Journal of Universal Computer Science, vol. 8, no. 5 (2002), 506-515
submitted: 4/2/02, accepted: 3/5/02, appeared: 28/5/02 © J.UCS

Ontology-Based Skills Management: Goals, Opportunities and Challenges

Jacqueline R. Reich
(Rentenanstalt/Swiss Life, Switzerland
jacqueline_renee.reich@roche.com)

Peter Brockhausen
(Rentenanstalt/Swiss Life, Switzerland
peter.brockhausen@swisslife.ch)

Thorsten Lau
(Rentenanstalt/Swiss Life, Switzerland
thorsten.lau@gmx.net)

Ulrich Reimer
(Rentenanstalt/Swiss Life, Switzerland
ulrich.reimer@acm.org)

Abstract: Establishing electronically accessible repositories of people's capabilities, experiences, and key knowledge areas is key in setting up Enterprise Knowledge Management. A skills repository can be used for e.g. finding people, staffing, skills gap analysis, and professional development. The ontology based skills management system developed at Swiss Life uses RDF schema for storing ontologies. Its query interface is based on a combined RQL and HTML query engine.
Key Words: skills management, ontologies, RDF
Category: H.3, H.4, K.6.1, K.4.3, I.2.4

1 Introduction

The tacit knowledge, personal competencies, and skills of its employees are the most important resources of a company for solving knowledge-intensive tasks such as decision-making, strategic planning, or creative design. They are the real substance of the company's success [Taubner and Brössler 2000]. Therefore, establishing an electronically accessible repository of people's capabilities, experiences, and key knowledge areas is one of the major building blocks in setting up Enterprise Knowledge Management. Such a skills repository forms the basis for a Skills Management System, which can be used to expose skill gaps and competency levels, to enable the search for people with specific skills, and can influence the requirements for training, education and learning opportunities as part of team building and career planning processes [Ackerman et al. 1999].

By making employees' experiences, knowledge and skills explicit, it is easier to find out what people know or to direct people to others who can be of help. This sharing of information improves the organisational productivity as well as the individual performance. It supports staffing and enables the planning of professional development [Auer 2000, Sure et al. 2000] – or, as Younker phrased it, "Skills management is a robust and systematic approach to forecasting, identifying, classifying, evaluating and analysing the work force skills, competencies and gaps that enterprises face" [Younker 1998].

Implementing a Skills Management system is a threefold effort. One has to address the technical, the content, and the cultural dimension. The *technical dimension* deals with providing the necessary functionality. The *content dimension* encompasses the set up of organisational and automatic processes for keeping the contents up-to-date. The concern of the *cultural dimension* is to ensure a climate of trust and openness so that employees are motivated to make their skills known – to their own and to the company's benefit. Skills Management may offer the means to affect a cultural change and instill real change into the organisational mind-set and value-set [Deiters et al. 2000, Liao et al. 1999].

The outline of the paper is as follows. We start with a description of Swiss Life's architecture of a Skills Management application [Section 2]. Section 3 is devoted to the ontology development process. In section 4 we give a brief description of the querying facilities before we conclude with an outlook on future work [Section 5].

2 Skills Management at Swiss Life

At Swiss Life we developed a Skills Management system (SkiM) that in its first version aims at finding people with a certain skills profile. This can either be used for staffing new projects, or for identifying experts who might help to solve a certain problem. Employees describe their skills themselves. They are totally self-responsible in this. However, as the skills are publicly visible within the company social pressure will work as a corrective, causing employees to be honest in describing their skills.

Furthermore, participation in SkiM is completely voluntary. Instead of making it obligatory we rely on the motivation of employees to become more visible within the company and thus to increase their career opportunities. SkiM can be seen as providing an internal job fair. An employee specifies his or her skills by selecting concepts from a terminology [see Section 3.1] and by indicating a level for each selected skill. Skills levels range over four steps from "elementary knowledge" to "expert". Although the skills are visible to every other employee, the actual skills levels are not, guaranteeing some privacy. However, this is subject to discussion, among others to better enforce the social pressure mentioned above.

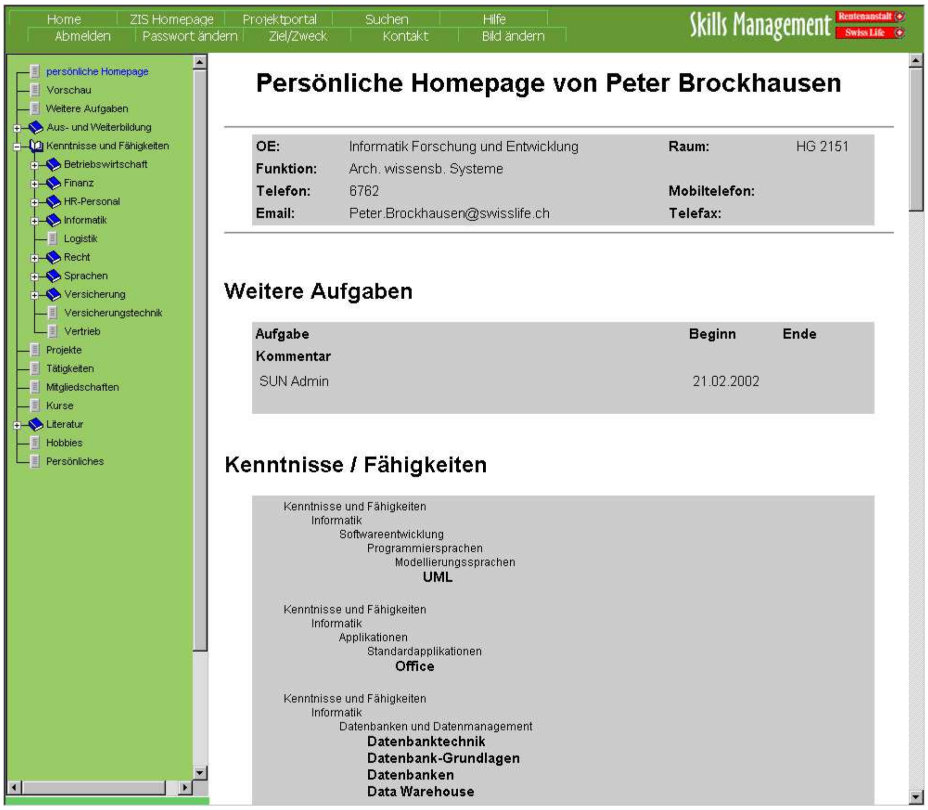

Figure 1: *A personal home page in SkiM (top part)*

Besides the skills more details can be given, like education, former affiliation, special interests, projects participated in, etc. Finally, from all statements given a personal home page is generated on the intranet, which can then be searched [see Fig. 1].

Whenever an employee wants to register with a skill or education type that is not part of the existing catalogue, she or he can easily extend the catalogue by forwarding the suggested new term and its supposed place in the hierarchy to the SkiM administrator who will care for its correct integration. The new term will be visible to all users as soon as the integration will be completed.

2.1 Architecture of the SkiM System

The SkiM system comprises several components [see Fig. 2]. The Ontology Editor OntoEdit allows an administrator to edit the ontologies for skills, education, and

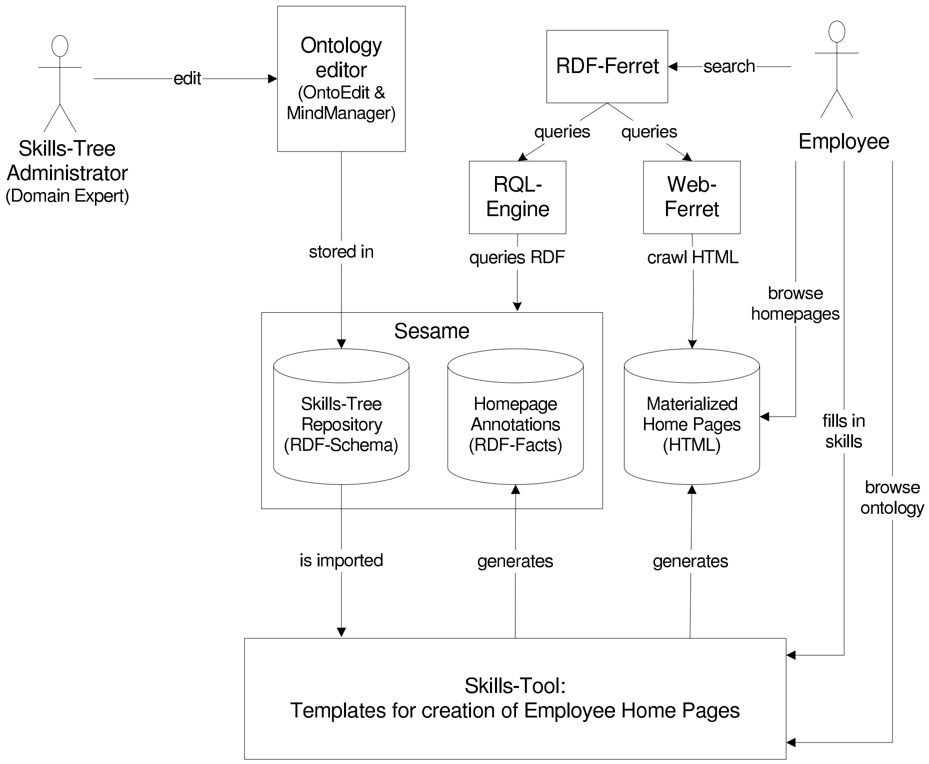

Figure 2: *Architecture of the SkiM System*

job functions. OntoEdit enables inspecting, browsing, codifying and modifying ontologies and therefore supports the ontology development and maintenance tasks. The ontologies are modelled at a conceptual level and independently of the formalism of the final representation language. OntoEdit offers views on conceptual structures, such as concepts, concept hierarchy, relations, or axioms [Sure and Studer 2001]. For the early phases of ontology development the tool MindManager was used to edit the ontologies because it better supports brainstorming processes [see Section 3.2].

The Web Application part of SkiM allows employees to build their personalised intranet home pages by filling in the information categories given by templates.

Sesame from Aidministrator is an RDF/RDF Schema Storage and Retrieval system [Broekstra et al. 2001]. Within SkiM Sesame stores the skills ontology as RDF Schema and the instances of the ontology concepts, namely the association of skills to employees, as RDF facts. It also stores any additional RDF annotations of the home pages which serve to characterise the content of the free text fields. Sesame supports expressive querying of RDF schema and RDF facts by means of a query engine for the RQL query language.

The query interface employs RDFferret [Davies et al. 2002] to do a combined ontology-based and free text retrieval [see Section 4]. RDFferret combines full text searching with querying RDF facts, in our case the skills data for each employee stored in Sesame as well as the additional annotations. Full text searching is provided to offer high recall and coverage of unannotated information, while precise ontological queries result in a high precision. Of course, a combination of both query modes is possible.

3 SkiM as an Ontology-Based Approach

3.1 The Underlying Ontology

Within SkiM, three ontologies are defined - for skills, education, and job function. At the moment, these ontologies are just taxonomies but will be extended to include structured concepts in order to allow a more advanced functionality of SkiM [see Section 5]. SkiM forces every skill, education or job description of employees to be formulated by terms selected from the corresponding ontology. We thus make sure that the terms used for describing skills, education or jobs will match with query terms when SkiM users search for information. This will guarantee a high recall and precision of the result sets. Moreover, the application of ontologies is a prerequisite for comparing skills descriptions, for generating a classification of the organisation's knowledge, and for doing a so-called gap analysis which identifies skills not sufficiently present in the organisation but needed.

The skills ontology consists of three rather independent branches which correspond to the three organisational units that were selected for the pilot phase, i.e. IT, Private Insurance, HR [see Fig. 3, *Informatik, Versicherung, HR-Personal*]. The ontologies for education and job function are not divided into sub-domains as the skills ontology. Currently, the skills ontology consists of 700 concepts, the education ontology consists of 180 concepts, and the job function ontology comprises 130 concepts.

The concept hierarchies are only that part of the underlying ontology which a SkiM user sees. The complete ontology additionally includes concepts and attributes to allow the connection between employees and their descriptions. An OIL fragment that gives an impression of the whole ontology is shown in Fig. 4.

3.2 Ontology building

The development and maintenance of appropriate ontologies are the main challenges in building a Skills Management system. The manual ontology develop-

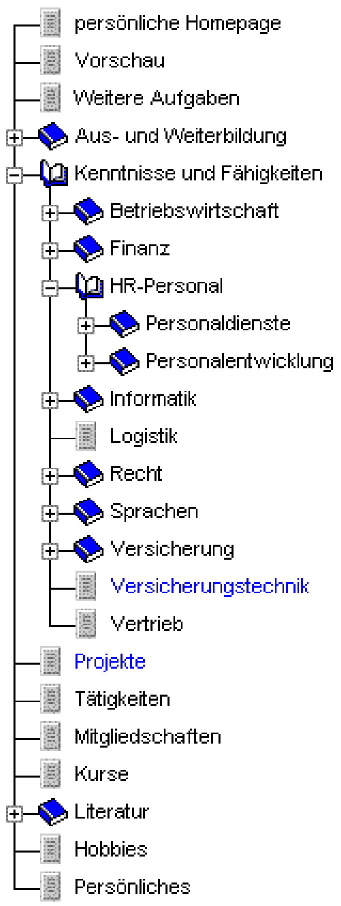

Figure 3: *Top levels of the Swiss Life skills ontology*

ment can be characterised as an iterative, incremental and evaluative process. In the beginning, we provided the domain experts with a simple top level ontology to give a better understanding of the domain to be covered by the ontology. According to an initial baseline methodology inspired by [Sure and Studer 2002] we advised the experts to use simple but helpful design rules, such as reducing the degree of branching by setting a maximum of 5 to 10 branches, or limiting the maximum depth of the ontology. Then, domain experts independently filled their specific domain area within this top level ontology.

Using the design rules resulted in an overall reduction of the concepts which was a welcome side effect. In total, this step resulted in an ontology with more than 1000 concepts, including many duplicates. We then discussed and freezed

```
class-def Skills
    slot-constraint HasSkillsLevel cardinality 1
slot-def HasSkills
    domain Employee
    range Skills
slot-def WorksInProject
    domain Employee
    range Project
    inverse ProjectMembers
slot-def ManagementLevel
    domain Employee
    range one-of "member" "head-of-group" "head-of-dept" "CEO"
class-def Publishing
    subclass-of Skills
class-def DocumentProcessing
    subclass-of Skills
class-def DesktopPublishing
    subclass-of Publishing and DocumentProcessing
instance-of GeorgeMiller Employee
related HasSkills GeorgeMiller DesktopPublishing3
instance-of DesktopPublishing3 DesktopPublishing
related hasSkillsLevel DesktopPublishing3 3
```

Figure 4: *A glimpse of the whole ontology*

the ontology layer by layer, thereby identifying and eliminating some semantic duplicates in the ontology. Moreover, parts of the ontologies were restructured and apparently missing concepts were added.

For the development process we chose the brainstorming and mind mapping tool MindManager from Mindjet. We created concept hierarchies, reorganised them using simple drag and drop mechanisms, and applied the export function to make the ontology public on the web for review purposes. In addition, we annotated ontology elements with symbols or short notes about decisions that were made. For instance, a question mark denotes an open topic to be discussed, while a tick stands for an approved part of the ontology [see Fig. 5]. For group discussions we made large printouts of the ontologies and put them on the wall. The group of developers could view the current state of the ontologies with the meta data describing the state of the discussion. Then the unclarified points of the ontology were discussed and the ontology was rearranged and completed step by step. This approach to ontology development proved to be very successful concerning the outcome, the time required and the satisfaction of the ontology developers.

The iterative approach as sketched above makes it very difficult to get a clear versioning of the ontologies. Since most of the decisions are an outcome of a discussion, part of the changes never physically exist as a version of their own.

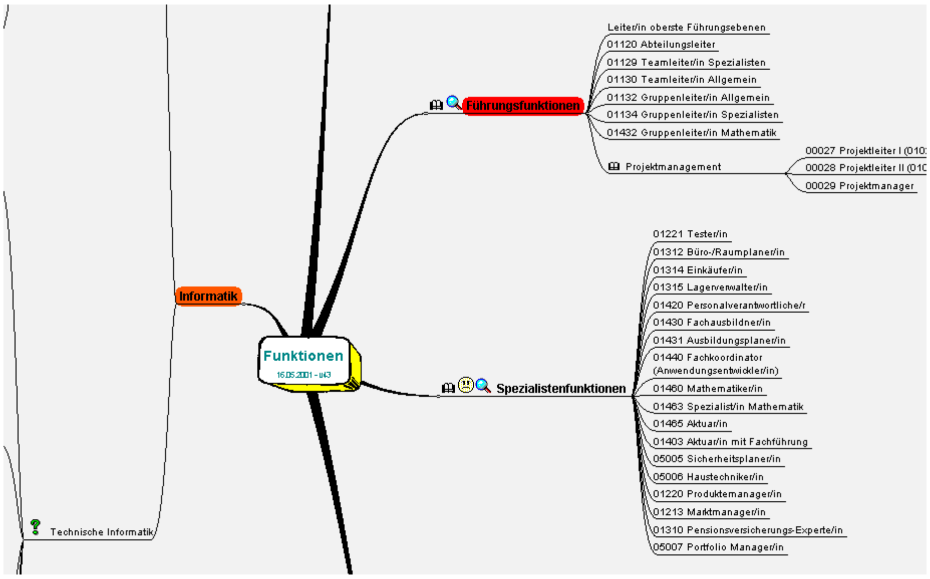

Figure 5: *Meta data for Ontology development (screen shot from MindManager)*

It is also very hard to record the arguments that led to a decision without making a detailed protocol of the discussion. As this is more or less impossible due to the dynamic nature of the discussions, we only documented the result of a discussion and the main arguments for the decisions but left out any intermediate parts of the decision process.

While MindManager is an excellent tool to develop hierarchies in a cooperative brainstorming process, it does not offer real editing functionality. It does not check for duplicates in the ontology, relations can not be restricted in any way (e.g. range, cardinality), nor does it distinguish between the identifier for a concept and its representation. Such a distinction is a prerequisite for the construction and maintenance of multi-lingual ontologies which are a basic requirement in an international company, such as Swiss Life. All these features are supported by the ontology editor OntoEdit [Sure and Studer 2001]. Therefore, a combination of both tools might be close to a perfect ontology development tool: MindManager for the early development phase while using OntoEdit for extensions, maintenance, and versioning.

4 Querying Facilities

Searching for employees with certain skills can be done via their skills only, or can be combined with search terms that aim at the other information categories of a personal home page, like education, special interests, projects worked in, etc. Query terms for skills are enforced to be from the ontology.

To make sure that search terms are only evaluated in the proper information category RDF annotations [Brickley and Guha 2000] are introduced in the home pages so that for each term it is known to which category it belongs. We employ the search engine RDFferret from British Telecom [Krohn and Davies 2001] which is capable of combining an ontology-based search (by interpreting RDF facts) with a free text search. It also allows to confine search terms to certain information categories of a home page by interpreting RDF statements in a web page.

In order to achieve a match between search terms for skills with an employee's skills description, an up- or down-posting along the concept hierarchy is done. The results are ranked according to the skills levels specified and the overall degree of matching between a home page and the query.

5 Evaluation and Outlook

We are currently evaluating the existing version of SkiM in a pilot phase with 150 users. We found them to be very open to such a system and very willing to publish their skills, provided their skills descriptions are publicly visible in the company. Most users said that they would not participate if their skills would only be seen by a few managers and a small group of people in the HR department. This confirms our hypothesis that employees will voluntarily participate in such a system if their personal benefit is a higher visibility in the company.

Many users complained that browsing the skills ontology is too cumbersome. Thus, we will have to look into how to make the ontology better searchable.

We are currently discussing to introduce skills management on the corporate level, i.e. with a visibility across all subsidiaries and branches of Swiss Life. In that case we would need a multi-lingual skills ontology because otherwise many people would feel uncomfortable in using a system with English terms only.

An approach complementary to ours is to identify people with certain skills by doing text mining on the documents in the intranet [Becerra-Fernandez 2000, McDonald and Ackerman 1998]. Adding such text analysis functionality to our system would be ideal for generating suggestions for each employee to extend his or her skills descriptions, and thus to make sure that skills descriptions once delivered stay up-to-date.

Acknowledgements

This work has been partially supported by the European Commission research project On–To–Knowledge (IST-1999-10132), and by the Swiss Federal Office for Education and Science (project number BBW 99.0174). The SkiM system was initiated and brought into existence by our former colleague Bernd Novotny.

References

[Ackerman et al. 1999] Ackerman, M. S., McDonald, D., Lutters, W., and Marumatsu, J. (1999). Recommenders for expertise management. In *Workshop on Recommender Systems: Algorithms and Evaluation at SIGIR '99*.

[Auer 2000] Auer, T. (2000). Wissens-Sicherung im Kontext mit der Personal-Entwicklung. *Schweizer Arbeitgeber*, 10.

[Becerra-Fernandez 2000] Becerra-Fernandez, I. (2000). Facilitating the online search of experts at NASA using Expert Seeker People-Finder. In Reimer, U., editor, *Proc. of the Third Int. Conf. on Practical Aspects of Knowledge Management PAKM 2000*.

[Brickley and Guha 2000] Brickley, D. and Guha, R. (2000). Resource description framework (RDF) schema specification 1.0 W3C candidate recommendation 27 march 2000. http://www.w3.org/TR/2000/CR-rdf-schema-20000327.

[Broekstra et al. 2001] Broekstra, J., Kampman, A., and van Harmelen, F. (2001). Sesame: An architecture for storing and querying RDF data and schema information. In D. Fensel, J. Hendler, H. L. and Wahlster, W., editors, *Semantics for the WWW*. MIT Press.

[Davies et al. 2002] Davies, N., Duke, A., Krohn, U., and Weeks, R. (2002). RDF-Ferret: Search technology for the semantic web. In Davies, N., Fensel, D., and van Harmelen, F., editors, *Ontology-driven Knowledge Management: Exploiting the Semantic Web*. Wiley. to appear.

[Deiters et al. 2000] Deiters, W., Lucas, R., and Weber, T. (2000). Skill-Management: ein Baustein für das Management flexibler Teams. *Information Management & Consulting*, 15(3):54–60.

[Krohn and Davies 2001] Krohn, U. and Davies, J. (2001). The search facility rdfferret. Technical Report Deliverable 11, On–To–Knowledge, EU Project IST–1999–10132.

[Liao et al. 1999] Liao, M., Hinkelmann, K., Abecker, A., and Sintek, M. (1999). A competence knowledge base system for the organisational memory. In Puppe, F., editor, *XPS-99: Knowledge-Based Systems - Survey and Future Directions*, volume 1570 of *LNAI*, pages 125–137. Springer Verlag.

[McDonald and Ackerman 1998] McDonald, D. and Ackerman, M. S. (1998). Just talk to me: A field study of expertise location. In *Proc. of the 1998 ACM Conf. on Computer Supported Cooperative Work (CSCW'98)*.

[Sure et al. 2000] Sure, Y., Maedche, A., and Staab, S. (2000). Leveraging corporate skill knowledge - from ProPer to OntoProper. In Reimer, U., editor, *Proc. of the Third Int. Conf. on Practical Aspects of Knowledge Management PAKM 2000*.

[Sure and Studer 2001] Sure, Y. and Studer, R. (2001). On-to-knowledge ontoedit. Technical Report Deliverable 3, On–To–Knowledge, EU Project IST–1999–10132.

[Sure and Studer 2002] Sure, Y. and Studer, R. (2002). On-to-knowledge methodology. In Davies, N., Fensel, D., and van Harmelen, F., editors, *Ontology-driven Knowledge Management: Exploiting the Semantic Web*. Wiley. to appear.

[Taubner and Brössler 2000] Taubner, D. and Brössler (2000). The people make the project. Project control: The human factor. In *Proceedings of ESCOM-SCOPE 2000*, pages 105–113.

[Younker 1998] Younker, E. (1998). Skills management: The linchpin for IT work force management. *InSide GartnerGroup*, pages 2–4.