy and efficient way, especially for non-automated and knowledge-intensive work.

Further on we proposed how already existing ontologies can be used and extended for the use of process and knowledge intensive work. Finally the paper described how the architecture and interface of a future Semantic Workflow Workbench could look like.

References

- (BPEL, 2003) BEA, Microsoft, IBM, SAP and Siebel Systems. Business Process Execution Language for Web Services. Retrieved on <http://www.ibm.com/developerworks/library/specification/ws-bpel/>, 23rd of June, 2009.
- (BPEL4People, 2007) Active Endpoints, Adobe, BEA, IBM, Oracle and SAP. WS-BPEL Extension for People (BPEL4People), Version 1.0. Retrieved on <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/30c6f5b5-ef02-2a10-c8b5-cc1147f4d58c> 23rd of June, 2009.
- (Dimitrov et al. 2007) Dimitrov, M.; Simov, A.; Motchev, V.; Konstantinov, M. WSMO Studio – A Semantic Web Services Modelling Environment for WSMO. Proceeding of ESWC 2007: 749-758.
- (Feldkamp et al., 2007) Feldkamp, D.; Hinkelmann, K.; Thönssen, B. KISS – Knowledge-intensive Service Support for Agile Process Management. In: M. Hepp, K. Hinkelmann, D. Karagiannis, R. Klein, N. Stojanovic (eds.): Semantic Business Process and Product Lifecycle Management. Proceedings of the Workshop SBPM 2007, Innsbruck, April 7, 2007, CEUR Workshop Proceedings, ISSN 1613-0073, online CEUR-WS.org/Vol-251/.
- (Geisler et al., 1998) Geisler, R.; Klar, M.; Pons, C. Dimensions and Dichotomy in Metamodeling. Technical Report 98-5, Technical University Berlin.
- (Grebner, 2008) Grebner, O. and Riss, U. Implicit Metadata Generation on the Semantic Desktop Using Task Management as Example. In: Borgo, S., Lesmo, L. (eds.) Formal Ontologies Meet Industry. Frontiers in Artificial Intelligence and Applications, Vol. 174 IOS Press, 33-44.
- (Hepp et al., 2005) 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 IEEE ICEBE 2005, October 18-20, Beijing, China, pp. 535-540
- (Hinkelmann et al., 2007) Hinkelmann, K.; Nikles, S.; Thönssen, B.; von Arx, L. An ontology-based modelling tool for knowledge intensive e-government services. Corradini, F., Polzonetti, A. (Eds.): MeTTeG07, pages 43-56.
- (NEPOMUK, 2008) NEPOMUK Project Summary: Networked Environment for Personal Ontology based. Management of Unified Knowledge. Retrieved on <http://NEPOMUK.semanticdesktop.org/xwiki/bin/view/Main1/Project+Summary>, 20th of May, 2009.
- (OMG, 2008) Object Management Group. Specification of the Business Process Modeling Notation (BPMN). Version 1.2. Retrieved on <http://www.omg.org/docs/formal/09-01-03.pdf>, 3rd of July, 2009.
- (Schmidt, 2005) Schmidt, A.. Knowledge Maturing and the Continuity of Context as a Unifying Concept for Knowledge Management and E-Learning. Proceedings of I-Know 2005, Special Track on Integrating Working and Learning. Graz, June/August 2005.

- (Sauermann, 2003) Sauermann L.. The Gnowsis, Using Semantic Web Technologies to build a Semantic Desktop. Diplomarbeit, Technische Universität Wien.
- (Sauermann et. al, 2005) Sauermann, L., Bernardi, A., & Dengel, A. (2005). Overview and outlook on the semantic desktop. ... 1st Workshop on The Semantic Desktop at the ISWC
- (Tudor et al., 2007) Groza T.; Handschuh S.; Moeller K.; Grimnes G.; Sauermann L.; Minack E.; Mesnage C.; Jazayeri M.; Reif G.; Gudjonsdottir R. The NEPOMUK Project - On the Way to the Social Semantic Desktop.
- (SUPER, 2009) SUPER Project: Semantics Utilised for Process Management within and between Enterprises. Retrieved on <http://www.ip-super.org/>, 20th of May, 2009.
- (WS-HumanTask, 2007) Active Endpoints, Adobe, BEA, IBM, Oracle and SAP. Web Services Human Task (WS-HumanTask), Version 1.0. Retrieved on <https://www.sdn.sap.com/irj/sdn/go/portal/prtroot/docs/library/uuid/a0c9ce4c-ee02-2a10-4b96-cb205464aa02>, 23rd of June, 2009.