ONTOLOGY-BASED KNOWLEDGE MANAGEMENT
Graphical Query Editor for OWL Ontologies

Markus Schwinn, Norbert Kuhn and Stefan Richter
Institut für Softwaresysteme in Wirtschaft, Umwelt und Verwaltung, Fachhochschule Trier
Campusallee, 55761 Birkenfeld, Germany

Keywords: Knowledge management, Ontology, Visual query system, Ambient user support, Graphical query language.

Abstract: The OnToBau research project aims to provide a way to classify, archive and effectively use business knowledge with the assistance of an ontology-based knowledge archive for small and medium companies from construction industry. This archive is intended to pro-actively provide users with information to assist them in their daily business process handling. The system consists of four main parts. The *document converters* prepare the different resources (EMails, Paperdocuments, PDFs etc.) that should be stored in the knowledge archive for the enclosed inference system. The *inference system* is the core component and extracts the information from the preprocessed resources. *Ontologies* provide the necessary domain knowledge. In order to exploit the available knowledge, a *personal agent* monitors the current activities of the user and tries to infer the intention from his behaviors. At certain points it automatically offers the user helpful information. Again ontologies are used to represent information about the business processes. In addition, the user has the option to search for information in the archive through the *graphical user interface*. The importance of simple query systems has already been identified in the area of database systems. This paper gives an overview of the OnToBau research project presenting a first approach to visual query for information in the knowledge archive.

1 INTRODUCTION

The growing importance of computers in the 70s and the associated opportunity to disseminate information in digital form, is considered as the turning point to today's information age. As with any other technological revolution in history, the new information technologies spread within two decades all over the planet (Castell, 2001). For the first time, it was possible to produce, to copy and to archive information in a simple way.

In the early 90s another technological milestone was passed, that moved the world to a new era of globalization (Böder, 2003). The Internet offered the opportunity to access and produce information on an increasing number of websites.

The largest flow of information in the Internet is produced by e-mails. A study of the Radicati Group in 2009 stated that in the same year about 247 billion e-mails per day would be sent, and that this number would be doubled by the year 2013. However, approximately 81% of those e-mails can be considered spam (Lyman and Varian, 2003)(Radicati and Khmartseva, 2009). This information overload will be a challenge for many companies in coming years. This is also covered by the same study: it is estimated that companies with 1,000 or more employees have to invest 1.8 million dollars per year in the processing of e-mails and spam (Radicati and Khmartseva, 2009).

However, not only the flood of information from the growing e-mail traffic will challenge the companies in the near future. A study of the Gartner Inc. in 2002 amongst around 300 companies showed that 96% suffer an information overload. To counteract the information overload, companies will have to spend 30 billion US-Dollars within the next years (Goasduff, 2002). A similar investigation of the Basex Inc. in 2008 stated, that the consequences of information overload produce costs of about 900 billion US-Dollars a year for the economy of the United States, caused by reduced productivity of the employees in knowledge intensive processes, which have to spent about 25% of their daily work with searching for information (Spira, 2008).

Interviews with our project partners from the construction industry provided similar results. In partic-

Schwinn M., Kuhn N. and Richter S..
ONTOLOGY-BASED KNOWLEDGE MANAGEMENT - Graphical Query Editor for OWL Ontologies.
DOI: 10.5220/0003491202350240
In *Proceedings of the 13th International Conference on Enterprise Information Systems* (ICEIS-2011), pages 235-240
ISBN: 978-989-8425-55-3

ular, the quotation preparation process has been described as a knowledge-intensive process. The reason is that different resources are used to produce a quotation. Basically, a product database is used. If the product is not in the database, the employee searches in product catalogs, previous invoices or web pages. In addition, business documents are archived in paper form in many small companies, as document management systems are often not tailored to their needs. So searching for information is often a very time consuming task (Schwinn, 2010).

In this paper we describe an architecture, which should pro-actively provide these small companies with relevant information stored in their knowledge archive to assist them in their knowledge intensive business processes. Because of the enormous variety of different business processes our system focuses on processes in the construction domain (e.g., the quotation preparation process). The knowledge base is built up from different resources a company has to deal with (e.g. invoices, product catalogs etc.). To offer the end-user an easy and effective search for information in the knowledge archive, we present an approach of a visual query editor.

The remainder of the paper is organized as follows. First we give an overview of related work in pro-active knowledge management and visual query editors in Section 2. The overall system architecture is described in Section 3, with an overview of the main components. In particular we discuss the visual query editor in Section 4. Finally, we end up with the conclusion and outline some future work.

2 RELATED WORK

In this section we will give an overview of similar research projects considered in process-oriented knowledge management and we present previous work in the area of graphical query construction, particularly in the context of ontologies.

2.1 Process-oriented Knowledge Management

There seems to be no other projects that specifically consider a knowledge-based process support in construction industry. However, the project DYONIPOS, which has a strong similarity to the aims of the OnToBau research project as described above, tries to optimize processes in public administration facilities by providing pro-actively the available knowledge to the employees (Makolm et al., 2007). DYONIPOS has adopted a strict process-oriented approach that moves the focus to the business processes (Tochtermann et al., 2006). This approach is appropriate in an environment with highly structured processes, like in public administration. Otherwise, in an environment like the construction industry most of the processes are semi-structured. In our approach we focus on the documents and the knowledge contained within them. Therefore, providing knowledge is more likely to be tied to the user's behavior than to rigid processes (Schwinn, 2010).

2.2 Visual Query Systems

Most approaches to support the end-user with the query formulation have focused on visual techniques to hide the target query language like SQL for databases or SPARQL in the context of ontologies. (Catarci, 1997) present a classification scheme of 4 different graphical query construction categories of visual query systems (VQS). The tool described in this paper belongs to the category of diagram-based systems, that tend to be the most popular. There have been a few previous approaches to support a visual query construction particularly for ontologies. Some examples include SPARQLViz (Borsje and Embregts, 2006) and NITELIGHT (Russell and Smart, 2008).

SPARQLViz (Borsje and Embregts, 2006) aims to support the user to query constructions for SPARQL. The main difference to our approach is the interaction with the user interface. SPARQLViz relies on a form-based VQS with a wizard-like interface design, guiding the user through different forms. In contrast, we present a diagram-based system. There seem to be no empirical studies on the different VQS categories, so it is difficult to compare this different approaches.

NITELIGHT (Russell and Smart, 2008) is a VQS that has much in common with the VQS presented in this paper and influenced our research to some degree. NITELIGHT supports the user with respect to the specification of all SPARQL query result forms (like SELECT, CONSTRUCT etc.). NITELIGHT offers the possibilities of result ordering, filtering and limiting the results. NITELIGHT is a diagram-based VQS that offers ontology browsing and drag-and-drop functionality with a graph-based visualization. Despite these similarties the following differences do exist between NITELIGHT and our approach. First, the VQL presented in this paper is richer compared to the VQL supported by NITELIGHT. The VQL presented in this paper offers further possibilities on property restrictions like range and cardinality restrictions (e.g. a person with only invoices before 2010). Interviews with our project partners from the construction industry revealed that they often search for information where partial statements are already known (e.g. they search invoices with a particular bathtube or a tender preparation from a specific person). Therefore our VQL supports the query construction including individual statements. The ontology browser in NITELIGHT consists of a series of columns that display the classes and subclasses of the ontology (Russell and Smart, 2008). Whereas our ontology browser provides access to the individuals too. The current approach is diagram-based, but we are planning to develop a icon-based system similar to the result view presented in figure 3. In addition, we use a live result view to give the user a direct feedback on his query construction, so he can remark early when his query goes in the wrong direction.

3 OVERVIEW OF THE ONTOBAU-ARCHITECTURE

There has been much literature about knowledge management systems (KMS) within large enterprises and little information available on KMS within SMEs (Rasheed, 2005). According to (Rasheed, 2005) SMEs have special requirements on KMS. Interviews with our project partners led to the same result. The managers in SMEs are in most cases the owners. The result is, that the decision-making process is shorter than in larger companies. They show a flat and less complex structure, with fewer layers of management (Wong and Aspinwall, 2004). Processes are often not as strongly structured as in larger enterprises and knowledge is distributed at various points in the company (file folder, product catalogs, databases)(Schwinn, 2010). A smaller number of people within a company is usually united by common beliefs and values, resulting in shorter and often less strategic ways for making decisions (Rasheed, 2005). Because of fewer human and financial resources, the introduction of a knowledge management system should not cause ongoing costs. Especially in SMEs there are no specialists for knowledge management and additional staff costs are not manageable. The goal of our project is to account for these special requirements. In summary, we need to consider the following requirements:

- the effort to record analogue documents has to be minimal and should not disrupt the daily work
- intuitive usability
- there should be no running costs, in particular no additional staff cost
- knowledge should be extracted from different resources
- access to the knowledge archive should be possible at any time
- the extracted knowledge should be linked in an effective manner to the usual business processes

During the daily work, the employee can decide whether certain resources should be transferred into the knowledge base. Those resources (e.g. an e-mail or a PDF document) are then passed to the OnToBau-System using the interface of his personal agent or plug-ins integrated in his office or e-mail software. To transfer analogue documents to the knowledge base, we are planning to install special document cameras directly at the workplace of every employee. In this way we prevent the scanning process from disturbing the workflows in the company and ensure that every employee can easily add new resources to the knowledge base. The purpose of this section is to give an overview of the architecture of our approach. As shown in Figure 1, the OnToBau-system consists of four main parts described in the following subsections.

3.1 Document Converters

These pre-processing components will prepare the resources for inclusion in the knowledge base. Therefore the resources are converted to a general representation language, so called OnToBau Representation Language (ORL). The pre-processing also includes various filters (e.g. segmentation filter, part-of-speech tagger, stop word filter), which are used to simplify the subsequent processing. The relevant information is extracted in the inference system component. Interesting e-mails can be convertet into the ORL directly out of the mail client like Thundebird or Mircosoft Outlook.

3.2 Inference System

First, a given resource is classified into process related categories (e.g. an invoice). The domain ontologies contain information which relevant data have to be extracted from this categorie. Once the information has been extracted, it is stored in the knowledge base. For this purpose, we try to find relevant relationships between the new data and the existing knowledge in the knowledge base (e.g. a link between an invoice and a corresponding quotation). If the resource is an e-mail with an attachment, it will be removed from the email and both are integrated into the knowledge archive.

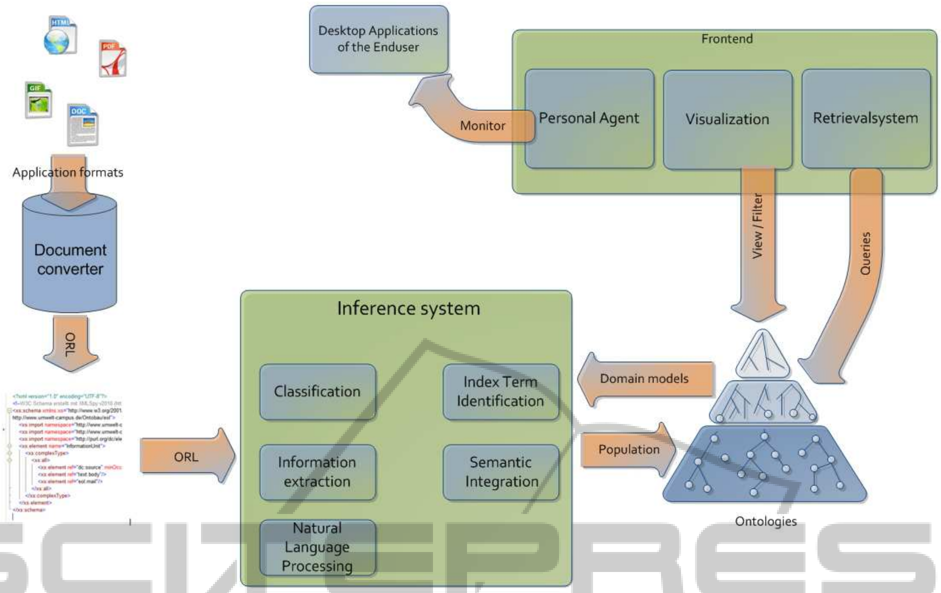

Figure 1: Overview of the OnToBau architecture.

3.3 Ontologies

The knowledge in the OnToBau-system is represented by using ontologies. For this, we decided to use the Web Ontology Language (OWL), which is an W3C recommendation since February 2004, so we can benefit from existing libraries and tools. The T-Box of our ontology defines the most relevant concepts and relations of the construction domain and in general of business documents, thus providing the OnToBau-system with the necessary knowledge to decide which information to extract from the resources (e.g. recipient details, product information etc.). The extracted information is stored as individuals in the ontology.

3.4 Personal Agent and User Interfaces

The personal agent of the employee performs two main tasks. First, he should allow the employee to access the knowledge base by making specific search queries (e.g. show me all in-voices of Mr. Brown with an invoice date later than December 2009). For this purpose we implement a graphical tool for query construction described in the following chapter. Second, the agent should monitor the employees behavior and provide him pro-actively with information to fulfill his task.

4 GRAPHICAL INTERFACE FOR SEMANTIC QUERY CONSTRUCTION

According to (Spira, 2008) employees in knowledge intensive processes have to spent about 25% of their daily work with searching for information. Particularly in small companies knowledge management systems are hardly be found due to the reasons mentionend in section 3. To use the corporate knowledge in the OnToBau-system actively the employees need an intuitive search facility. Since the knowledge is represented using OWL ontologies the queries for information are restricted to a specialized query language for ontologies.

Using the example of searching an invoice with some parameters we will describe our approach: the knowledge worker is searching for an invoice that was sent to a recipient who is known as 'Markus Schwinn' and the invoice date is before january, 1st 2011. He additionally wants only the invoices that contain ordered products of a bathtube or a wash-bowl. In pseudo code the query string for receiving this information from an ontology could look like this:

```
SELECT ?invoice WHERE {
 ?invoice a (invoice and
  (hasInvoiceRecipient value
   MarkusSchwinn) and
   invDate some ?date and
   (hasProduct some
```

```
    (bathtub or wash−bowl))).
  filter(dateTime(?date) < dateTime
  ("2011−01−01T00:00:00Z")).
}
```

This seems very familiar to a technically experienced person. But you can not expect that the end-user is familiar with it too. Visual query systems try to support an easier way for constructing such a query string. Like (Russell and Smart, 2008) we decided to use SPARQL as the target query language for our approach. The user can select the entities, data and object properties of the ontology from a treeview. Contrary to NITELIGHT we decided to use individuals for query constructions too. This reflects the fact that the knowledge worker often knows some details exactly. In our example he knows the individual of the invoice recipient. Possibly this will lead to a more precise and unique query result.

The knowledge worker could drag-and-drop the objects onto the query editor frame where they are presented as graphical nodes. They can be dropped onto the coresponding nodes and are connected automatically with this node by an edge. The type of the edges depends on the node types and reflects the query structure of SPARQL. When needed you can edit the value of a node to specify cardinalities or ranges (e.g. when you are searching for a person with at least three invoices). Figure 2 shows the same query above but much more intuitive for the knowledge worker.

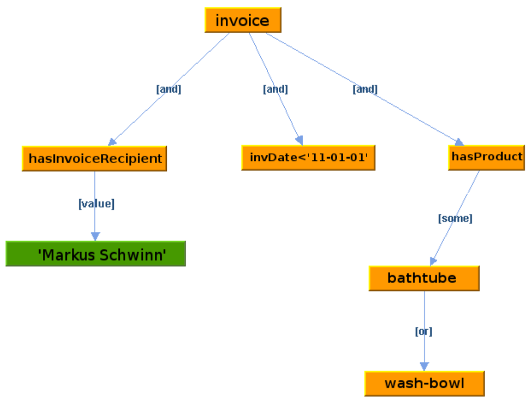

Figure 2: Example of graphical query structure.

After constructing the graphical query with all the information he knows, ideally, the knowledge worker get back one result or a list with all results matching the query. Choosing an item will present the selected individual with all its connections in a well-arranged graph. The implementation of this resulting graph is based on the Protege-Plugin Ontograf, which is slightly enhanced to improve the user experience (see figure 3). In our example this is the requested invoice with its connection to the individual 'Markus Schwinn'. Also two products from this invoice are noticable, a bathtube and a wash-bowl, represented by an icon and their name. The result graph shows additional connections, like the mail which contained the invoice, the corresponding proposal and the company sending it.

It is planned but not yet implemented to use this icon-based approach also for the query editor. In changing the diagram-based approach to a real graphical approach we undertake a better acceptance on the user-side because of the re-improved usability.

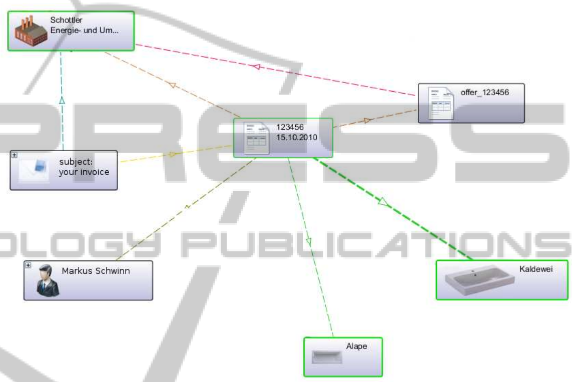

Figure 3: View of the query result.

A feature called live search is currently under development. With every change on the graphical query frame, like adding or removing nodes or changing the type of the edges, the knowledge worker should get an updated list with results matching the current query term. Thus he get an instant reply if the query makes sense or not. With this feature the workflow will be more fluently.

5 CONCLUSIONS AND FUTURE WORK

Because there are no specific projects supporting the knowledge management process of companies in the targeted domain the presented architecture is our way to provide small and medium companies with an ontology-based knowledge management system. To reach this goal we have developed the ORL. This representation language is the link between different resources used within companies and the information extraction. With specific converters the system is able to transform the resources to a unitary format which is needed for extraction. Currently we are working mainly on the extraction step, to enable the system to build up a knowledge base for a company. This base will be represented by using ontologies.

An integral part of the OnToBau system which is not yet im-plemented will be the personal user agent. The agent will moni-tor the users activities and provide him with the relevant infor-mation for the process in real-time. To achieve this, it needs underlying behavior patterns and must try to anticipate the users intention.

The editor for the visual query construction is still under development. Currently it has a diagram-based approach, but it will change to an icon-based visual style. This might improve the user experience with our tool because there will be no ”visual gap” between query construction and result graph. We have not performed any user evaluations with our project partners, but aim to undertake such studies in the near future. Another extension is the live search functionality. When finished it will provide the knowledge worker with a direct response on what he does in the query editor. Currently the query editor and the presentation of the result is divided in two program parts. For a fluently workflow it will be neccessary to combine this in one part.

Our research project focuses on the needs of construction industry but of course this approach could be adapted in other domains dealing with ontologies.

ACKNOWLEDGEMENTS

This work is supported by the German Ministry of Research and Technology under grant FKZ 1731X09.

REFERENCES

Böder, J. (2003). Das Internet und seine Folgen - Was Luthers Reformation und unsere Zukunft gemeinsam haben. http://www.intelligenzia.de/Texte/Internet.pdf. 17. August 2009.

Borsje, J. and Embregts, H. (2006). Graphical Query Composition and Natural Language Processing in an RDF Visualization Interface. Bachelor, Erasmus University Rotterdam.

Castell, M. (2001). *Das Informationszeitalter. Wirtschaft, Gesellschaft, Kultur*, volume 1. Leske + Budrich Verlag.

Catarci, T. (1997). Visual Query Systems for Databases: A Survey. *Journal of Visual Languages & Computing*, 8(2):215–260.

Goasduff, L. (2002). Gartner Says 90 Percent of Businesses Suffer from Information Overload. http://www.gartner.com/5_about/press_releases/2002_-05/pr20020507c.jsp. 24. August 2009.

Lyman and Varian (2003). How Much Information? http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/printable_report.pdf. 19. August 2009.

Makolm, J., Weiß, S., and Reisinger, D. (2007). Proactive knowledge management: the DYONIPOS research and use-case project. In Janowski, T. and Pardo, T. A., editors, *Proceedings of the 1st international conference on Theory and practice of electronic governance (ICEGOV '07)*, pages 85–88, New York, NY, USA. ACM.

Radicati, S. and Khmartseva, M. (2009). Email Statistics Report, 2009-2013. http://www.radicati.com/wp/wp-content/uploads/2009/05/email-stats-report-exec-summary.pdf. 19. August 2009.

Rasheed, N. (2005). The Impact Of Knowledge Management On Smes. Technical report, Knowledgeboard.

Russell, A. and Smart, P. R. (2008). NITELIGHT: A Graphical Editor for SPARQL Queries. In *International Semantic Web Conference (Posters & Demos)*.

Schwinn, M. (2010). Automatic Information Extraction from Documents for Setting Up a Knowledge Archive as Basis for an Automated Tender Preparation. Masters thesis, University of Applied Sciences, Trier.

Spira, J. B. (2008). Information Overload: Now $ 900 Billion - What is Your Organizations Exposure? http://www.basexblog.com/2008/12/19/information-overload-now-900-billion-what-is-your-organizations. 10. December 2010.

Tochtermann, K., Reisinger, D., Granitzer, M., and Lindstaedt, S. (2006). Integrating Ad Hoc Processes and Standard Processes in Public Administrations. In *Proceedings of the OCG eGovernment Conference, Linz (Austria)*,.

Wong, K. and Aspinwall, E. (2004). Characterizing knowledge management in the small business environment. *Journal of Knowledge Management*, 8:44–61.