LNAI 4333

Ulrich Reimer
Dimitris Karagiannis (Eds.)

Practical Aspects of Knowledge Management

6th International Conference, PAKM 2006
Vienna, Austria, November/December 2006
Proceedings

Springer

Lecture Notes in Artificial Intelligence 4333

Edited by J. G. Carbonell and J. Siekmann

Subseries of Lecture Notes in Computer Science

Ulrich Reimer Dimitris Karagiannis (Eds.)

Practical Aspects of Knowledge Management

6th International Conference, PAKM 2006
Vienna, Austria, November 30 - December 1, 2006
Proceedings

Springer

Series Editors

Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Ulrich Reimer
University of Applied Sciences St. Gallen
Institute for Information and Process Management
Teufener Strasse 2, 9000 St. Gallen, Switzerland
E-mail: ulrich.reimer@fhsg.ch

Dimitris Karagiannis
University of Vienna
Faculty of Computer Science, Department of Knowledge and Business Engineering
Bruenner Str. 72, 1210 Vienna, Austria
E-mail: dk@dke.univie.ac.at

Library of Congress Control Number: Applied for

CR Subject Classification (1998): I.2, H.2.8, H.3-5, K.4, J.1

LNCS Sublibrary: SL 7 – Artificial Intelligence

ISSN 0302-9743
ISBN-10 3-540-49998-9 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-49998-5 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 11944935 06/3142 5 4 3 2 1 0

Preface

The biennial PAKM Conference Series offers a communication platform and meeting ground for practitioners and researchers involved in developing and deploying advanced business solutions for the management of knowledge in organizations. PAKM is a forum for people to share their views, exchange ideas, develop new insights, and envision completely new kinds of knowledge management solutions.

PAKM 2006, the Sixth International Conference on Practical Aspects of Knowledge Management, was held again in Vienna. It was a milestone for two reasons: First, it marked an anniversary – 10 years of PAKM conferences. The first conference was held in Basel, Switzerland, in 1996, followed by the conferences in 1998 and 2000, in Basel as well. After that PAKM moved to Vienna where it was held in 2002, 2004, and 2006.

Secondly, from now on PAKM will be "on tour": It will be organized by different people and be hosted at different places all over the world. The PAKM Steering Committee will be responsible for selecting the conference chairs and the conference locations. The Steering Committee will also be responsible for the direction the PAKM conferences will take and will ensure their continuing high quality.

For this year's conference we received 123 submissions from 30 countries. Based on the reviews by the members of the Program Committee and the additional reviewers, 29 papers were selected. They cover a great variety of approaches to knowledge management, which tackle the topic from many different angles. It is this very diversity that makes PAKM unique, while at the same time focussing on the one issue of managing knowledge within organizations.

Many people were involved in setting up PAKM 2006. We would like to express our warm thanks to everybody who contributed to making it a success. First of all, this includes all the authors who submitted a paper to the review process, the members of the Program Committee and the additional reviewers who made such an effort to select the best papers and to ensure a high-quality program. Our thanks also go to Xiulian Benesch, who was responsible for all the organizational work and the University of Vienna for providing an excellent environment for the conference.

November 2006

Ulrich Reimer
Dimitris Karagiannis

Organization

Program Chairs

Ulrich Reimer, University of Applied Sciences, St. Gallen, Switzerland
Dimitris Karagiannis, University of Vienna, Austria

Program Committee

Irma Becerra-Fernandez, Florida International University, USA
Xavier Boucher, Ecole des Mines de St. Etienne, France
John Davies, British Telecom, UK
Kemal A. Delic, Hewlett-Packard, France
Juan Manuel Dodero, University Carlos III Madrid, Spain
Johann Eder, University of Vienna, Austria
Joaquim Filipe, Escola Superior de Tecnologia Setubal, Portugal
Thomas Fischer, DITF Denkendorf, Germany
Naoki Fukuta, Shizuoka University, Japan
Aldo Gangemi, ISTC, Italy
Ulrich Geske, Fraunhofer Gesellschaft FIRST, Germany
Enrico Giunchiglia, Universita di Genova, Italy
Norbert Gronau, University of Potsdam, Germany
Knut Hinkelmann, University of Applied Sciences Nordwestschweiz, Switzerland
Hans Hinterhuber, University of Innsbruck, Austria
Achim Hoffmann, University of New South Wales, Australia
Byeong Ho Kang, University of Tasmania, Australia
Noriaki Izumi, AIST, Japan
Manfred Jeusfeld, University of Tilburg, Netherlands
Iluju Kiringa, University of Ottawa, Canada
Niklaus Klaentschi, EMPA, Switzerland
Edith Maier, University of Applied Sciences St. Gallen, Switzerland
Vladimir Marik, Czech Technical University, Czech Republic
Frank Maurer, University of Calgary, Canada
Hermann Maurer, Technical University of Graz, Austria
Heinrich Mayr, University of Klagenfurt, Austria
Michele Missikoff, Italian National Research Council, Italy
Katharina Morik, University of Dortmund, Germany
Nicos Mylonopoulos, ALBA, Greece
Alun Preece, University of Aberdeen, UK
Peter Reimann, University of Sydney, Australia
Debbie Richards, Macquarie University, Australia
Bodo Rieger, University of Osnabrueck, Germany

Marcin Sikorski, Gdansk University of Technology, Poland
Marcus Spies, Munich University, Germany
Steffen Staab, University of Koblenz-Landau, Germany
Rudi Studer, University of Karlsruhe, Germany
Ulrich Thiel, Fraunhofer Gesellschaft, Germany
A Min Tjoa, Technical University of Vienna, Austria
Klaus Tochtermann, I-Know Center Graz, Austria
Robert Trappl, Medical University of Vienna, Austria
Eric Tsui, Hong Kong Polytechnic University, China
Roland Wagner, Johannes Kepler University Linz, Austria
Fritjof Weber, EADS , Germany
Mary-Anne Williams, University of Technology Sidney, Australia
Takahira Yamaguchi, Keio University, Japan

Additional Reviewers

Anupriya Ankolekar
Richard Arndt
G. Barchiesi
Martine Collard
Olivier Corby
Klaas Dellschaft
Timm Euler
Daniela Feldkamp
Thomas Franz
Olaf Goerlitz
Arvind Gudi
Hans-Jörg Happel
Benjamin Johnston
Ingo Mierswa
Antonio De Nicola
Simon Nikles
Valentina Presutti
Jose Rocha
Carsten Saathoff
Federica Schiappelli
Sergej Sizov
Max Völkel
Johanna Voelker
Valentin Zacharias

Table of Contents

Web Service Based Business Processes Automation Using Semantic Personal Information Management Systems – The Semantic Life Case

Amin Anjomshoaa, Tho Manh Nguyen, Ferial Shayeganfar, and A Min Tjoa

Institute of software technology and Interactive Systems
Vienna University of Technology
Favoritenstrasse 9-11, 1040 Vienna, Austria
{andjomshoaa, tho, ferial, amin}@ifs.tuwien.ac.at
http://www.ifs.tuwien.ac.at

Abstract. Business today is the crossing point of information which are originated or deducted from different information resources. The process of assembling pieces of functionality into complex business processes very often necessarily involves human interaction which in turn heavily depends on environment and domain-specific Knowledge.

This paper deals with the use of Personal Information Management Systems and Semantic Web technology as enabler of business processes to realize the auto-interaction of customized processes, resources and events. Using an approach to integrate Semantic Filters in the proposed *"business pipelines"*, it is possible to address the most important issues of Process Integration and Process Automation.

The paper presents the SemanticLIFE research project and its applications in the tourism domain for business process automation by providing semantics for business pipelines and localization of process pipelines based on the semantics of Personal Information of potential clients. We will also provide a solution for automatic service orchestration of semantic services as semantic pipelines. Finally, we propose the Semantic Ranking model to evaluate the Semantic Matching in a typical tourism recommendation scenario.

1 Introduction

The fast growth of the World Wide Web and the emerging pervasiveness of digital technologies within our information society have significantly revolutionized business transactions, trade and communication between people and organizations. [3].Besides the augmentation effect, business-related information is characterised by the fact that it also originates from heterogeneous sources and get more and more complex in structure, semantic and communication standard. Therefore, mastering heterogeneity becomes a more and more challenging issue for research in the area of Business Process Management. This challenge involves all facets of process integration, composition, orchestration, and automation amongst heterogeneous systems.

Web services [9], built on top of existing Web protocols and open XML standards, recently emerge as a systematic and extensible framework for application-to-application

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 1 – 12, 2006.

interaction. Web services allow automatic and dynamic interoperability between systems to accomplish business tasks. However, due to the lack of the explicit semantic context, the process of assembling “pieces of functionality” into complex business processes is still unthinkable without significant human involvement.

Semantic Web [4] ,another emerging technology, is being increasingly applied in a large spectrum of applications in order to support diversified knowledge processing by machines. It is a paradigm shift to fulfil the goal of enabling machines to “*interpret*" and *automatically process* the data available on the Web. It has been applied in a variety of application-domains such as knowledge management, e-commerce, healthcare, e-government, data and services integration, searching, and so on.

In SemanticLIFE project [2], we use the Semantic Web technology to build up a long-term ontological Personal Information Management (PIM) system with the aim of creating a semantic repository of all personal data from a variety of sources like emails, contacts, running processes, web browsing history, calendar appointments, chat Sections, and other documents. This PIM system acts as a *digital memory* and provides the “*personal profile*” for acquired persons.

To our understanding, *Web Service* and *Semantic Web* technology are two sides of the same coin which could enable the automation and integration of business processes. Business services are implemented and distributed as web services. With the Semantic information (in our case: personal profile information), it is possible to automatically select customized (web) services, orchestrate them into one complex process and to finally execute the combined process. An example of such orchestration process automation is the Tourist Plan Recommender described in Section 4.

The remainder of the paper is organized as follows. Section 2 briefly reviews the state of the art of Recommender Systems. The SemanticLIFE project with its complete plug-in architecture is described in Section 3. Section 4 presents the integration of SemanticLIFE components with Business Process Execution Language for Web Services (BPEL4WS) standard. The Tourist Plan Recommender and its business pipelines are described in Section 5. In Section 6, we describe the Semantic Ranking method including the Frequency Ranking and Relevance Ranking. Finally, Section 7 concludes the paper with some evaluation discussion and sketch of future work.

2 Recommender Systems: The State of Art

Recommendations systems in the context of travel and tourism became increasingly important since the amount of available information is exploding and users are not always experienced in processing the multitude of information resources. Recommender systems are commonly viewed in the e-commerce domain as applications that are exploited to suggest products and provide consumers with information to facilitate their decision-making processes [13]. They can be classified into the following three types: (1) collaborative-filtering or social-filtering; (2) content-based and (3) knowledge-based [5].

Amazon can be considered as a very popular example of a collaborative-based filter. It collects user ratings on currently proposed products and/or previously purchased items to infer the similarity between users.

In content-based filtering, the user expresses needs and preferences on a set of attributes and the system retrieves the items that match the preferences. New Dudes can serve as an example for a content-based recommendation approach.

The knowledge-based recommender uses knowledge about users and products to build up a recommendation. It integrates both content-based and collaborative-based techniques. Knowledge-based recommender could be based on case based reasoning (CBR) [11]. CBR is a problem solving technology that faces the new problem or situation by first looking back into the past, already solved similar case and use it to solve the present problem. Triplehop's TripMatcher (www. ski-europe.com) and VacationCoach's expert advice platform, Me-Print (used by travelocity.com) are examples of most successful CBR recommendation e-commerce sites [11].

Tourist recommender research efforts are conducted by many universities and organizations due to the necessity of packaging different tourism services for the user (e.g. flight services, car-rentals, hotel accommodation, cultural events etc.). The eCommerce and Tourism Research Laboratory (eCTRL) proposes DieToRecs [10] as a web-based recommendation system that will aid the tourist destination selection process and attempts to accommodate individual preferences. Trip@dvice [7] is another successful eCTRL project, which assists e-travellers in their search for tourism products on the internet. ITR [12]-Intelligence Travel Recommender- integrates case-based reasoning with interactive query management.

L. Ardissono et al. [8] propose Seta2000, an infrastructure for developing personalized recommender systems. They also proposed INTRIGUE [6], a recommendation technique which takes into account the characteristics of the group of participants and addresses possibly conflicting preferences within the group.

So far, most of the existing Recommender systems use traditional methods based on machine learning or case-based reasoning (CBR) techniques to issue the deduced recommendations. Most of them request the user to specify his/her interests and preferences to build up the user profile. Very few of them could ***"remember"*** the user behaviour or interest, even though some click stream analysis systems [14] keep the (short term) history of user's behaviour.

3 SemanticLIFE

SemanticLIFE [2], developed using the Eclipse Rich Client Platform (Eclipse RCP), is designed to store, manage and retrieve the ***lifetime's information entities of individuals using ontologies***. It enables the acquisition and storage of data, giving annotations and retrieval of personal resources such as email messages, browsed web pages, phone calls, images, contacts. The ultimate goal of the project is to build a Personal Information Management system over a Human Lifetime using ontologies as a basis for the representation of its content. . Fig. 1 shows the basic components of SemanticLIFE. The dotted boxes denote the significant plug-ins of the use cases proposed in this paper.

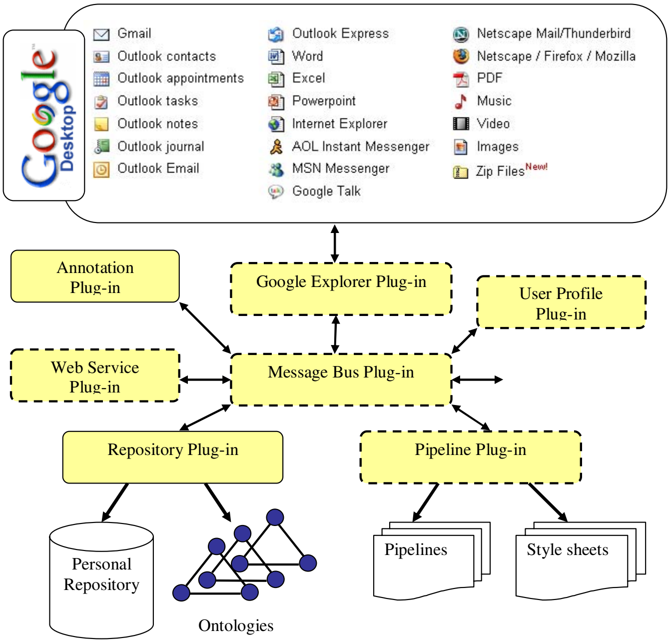

Fig. 1. SemanticLIFE System Architecture

3.1 SemanticLIFE User Profile

A plug-in is developed to import the Google Desktop's captured entities into SemanticLIFE's repository where they are ontologically stored in RDF metadata framework. This will then facilitate semantic meaningful queries, life trails, ranking and processing of life events. The user profile will be partly shaped from this activity logs. User demographics, user interests, contacts could be considered as the static part of the profile. These data will be gradually elaborated and enriched either by automatic or by manual annotation and additions. Fig. 2 shows a fragment of the user model used in the SemanticLIFE system.

This schema helps the system to create a matching behaviour model for user and enhance the user modelling in the following ways:

1. User will have a unique profile that can be reused for many business processes.
2. User model is dynamic and will be adjusted based on the long term user interactions. This approach gains advantages compared with the click-tracking mechanism that captures the user interactions for a limited period on a few web pages.

3.2 SemanticLIFE's Plugins

SemanticLIFE is built upon on several plug-ins components which communicate via the messaging and collaboration component. Message Bus, Web Service and Pipeline plug-ins are the fundamental plug-ins support the communication framework.

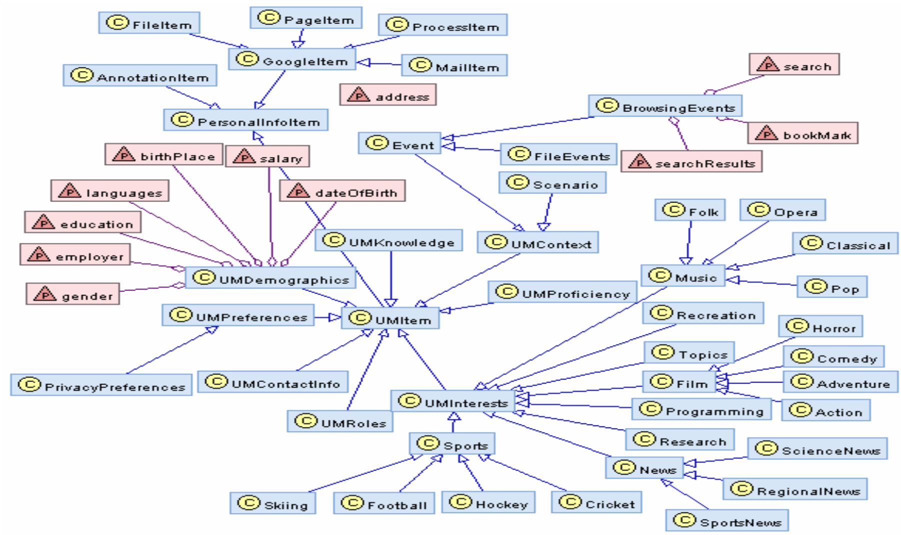

Fig. 2. User Model Ontology

Message Bus plug-in manages all information exchanges between the SemanticLIFE processes. This plug-in supplies also a level of abstraction between system services by providing an transparent, uniform access interface to all services.

With Web Service plug-in (Fig. 3), Web Services can be plugged at anytime to the SemanticLIFE system by locating the corresponding configurations (WSDL file's url). More importantly, the plug-in supports capturing service's semantic which is used in both "*locating appropriate services*" and "*ranking the competitor services*" tasks. The service's semantic, defined in OWL-S standard, [15] describes the functions of the service in terms of the transformation effected by the accordant service. It also specifies required inputs, pre-condition to invoke a service, the generated outputs, and the expected effects that result from the execution of the service.

The Web service plug-in offers the following three categories of services:

- **Service-finder service:** Finds the appropriate services for a specific request. The return-value is a list of services ranked by the user preference or by the semantic ranking of services (based on the user profile and SemanticLIFE items).
- **Service-invocation service:** Invokes the requested service using the SemanticLIFE platform specifications. The invocation mechanism may call an internal pipeline, an internal service or an external web service.
- **Semantic-recommender service:** Invokes the recommendation pipeline of SemanticLIFE for a given service. The pipeline in turn will invoke the underlying service, semantically rank the service-results, provide a ranked list of options for user choices and finally return the selected item.

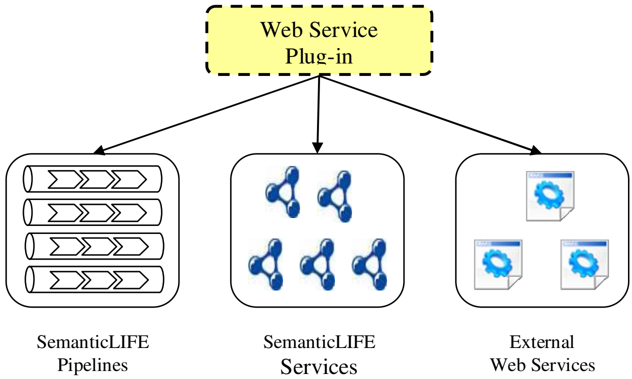

Fig. 3. Web Service plug-in services

Another fundamental plug-in is the pipeline plug-in that plays a central role in the orchestration of basic system services and in the creation of new business services. We introduce the notion of a pipeline as **a uniquely named set of service-calls and intermediate transformations**. The pipelines are defined using an XML structure that specifies *pipeline steps* and *relevant transformations*.

4 SemanticLIFE Business Process Integration

To magnitude the generality of the proposed approach, we integrate the SemanticLIFE components with a standard business process described in Business Process Execution Language for Web Services (BPEL4WS) [16]. The backbone process is a dynamic business process that does a runtime based routing for the required services. The services that are going to be invoked are not known in advance, instead, they will be queried and selected by SemanticLIFE's Web Service Plug-in at run time. The Routing Service will call the "service-finder service" (in a web service plug-in) to find out the service endpoint (a pipeline, a plug-in or external web service) that should be invoked for a specific purpose.

Unlike other recommendation systems which are mostly based on static user profile and case-based recommendation, our system takes into consideration the dynamic, long term user's activities to find out the user interests which may be even change from time to time. The information captured by this plug-in can later on be refined and annotated by the user to make a more precise *user behavioural model*.

The most interesting feature is the "*semantic-recommender service invocation*". The Web Service plug-in will first invoke the specified service, then rank and filter the service results by applying a semantic evaluation and ranking mechanism. This will be performed by considering the *semantic user profile* and the *user information items* captured via Google Desktop plug-in. The user profile and domain ontology will play an important role in the ranking process. For example the list of suggested destinations is filtered on the basis of user interests and also affected by the tourism ontology which specifies the ideal conditions in the requested period of time

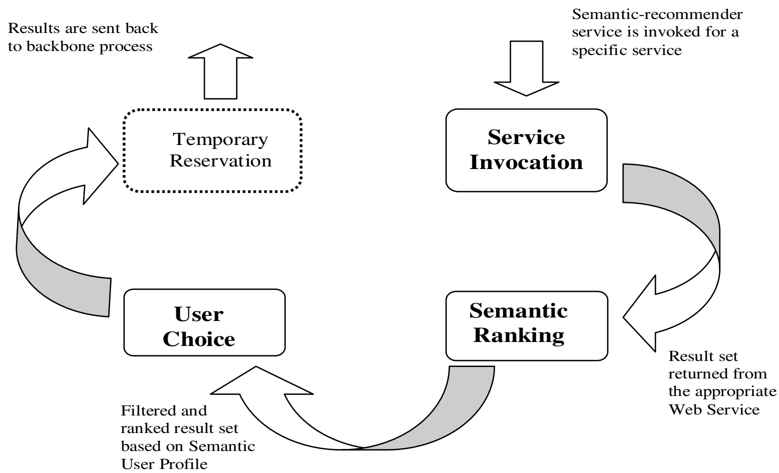

Fig. 4. Semantic-recommender Service Overview

(e.g. weather conditions). The weather information will be treated according to the purpose of a tourist (e.g. different treatment of weather forecasts for different classes of tourists according to a typology such as congress-tourist, family-tourist etc.). The ranked-results will be displayed and the user may select the desired choice(s). An extra step that may take place after the user's choice is to accomplish the reservations or to lock the suggested resources. For example, a hotel-room will be temporarily reserved for the lifetime of the backbone process with a timeout option. This is necessary to make sure that the suggested resources will be available at the final booking step of the backbone process which finalizes the tourist package components.

The user selection will be returned to the backbone process for continuing of business process. It is important to mention that the backbone process should be defined as a process that will need the human user interactions. Such business processes are described in WS-BPEL Extension for People (BPEL4People) specification [17]. Fig. 4 depicts the semantic-recommender phases and the messaging between the phases.

5 The Tourism Scenario as Proof of Concept

To illustrate and evaluate the SemanticLIFE approach we will step by step sketch a business process for a typical problem of a Tourism Plan Recommender. As precondition, we assume that the required web services are available and plugged to Web Service plug-in. Some typical web services for our case are: Country Information Web Servce, City Information Web Service and also Flight and Hotel Reservation Web Services. A business process is initialized by calling the backbone process that respectively calls SemanticLIFE services to perform the relevant business activities and processes. Fig. 5 depicts such a business process that calls relevant SemanticLIFE services like city recommender, hotel recommender, restaurant recommender and so on.

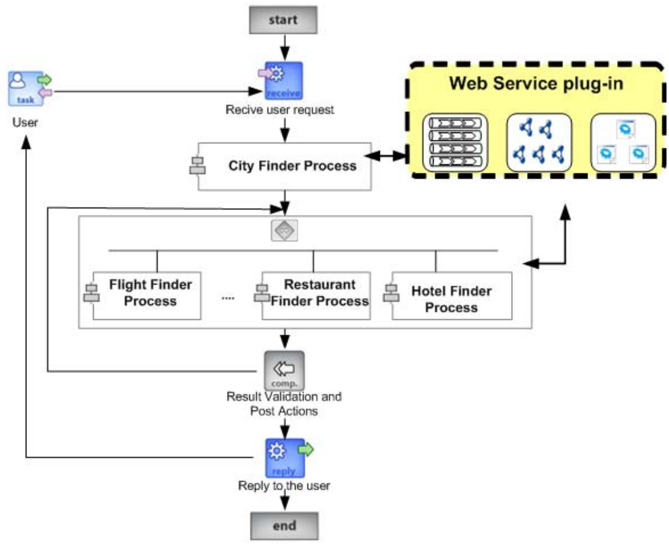

Fig. 5. Backbone process

The ultimate goal of this process is to maximally fulfil the requirements of the tourist vacation such as the proper destination city due to tourist desires, the right hotel, restaurants that match tourist culinary taste, cultural preferences from tourist point of view and necessary needed infrastructures here fore (e.g. barrier free access for people with special needs etc.). Each recommender backbone process is decomposed into two phases namely the *web service discovery phase* and the *service invocation phase* which will be respectively performed via the *service-finder* service and the *semantic-recommender* service of the Web Service plug-in. More precisely, the backbone process will first look for a service that matches the required results and input parameters. Afterward, the backbone process will dynamically set this service's end point and call it through the semantic-recommender service. This call will activate a pipeline that invokes the given call, semantically rank the results and return the user selected choice(s).

The semantic-recommender pipeline may also optionally make some temporary reservations for the potential resources that may possibly vanish due to the concurrent competitors (i.e hotel rooms, flights). The backbone process will then finalize the reservations when the tourist packaging process is successfully completed. In case that for some reason the process is interrupted or cancelled, the resource-locks will be automatically released by lock-timeout option. For the discussed scenario, the backbone process will start with the user request to find the touristy city in a specific country. The following steps are necessary:

Step 1: The Web Service Plug-in will be queried to find an applicable service that can provide the requested information (list of touristy cities in the desired country). The services are ranked and the best matching service will be returned.

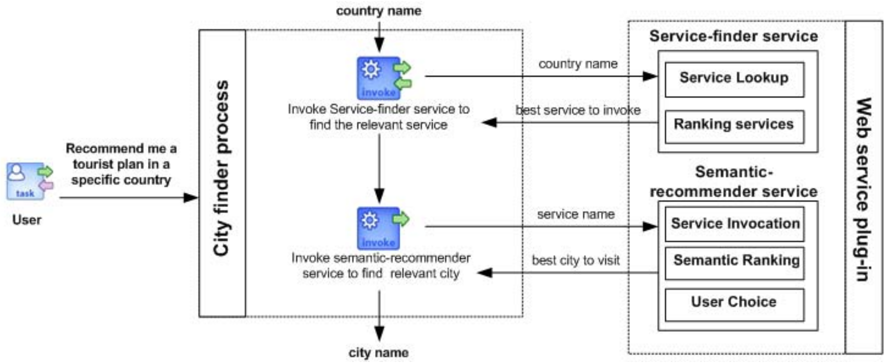

Fig. 6. City finder process

Step 2: The backbone process invokes the service resulted in the previous step via semantic-recommender plug-in. Subsequently, the semantic-recommender plug-in invokes the underline service, semantically ranks service results and presents the ranked results to the user for selection. Finally the user choice will be fed back into the backbone process for further steps. Fig. 6 shows the interaction between backbone process and Web Service plug-in to find a touristy city in a specified country.

The similar scenario occurs in other backbone sub-processes such as the hotel finding sub-process, the concert finding sub-process, the flight finding sub-process etc. The only difference is that the suggested resources may be locked (i.e. hotel, flight) to ensure that it is possible to book them at the final step of backbone process (in case user agree with the suggestion) to complete the tourist package. This extra step may take place as part of the semantic-recommender service.

6 Semantic Ranking

In this section we present the web service ranking technique aiming at finding the most relevant web service to use. The ranking process will follow the "semantic search" phase in which the services' semantics and the corresponding input/output parameters are evaluated to match a specific query. The ranking is performed by measuring the relevance of a service with our domain ontology that is enriched with Google Desktop items. The more related contacts, emails, web pages, etc., the higher rank the service has. The items returned from the appropriate web service will be evaluated and ranked in two different ways: frequency and relevance rank. Both ranking methods benefit from user profile and underneath ontology.

6.1 Frequency Ranking

In this method the number of relevant items and their timestamps will be used to create a ranking measure. This method benefits from the domain ontology by taking the related items into account. For example consider the case that we are looking for information about "Austria". In traditional frequency methods only the items including the term "Austria" could be counted. But using a domain ontology we can find out more about a term, for example we may also add all the items containing any of the following terms too: Vienna, Linz, Salzburg, Austrian and even though "Österreich" (the term Austria in German language). Mathematically speaking our result set will be union of all result sets of related concepts:

Result set **R** = $\bigcup_{c_i} F_{c_i}(C)$ where C_i is related class to our target class with "*same-as*" or "*subclass-of*" predicates and F denotes the frequency of the corresponding term. To apply the notion of time to the result set items we may give a heavier weight to recent items for result ranking which converges to the forgetting mode of memory. For example a city name that has been searched for, one hour ago should have a higher importance than web pages that are browsed one year ago and contain the city name. For this purpose a monotonic decreasing function has to be used to weight the result set (as explained above, including all semantically related concepts) and calculation of the total frequency rank.

6.2 Relevance Ranking

This ranking method is built upon the matches between ontology of the items based on the immediate user requirement and the user profile ontology based on his life-time history. For this purpose we will use a computational model that assesses similarity by adding a semantic measure to feature matching process and is based on ratio model that is introduced by Tversky [1]. To assess the semantic distance between two concepts (classes) A and B, we will first build an extension set for each of classes. This extension set includes all semantic terms that are somehow related to the core concept. For example the extension set of Vienna which is denoted as Ext(Vienna), will include all related features that are related to Vienna. Based on these assumptions the Semantic distance between concepts is defined as follows:

$$D(A,B) = \frac{|Ext(A) \cap Ext(B)|}{|Ext(A) \cap Ext(B)| + \alpha|Ext(A) - Ext(B)| + (1-\alpha)|Ext(B) - Ext(A)|}$$

where α is a real number between 0 and 1. The distance function returns a real measure between zero and one, depends on the similarity of concepts. For a user searching of a destination city scenario, the situation is shown in Fig. 7. The dotted lines between the different concepts demonstrate the semantic relationships between them.

6.3 Combined Ranking

To establish a uniform rank from frequency and relevance ranks, we propose a combined rank that takes into account both of them. Consider that the semantic distance between user's profile and all suggested options is calculated and depicted as $d_1,d_2,\ldots,d_n$ where d_i is the semantic distance between the ith option and user profile. The frequent ranks for system suggested options are $f_1,f_2,\ldots,f_n$ respectively. The combined rank can be defined as follows:

$$\text{Combined Rank for ith option } R_i = \frac{f_i * d_i}{\sum f_i}\ ;\ (0 \le R_i \le 1)$$

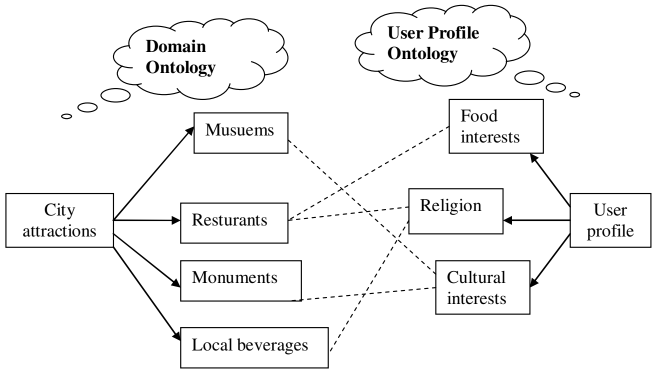

Fig. 7. Ontology matches

Finally the end user will be able to view the recommendation list sorted by combined rank with the option to switch between the two other ranking methods

7 Conclusion and Future Work

The evolution of Semantic Web technology has opened a new window in IT and specially data engineering fields. The proposed scenario showed how this technology can be utilized to make the daily life scenarios easier to organize. Also, the presented SemanticLIFE platform as a personal information manager has the capacity to be used in other business processes dealing with personal information. However before the realization of SemanticLIFE application and scenarios some crucial issues like privacy and security should be addressed. We have also started to run more business scenarios like collaborative environments and providing semantic feeds for enterprise systems [13]. Some other challenging module like Semantic Web Services is still under development progress and we try to enhance the features and keep up with the latest advances. Known tourism, travel and weather ontologies will be included.

Acknowledgements

We would like to acknowledge the support of the Austrian National Bank (OeNB) for the work described in this paper by the generous funding of the BIM–Project (Business Information Management, Grant No. 11284) and the ZELESSA-Project (ZELESSA: An Enabler for Real-time Business Intelligence, Grant No. 11806).We are very indebted to all members of SemanticLIFE team for the various discussions and for their implementation work.

References

[1] A. Tversky (1977). Features of Similarity. Psychological Review 84(4):327-352.
[2] Ahmed et al., 'SemanticLIFE' - A Framework for Managing Information of A Human Lifetime, *Proc. of the Int. Conf. on Information Integration, Web-Applications and Services (IIWAS'04, Indonesia).*

[3] Brahim Medjahed, Boualem Benatallah, Athman Bouguettaya, Anne H. H. Ngu, Ahmed K. Elmagarmid, "Business-to-business interactions: issues and enabling technologies", *The VLDB Journal (2003) 12: pp 59–85*.

[4] Tim Berners-Lee, Semantic Web Road map, www.w3.org/DesignIssues/Semantic.html

[5] Burke, R. "Knowledge-based recommender systems". In J. E. Daily, A. Kent, and H. Lancour (eds), *Encyclopedia of Library and Information Science*, volume 69, 2000.

[6] L. Ardissono, A. Goy, G. Petrone, M. Segnan, and P. Torasso, "Tailoring the Recommendation of Tourist, Information to Heterogeneous User Groups", *Proceedings of OHS/SC/AH 2001, LNCS 2266, pp. 280–295, 2002*. Springer-Verlag 2002.

[7] Dario Cavada, Nader Mirzadeh, Francesco Ricci and Adriano Venturini, "Interactive Itinerary Planning with Trip@dvice", *Human-Computer Interaction — INTERACT'03 M. Rauterberg et al. (Eds.) IFIP, 2003, pp. 1105-1106.*

[8] L. Ardissono, A. Goy, G. Petrone, and M. Segnan, "A Multi-Agent Infrastructure for Developing Personalized Web-based systems". *Vol V, No. N, August 2003.*

[9] WC3, Web Service Activity http://www.w3.org/2002/ws/

[10] Fesenmaier D. R. , Ricci F., Schaumlechner E., Wöber K., Zanella C., "DIETORECS: Travel Advisory for Multiple Decision Styles", *in Proceedings of Enter conference, Helsinki, Finland, January 29 - 31 , 2003.*

[11] Lorenzi F. and Ricci F., "Case-Based Recommender Systems", *in J.Wang (ed.), The Encyclopedia of Data Warehousing and Mining, Idea Group Publishing, 2005.*

[12] Ricci F., Arslan B., Mirzadeh N., Venturini A., "ITR: a case-based travel advisory system", in *Proceedings of the 6th European Conference on Case Based Reasoning [ECCBR F2002],* Aberdeen, Scotland, September 4-7, 2002.

[13] A Min Tjoa, Amin Andjomshoaa, Ferial Shayeganfar, M. Shuaib Karim."Exploitation of Semantic Web in ERP Systems". The IFIP International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS 2006) April 24-26, 2006.

[14] Alan L. Montgomery, Shibo Li, Kannan Srinivasan, and John C. Liechty, "Modeling Online Browsing and Path Analysis Using Clickstream Data"

[15] Semantic Markup for Web Services (http://www.w3.org/Submission/OWL-S)

[16] Business Process Execution Language for Web Services (BPEL4WS) (http://www. 128.ibm.com/developerworks/library/ws-bpel/)

[17] WS-BPEL Extension for People: (http://www-128.ibm.com/developerworks/webservices/library/specification/ws-bpel4people)

Activation of Knowledge in an Integrated Business Process Support/Knowledge Management System

Ilia Bider[2], Lena Johansson[1], Erik Perjons[1], and Alexey Striy[2]

[1] Royal Institute of Technology, Department of Computer and Systems Sciences
Forum 100, SE-164 40 Kista, Sweden
lenaj@dsv.su.se, perjons@dsv.su.se
[2] IbisSoft AB, Box 19567, SE-104 32 Stockholm, Sweden
ilia@ibissoft.se, a_streey@yahoo.co.uk

Abstract. The paper is devoted to the issue of activation of knowledge in automated Knowledge Management Systems (KMS). According to the authors, activation of knowledge means that a system, based on the knowledge stored in it, automatically suggests a solution appropriate for a task at hand and/or guards against the user invoking inappropriate solutions. The paper discusses activation of knowledge, first, in general, and then, in a more specific manner, while applying general concepts to an integrated Business Process Support and Knowledge Management System (BPS/KMS) that is based on the state-oriented view on business processes. Activation of knowledge in such a system is done through rules of planning. The paper presents a classification of such rules, which is based on deontic logic concepts, and shows how rules of different categories can be used for activation of knowledge. The discussion is illustrated by an example already implemented in a working system. Some details of the current implementation of rules of planning are also presented in the paper.

1 Introduction

The paper is aimed at describing the progress in the INKA project devoted to integration of business process support with management of operational knowledge. By operational knowledge we mean knowledge needed for completing the everyday work, e.g. planning, reporting, communicating with colleagues and people outside the organization, etc. The INKA project was first presented at PAKM'04 [1]. The goals of the project and its initial achievements and setbacks are described in [1,2]. To give an idea about the project we repeat some parts of [1,2] below. However, the repetition concerns only the parts that are relevant to the central topic of this paper, i.e. activation of knowledge in computerized Knowledge Management Systems (KMS).

As was stated in [1], the main assumptions for the INKA project are as follows:

1. As far as operational knowledge is concerned, for the knowledge management to be of use in an organization, it should be seamlessly incorporated in everyday business activities. Thus we need an integrated IT system that supports both daily operations and knowledge management.
2. Operational knowledge is structured around business processes, such as processing an order, insurance claim, or bug in a software system. Therefore, for a business

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 13 – 24, 2006.

support system to be able to automatically gather and distribute knowledge, both the business and support system should be process-oriented. We will refer to such a support system as BPS/KMS (Business Process Support/Knowledge Management System).
3. An important prerequisite for a BPS/KMS to function in practice is that a majority of the staff actually use the system in a major part of their daily work.

According to [2], to facilitate the use of a BPS/KMS by a majority of the staff, the following functionalities should be incorporated in the system:

1. Provision of context in a form recognizable to the users. A context is all information a user needs to act in a particular situation. This includes current state of the process, historical information and future plans. Historical information, e.g. the events that lead to a current state, provides an explanation why it came about which might be necessary to know in complicated cases. Information on the future plans may be needed for not doing a job already planned for somebody else.
2. Structured knowledge base consisting of two parts:
   - Experience-based knowledge (EBK), which contains records on actions undertaken in the past together with description of the contexts in which these actions were completed and the outcomes of these actions. EBK is continuously fed with facts from the use of the system. Having EBK, a user working in a particular context can search for facts about how work in similar contexts was conducted in the past, and what was the outcome.
   - Generalized knowledge base (GK), which contains business rules that constrains and regulates all operations in an organization. The rules concern various aspects of the business, from the laws of the given country to fairly detailed rules about actions to be taken in a specific context.

To make the system even more attractive to the users, the knowledge stored in the system should be made active. By activation of knowledge, in general, we mean that the system suggests a solution for a task at hand (based on the context) rather than asks the user to search the knowledge base for the knowledge relevant to the context, and then decide on a solution based on the findings. It also means that the system guards against invocation of unacceptable solutions. Activation of knowledge constitutes the central topic of this paper.

2 Activation of Knowledge in an Integrated KMS/BPS

In respect to everyday work, the operational knowledge stored in a KMS can be used in two ways:

1. To decide on a course of action to take in the frame of the given business process instance (i.e. a process case), e.g. make a phone call, write a letter, send a reminder, rewrite a report, etc. The course of action should maximize the chances for successfully completing this process instance.
2. To avoid taking a course of action that can lead to failure, for example avoid actions that have lead to failure before, or actions that are prohibited by laws and/or regulations.

Activation of knowledge in a system means that the system gets a possibility to:

1. Suggest a course of action in a given context.
2. Prevent the user from taking the wrong course of action, if the system cannot give any recommendation in a particular context, or the user chooses not to follow the recommended course.

Both experience based knowledge (EBK), and generalized knowledge (GK) can be activated. Activation of the EBK requires that the system, first, searches the experience base for actions that have been undertaken in contexts similar to the current one. Then, based on the outcome of these actions, i.e. success or failure, it suggests some courses of action, or warns against others. Activation of EBK requires that the system has information about the outcomes. One practical way of implementing this functionality is by a user evaluating the outcome of each finished business process instance according to some failure-success scale, and storing this information in the system. We consider the activation of EBK to be quite a complicated issue, which is left outside the scope of the INKA project.

Activation of GK requires special structuring of the rules. First, it should be formally possible to identify the rules relevant to the context of the current situation. Secondly, rules should include actions to recommend or/and warn about. The specific structure of the rules depends on the conceptual/theoretical model upon which a BPS/KMS is built. Our BPS/KMS is built based on the state-oriented view on business processes [9]. A natural way of activating GK in such a system is via rules of (automatic) planning. The structure of such rules and their implementation in a system constitute the main topic of this paper.

To be of use for activation of knowledge, the rules of planning need to be classified and structured according to how they can be used for suggesting a proper course of action or warning against the improper courses of actions. Our classification is based on deontic logic concepts [6], policy-based thinking [7,10] and theoretical findings in the domain of Business Rules (BR) [13]. We classify the rules into 4 categories:

1. Obligations
2. Recommendations
3. Prohibitions
4. Negative recommendations

In the paper we will explain the meaning of this classification and how different kinds of rules can be used for activation of knowledge. We will also briefly overview an implementation of such rules in an operational BPS/KMS. The explanation will be done with the help of an example. Though the example we use may seem a bit artificial, we consider it quite representative for the problems we face when activating knowledge via rules of planning.

The rest of the paper is written according to the following plan. In section 3, we shortly review the main ideas of the state oriented view on business processes. In section 4, we suggest and explain the classification of rules of planning. In section 5, we demonstrate how various types of rules can be used for activation of knowledge by means of a simplified example. In section 6, we shortly review a current implementation of such rules in a BPS/KMS system. Section 7 overviews the related research, and, finally, Section 8 contains concluding remarks and directions of current and future research.

3 State Oriented View on Business Processes

The main concept of the state-oriented view on business processes is the process's *state* [9]. The process's state is aimed to show how much has been done to achieve the operational goal of the process instance, and how much is still to be done. A state of a process is represented by a complex structure that includes attributes, and references to various active and passive participants of the process, such as process owner, documents, etc, see Fig. 1 for an example. A state of a given process instance does not show what activities have been executed to reach it, it only shows the results achieved so far.

A goal of a business process can be defined as a set of conditions that must be fulfilled before a process instance can be considered as finished. A process state that satisfies these conditions is called a *final state* of the process.

The process is driven forward through activities executed either automatically or with a human assistance. An activity can be viewed as an *action* aimed at changing the process state in a special way. Activities can be planned first and executed later. A *planned activity* records such information as type of action (goods shipment, compiling a program, sending a letter), planned date and time, deadline, name of a person responsible for an action, etc.

All activities currently planned for a process instance make up its *operational plan* or to-do list, see Fig. 2 for an example. The plan lists activities, the execution of which diminishes the *distance* between the current state of the process instance and the *projected* final state.

The plan together with the "passive" state (attributes and references) constitutes a *generalized state* of the process; the plan being an "active" part of it. When an activity is executed, a process changes its generalized state. Changes may concern the passive and/or active parts of the state. At the minimum, the executed activity disappears from the plan. In addition, changes are introduced in attributes and references and/or new activities are planned to drive the process forward.

With regards to the generalized state, the notion of a *valid* state can be defined in addition to the notion of *final state*. To be valid, the generalized state should include all activities required for moving the process to the next stipulated state. A business process type can be defined as a set of valid generalized states. This definition can be converted into an operational procedure called *rules of planning*. The rules specify what activities could/should be added to/deleted from an invalid generalized state to make it valid. Using these rules, the process instance is driven forward in the following manner. First, an activity from the operative plan is executed and the state of the process is changed. Then, the operative plan is corrected to make the generalized state valid.

4 Classification of Rules of Planning

This section is devoted to classifying rules of planning and discussing how rules of different categories can be used for activation of knowledge. The basic ideas for our classification were taken from the works in three areas: deontic logic [6], policy-based thinking [7,10] and theoretical findings in the domain of Business Rules (BR) [13]. Our classification does not exactly follow any of the theories listed above, but is adjusted to our specific goal of activation of knowledge via rules of planning. We differentiate 4 categories of rules of planning:

1. *Obligation:* for a given class of the process states, some activities must be present in the process's plan. In case of absence, they are added. For example, suppose in an order processing process, all the goods have been delivered, but not invoiced. Then the *invoice* activity must be included in the plan.
2. *Recommendation:* for a given class of the process states, some activities are, normally, present in the process's plan. In case of absence, they are suggested for inclusion.
3. *Prohibition:* for a given class of the process states, some activities cannot be present in the process's plan. In case of presence, they are removed.
4. *Negative Recommendation:* for a given class of the process states, some activities are, normally, not present in the process's plan. In case of presence, they are suggested for removal.

Examples of various rules, and how a change of the rule strictness from, e.g. obligation to recommendation, can change the behavior of the system are discussed in the next section.

The wording in the names of categories above implies that the rules can be used for communication between the system and its users, for example, by *recommending* some actions, or *not recommending*, i.e. warning against others. While the rules can certainly be used in this way, we do not discuss this issue in the paper. We focus on *automated generation/modification* of the plan, and *automated correction* of the manual planning done by the user.

In our discussion on automated planning/correction we will differentiate 3 situations:

1. The process plan is empty, for example at the start of the given process instance, or after the execution of the last of the previously planned activities.
2. The process plan is not empty, and the current state of the process instance has been changed.
3. The user has manually changed the plan.

Note that situations 1 and 3, and 2 and 3 can be mixed. For example, the plan can become empty because the user manually deleted all activities from it, or when in a situation 2, the user manually corrected the plan directly in connection to (in the same session as) registering changes in the process state.

Consider which categories of rules of planning can be used in each of these situations:

1. The process plan is empty. Obligations and recommendations are treated equally to suggest a plan.
2. The process plan is not empty, and the current state has been changed, i.e. the user has registered changes just occured in the real world, or has executed a planned activity that changed the state. Then, as in the first situation, obligations and recommendations are treated equally to suggest new activities to be added. In addition, prohibitions and negative recommendations are treated equally to remove activities that no longer correspond to the new state of the given process instance.
3. The user manually changes the plan. Only obligations and prohibitions are used to correct the plan. Obligation will restore mandatory activities, while prohibitions will remove activities that are not allowed in the given state. Neither recommendations, nor negative recommendations are used in this case, as the user, by manually changing the plan (for example by removing a recommended activity or adding a not-recommended activity), has made an explicit choice about how to proceed.
4. When a mixed situation occurs, rules are used dependently on whether the given activity was added/deleted manually, or was in the plan before the last change of the state, see next section for details.

As we see from the discussion above, when the system does planning on its own it treats obligations and recommendations equally. The same is true for prohibition and negative recommendations. However, when the system corrects manual planning performed by the user, it interprets these categories in a different manner.

5 Examples

The main ideas of our approach are demonstrated and explained on an example of a process of organizing a meeting. This example was implemented as a specialized module in a general-purpose business process support system called *ProBis* [3]. The state of a meeting process can be represented as a screen capture in Fig. 1. Each meeting has a number of so called core participants (see "*Core participants*" in Fig. 1), meeting date and place. Fig. 2 represents the list of activities currently planned for the meeting process from Fig. 1. In this list, each core participant has an activity *Meeting* planned for him/her that indicates that he/she should attend a meeting at the specified date and time. The list on Fig. 2 represents the "normal" correspondence between the state and the plan. Below we consider several scenarios that ensure that such correspondence is imposed strictly or with some deviations.

Scenario 1 - Strict regulation. Core participants must attend, and only they are allowed to attend. This scenario can be described by a combination of obligations and prohibitions as follows:

1. *Obligation:* If a person belongs to the core participants, there should be an activity "Meeting" assigned to him/her in the plan.
2. *Obligation:* Date and time of a *Meeting* activity must be the same as prescribed by the process state.
3. *Prohibition:* A *Meeting* activity is allowed to be in the plan only if it is assigned to a core participant.

4. *Prohibition:* Only one *Meeting* activity per person is allowed in the plan for one given meeting process. There can be several meeting processes running at the same time, which means that any person can have several meeting activities in his calendar, each belonging to a different Meeting Process.

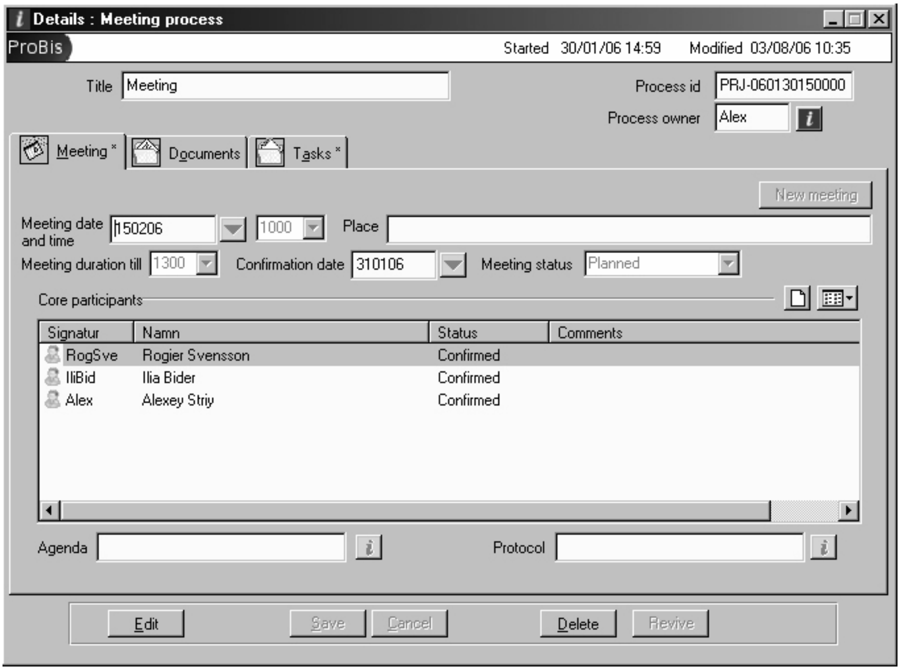

Fig. 1. State of the meeting process

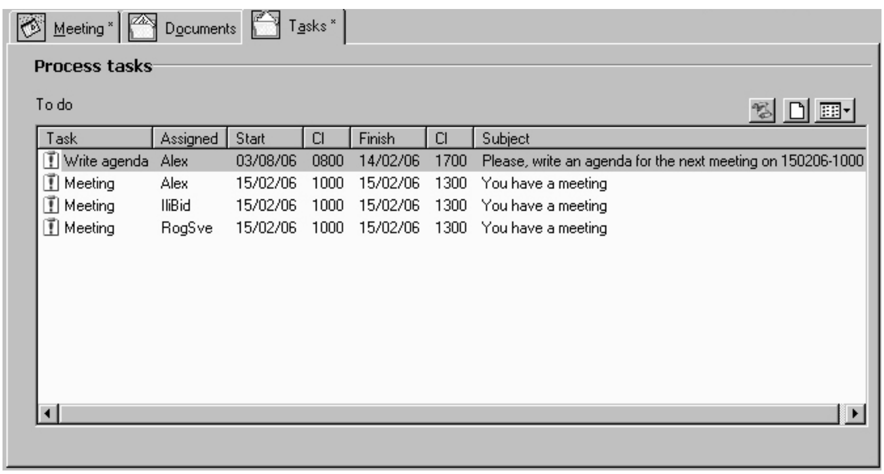

Fig. 2. Activities planned in the frame of the process instance

The rules are applied in the following manner:

a) If a person is added to the participants list, a new *Meeting* activity assigned to him/her is automatically added to the plan.
b) If this activity is later manually removed, it will automatically be added to the plan again.
c) If a person is removed from the participants list, his/her *Meeting* activity is also removed.
d) If a *Meeting* activity is manually added to the plan and assigned to a person who currently is not on the participants list, the activity will automatically be removed.
e) In addition, date and time parameters are always corrected to the actual date and time for the meeting as specified in the process state.
f) Multiple *Meeting* assignments to the same person in the same meeting process are reduced to one.

Note. Application of the rules described above is based on the following assumption. The user is allowed to manipulate the process plan freely. It means that he is permitted to add or delete any activities he wants. Rules of planning are applied immediately after he/she finishes his/her job, and presses the *Save* button. The application of rules could be made more sophisticated, so that a user is not allowed to delete a mandatory activity (task), however, we will not discuss this issue in more detail in this paper.

Scenario 2 – Guests allowed. Core participants must attend and guests are allowed. This scenario can be obtained from Scenario 1 by changing the strictness of Rule 3 from *Prohibition* to *Negative recommendation*:

3. *Negative recommendation*: A *Meeting* activity is not recommended to be in the plan if it is assigned to a person who is not a core participant.

The rules application would differ from scenario 1 only in situation (d):

d) If a *Meeting* activity is manually added to the plan and assigned to a person not on the participants list, the activity will stay in the list.

Scenario 3 – Permission to skip, but no guests. Core participants are recommended to attend, and only they are allowed to attend. This scenario can be obtained from Scenario 1 by changing the strictness of Rule 1 from *Obligation* to *Recommendation*:

1. *Recommendation*. If a person belongs to the core participants, it is recommended to have an activity *Meeting* assigned to him/her in the plan.

The rules application would differ from scenario 1 only in situation (b):

b) If this activity is later manually deleted, it will not appear again after the rules have been applied.

Scenario 4 – Permission to skip and guests allowed. Core participants are recommended to attend, and guests are allowed. This scenario can be obtained from Scenario 1 by changing the strictness of Rule 1 from *Obligation* to *Recommendation*, and the strictness of Rule 3 from *Prohibition* to *Negative recommendation*:

1. *Recommendation.* If a person belongs to the core participants, it is recommended to have an activity *Meeting* assigned to him/her in the plan.
3. *Negative recommendation:* It is not recommended to have a *Meeting* activity in the plan if it is assigned to a person who is not a core participant.

The rules application would differ from scenario 1 only in situations (b) and (d):
b) If this activity is later manually removed, it will not appear again after the rules have been applied.
d) If a *Meeting* activity is manually added to the plan and assigned to a person not on the participants list, the activity will stay in the list.

6 Implementation in ProBis

The approach of activation of knowledge via rules of planning has been implemented in a general purpose BPS/KMS called *ProBis* [3]. The planning system consists of a set of independent rules. Each rule is manually coded, but its inclusion in the system is done via an invocation table stored in the database. Rules can be activated/ deactivated by a system administrator via a special system screen.

In *ProBis*, a process state is represented as a tree structure; the tree relevant to the example from the previous section is shown in Fig. 3. In this structure, the process itself is represented as a root node, whereas child nodes represent various elements included in the definition of the process state, like meeting participants, planned activities etc. Each node, root, as well as child, has a set of attributes assigned to it.

Each individual rule in the invocation table is associated with a node in the process tree. A rule may be supplied with a pre-condition on attribute values of the node with which it is associated, or the upper nodes. A pre-condition can take into account current values of attributes as well as information about changes. If a pre-condition is supplied, the rule is applied only if the pre-condition yields true for a given process instance. The rule itself is implemented as a procedure written in a programming language for which *ProBis* has an API. This procedure may include additional conditions that should be met before any changes in the plan can be introduced by this rule. If the conditions are not met, the rule invocation does not affect the plan.

Application of rules of planning is governed by a so-called session principle. A session is started when a user presses the *Edit* button (see Fig. 1), or chooses an activity in the plan for execution. The session ends when the user presses the *Save* button; then all changes made on the screen are introduced in the database. Rules are applied after the *Save* button has been pressed, but before the changes introduced in the process state are stored in the database. Thus, the rules of planning can be considered as a kind of ECA rules, where ECA stands for Event-Condition-Action. In our case, pressing the *Save* button serves as an event trigger.

The process tree is traversed in the top down left to right manner starting from the root. Rules associated with each node are applied in an order defined in the invocation table: a rule can be executed either before traversing the sub-tree attached to the given node, or after traversing.

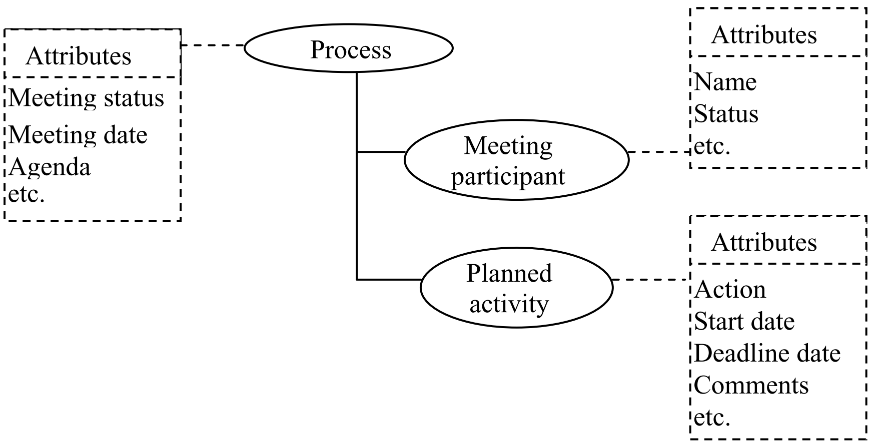

Fig. 3. Process tree structure

A rule associated with a node other than *planned activity* can only add new activities to the list, while a rule associated with a *planned activity* node can delete an activity being traversed or change the values of its attributes. Rules of the first kind implement obligations and recommendations, while rules of the second kind implement prohibitions and negative recommendations. Rules of the second kind are applied last.

7 Related Research

There is a common understanding that the success of automated KMS depends on the possibility of providing knowledge relevant to the situation with which the user currently deals [11]. This understanding shows itself in the appearance of a Knowledge Management research direction called Just-in-Time Knowledge Management (JiT-KM), see [8] for a definition. As [4,5] argue, business processes are the key to identifying the right time for knowledge provision, since business processes are context-giving, structuring elements prevalent in most organizations, see also [11].

Activation of knowledge in automated KMS, which is in the focus of this paper, goes just one step further, requiring the system to automatically propose actions, not only "open" relevant pages of a manual or provide templates to use.

As far as our approach to activation of knowledge is concerned, it is also related to a number of areas outside the field of Knowledge Management and Business Process Management. The basic concepts of obligation and prohibition come from deontic logic [6]. Its application to our rules of planning was inspired by policy-based thinking [10]. The main difference of our approach from [10] is that:

- The deontic concepts are integrated in the framework of the state-oriented view on business processes.
- We do not use the concept of permission; everything that is not prohibited is permitted. Instead, we use recommendations and negative recommendations to show various degrees of permissibility.

Another research area related to our work is Business Rules (BR). The literature on BR and its application to software design is vast, see, for example, [13] and its list of references. BR literature also categorizes rules according to the degree of obligation with which they should be followed. For example, [13] differentiate two types of constraint rules: *Mandatory Constraint* and *Guideline*. Business events that violate the *Mandatory Constraint* should be rejected, whilst violation of *Guideline* rules should raise a warning to the user. From the BR view, our rules of planning represent a special kind of BR aimed at providing help in running business processes.

One more area related to our work is Expert Systems (ES) [12]. Initially, the aim of ES was to help humans in finding solutions in cases where a large knowledge base was needed for finding a solution. Recently, a new field of ES, called Critiquing Systems (CS) has emerged [12]. A critiquing system critiques solutions chosen by humans, warning them when a solution might be wrong, and explaining why. From the ES perspective, our work consists of integrating both traditional ES and CS capabilities into business process support systems.

8 Conclusion

As we discussed in the Introduction, activation of knowledge in an automated KMS can increase the usability of the system, and thus make introduction of it in operational practice easier. This paper proposes a definition of what activation of knowledge in automated KMS means, and gives an example of how it can be implemented in a working system.

Our general definition states that activation of knowledge in an automated KMS means that the system can: (1) suggest a solution appropriate for a task at hand, and (2) prevent the user from invoking inappropriate solutions, if the system cannot give recommendations for a particular task, or the user chooses not to follow the system's recommendations.

The example concerns an integrated BPS/KMS built based on the state-oriented view on business processes. Activation of knowledge in the system is based on rules of planning. As we succeeded in implementing some rules in a real BPS/KMS, our approach deserves to be considered as a practically feasible alternative for activation of knowledge in a particular class of KMS systems. Rules of planning discussed in the paper will be included in the production version of *ProBis* in the nearest future.

Though some details of our approach to activation of knowledge are specific to a BPS/KMS based on the state-oriented view on business processes, they can also be of general interest for the reader. The main achievement here is that our rules of planning can formally differentiate obligations from recommendations, and prohibitions from negative recommendations.

Though some success in activation of knowledge via rules of planning has been achieved, a lot more is required to convert our research to a practically feasible methodology. Work in this direction includes fine classification of rules of planning in order to express the rules in a formal language instead of manually coding them in a programming language. So far, while screening the BR literature, we have not found a suitable language that we can use. Therefore, most probably, we will need to design one ourselves. This will be complemented with some kind of "inference" engine.

Another important direction in which we are engaged is ad-hoc planning, i.e. allowing the end user to add own rules in the frame of a particular process instance. For example, the user can request adding a notification activity to his plan when a certain state has been reached or not reached by some deadline.

Though in our work, we concentrate on automatic correction of the plan, we do not reject the needs of having an explanatory mechanism that will clarify to the user why certain activities have been added/removed, or are not appropriate for the current state of the process. This can constitute a new direction in our future research.

Acknowledgements. Writing of this paper was supported by the Swedish Agency for Innovation Systems (Vinnova) under the grant for the INKA project.

References

1. Andersson B., Bider I., Perjons E. Integration of Business Process Support with Knowledge Management - A Practical Perspective. In *Practical Aspects of Knowledge Management: 5th International Conference, PAKM 2004*. LNCS, Vol.3336, pp. 227-238, Springer, 2004.
2. Andersson B., Bider I., Perjons E. Business Process Support as a Basis for Computerized Knowledge Management. In *Professional Knowledge Management*, LNAI, Vol. 3782, pp. 542 – 553, Springer, 2005.
3. Andersson T., Bider I., Svensson R., “Aligning people to business processes. Experience report.” Software Process: Improvement and Practice (SPIP), V10(4), 2005. pp.403-413.
4. Cole K., Fischer O., and Saltzman P.: Just-in-time knowledge delivery, Communication of the ACM, 40 (7), p 49-53, 1997
5. Fenstermacher K.D.: A Process for Delivering Information Just in Time. In: Professional Knowledge Management, LNAI, Vol. 3782, pp. 679-687, Springer, 2005.
6. R.Hilpinen (ed) *Deontic Logic: Introductory and Systematic Readings*. Reidel, 1971
7. ISO/IEC 10746-2. Open Distributed Processing, Reference Model, Part 2, 1995.
8. Kerschberg, K. and Jeong, H. Just-in-Time Knowledge Management. In *Professional Knowledge Management*, LNAI, Vol. 3782, pp. 1-18, Springer, 2005.
9. Khomyakov, M., and Bider, I., “Achieving Workflow Flexibility through Taming the Chaos.” In OOIS 2000, pp. 85-92, Springer, 2000.
10. Lupu E., and Sloman M. *A Policy Based Role Object Model*. In the Proceedings of EDOC’97, pp. 36-47. Gold Cost, Australia, October 1997.
11. Pavassiliou, G., Ntioudis S., Abecker A., Mentzas, G. A.: Supporting Knowledge-Intensive Work in Public Administration Processes, Knowledge and Process Management, Vol. 10, No 3, pp. 164-174, 2003.
12. Silverman, B. G.: Survey of Expert Critiquing Systems: Practical and Theoretical Frontiers, Communications of the ACM 35: 106–127, 1992.
13. Wan-Kadir W.M.N., Loucopoulos P. “Relating evolving business rules to software design”. *Journal of Systems Architecture* 50, pp. 367–382, 2004.

Innovation Management in a Multi-national Corporation's Subsidiary of Ireland's Evolving Knowledge Economy

Gabriel J. Costello[1], Brian Donnellan[2], Michael L. Ginn[3], Colm Rochford[4], Eoin Whelan[2], and Susanna Xu[2]

[1] Galway-Mayo Institute of Technology, Galway, Ireland
gabrielj.costello@gmit.ie
[2] National University of Ireland, Galway
brian.donnellan@nuigalway.ie, eoin.whelan@nuigalway.ie, susannaxin.xu@nuigalway.ie
[3] Fielding Graduate University, Santa Barbara, CA U.S.A.
mikeginn@bethechange.com
[4] Platform Engineering Manager, APC Ireland
colm.rochford@apcc.com

Abstract. Innovation management is now seen as an important competitive advantage for Multi-national Corporation (MNC) subsidiaries located in Ireland's changing economy. This paper reports on the initial stages of a case study in the Operations division of American Power Conversion (APC) Ireland. The results of an innovation audit are presented that provide a reference point to begin the transformation to an innovative supply chain organization. The paper proposes the development of Networks of Practice to enable the diffusion of resulting innovations across the corporation.

Keywords: Innovation Management, Networks of Practice, Innovation Audit.

1 Introduction

The development of Ireland's knowledge-economy was initially driven by foreign direct investment (FDI) from North American multi-national corporations (MNCs) setting up manufacturing facilities to avail of low tax incentives, a young educated workforce and proximity to their growing number of European customers. However, this initially successful model is increasingly being threatened by the low cost economies of Eastern Europe, India and China. Irish enterprises rapidly need to build new sources of competitive advantage to sustain employment and standards of living. The Enterprise Strategy Group's report "Staying Ahead of the Curve" states that the application of research and development (R&D) and technology to the "creation of new products and services, now require comprehensive and intensive development and will mark the decisive new orientation of Irish enterprise policy" [1].

This paper reports the initial findings of a case study on the management of innovation within the Irish Operations function of the American Power Conversion (APC) Corporation. Firstly a literature review of innovation management is presented and

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 25 – 36, 2006.

the role of information and communications technology (ICT) to support the enabling and diffusion of resulting innovations is discussed. The next section provides an overview of the case study and the proposed research approach. The results of an innovation audit carried out in APC Ireland Operations are then presented and the emerging concepts, ideas and insights from this initial study are considered. Finally the conclusions, implication for practice, research limitations and future direction of the research are outlined.

2 Literature Review

2.1 Innovation Management

This section provides an overview of the current changing landscape and paradigms of innovation and knowledge management. One of the main challenges facing organizations that aspire to being innovative is that of dualism: mixing the need for operational efficiency in the present while at the same time trying to innovate successfully for the future. There is the problem of "opposing logics" between operating and innovating organizations. Furthermore, according to Ralph Katz, the main issues facing innovation managers is not the technical area but in overseeing the complex interplay and motivation of the people involved [2]. Eric von Hippel speaks about the democratization of innovation where product and service users increasingly have the ability to innovate for themselves and the resulting move from manufacturing-centric to user-centric innovation processes [3]. Software examples include open source development such as Linux and Apache Web server software. There are a number of product examples from the field of sporting equipment such as the development of high-performance surf boards in Hawaii. This shift in the locus of innovative activity has consequences for current business models and government policies which currently favor the "manufacturer". Consequently, von Hippel advocates the need to educate managers on the management of user-centered innovations as important innovations are often brought into an organization through informal channels such as attendance at conferences. Chesbrough [4] argues that in many industries the centralized approach to R&D, which he terms "closed innovation", characterized by in-company methodologies [5], has become outdated and must be replaced by "open innovation" which adopts external ideas and knowledge in conjunction with the internal process. A number of factors are influencing this change such as: the mobility of skilled people; the increasing presence of venture capital, emergent high-tech start-ups and the significant role of university research. One of his principles is that "not all the smart people work for us" and he advocates that the smart people within the organization connect with the smart people outside.

"Lean" is a supply chain term defined as the "enhancement of value by the elimination of waste" [6]. Because of where it originated it is commonly known as the Toyota Production System (TPS) and contains many Japanese terminology such as muda (waste) and poka-yoke (mistake-proofing). An analysis of many of the techniques employed revels that the methodology is an umbrella for many standard industrial engineering practices. The promoters of Lean thinking insist that it is not the latest quick-fix program but requires a five-year commitment for an organization to effect the desired transformation [6]. Such a relatively long term investment can provide significant stability in these turbulent times for manufacturing facilities. Lean also requires a critical organizational transition from top-down directives to bottom-up initiatives where managers become coaches and employees become pro-active and a move to more strategic buyer-supplier relationships [7].

The importance of the motivation of technical professionals is of paramount importance as evidence suggested that it is better to have a team with an A-rated motivations and B-rated capabilities than visa-versa [8]. In the related area of creativity, Nemeth [9] proposes that creativity begins with a questioning attitude and the ability to "look outside the box". Recent research in psychology indicates that teams can stimulate creativity and problem solving by being open to dissenting voices and minority viewpoints that in normal circumstances would be rejected or ridiculed and that "cult-like" corporate cultures stifle creativity. The work of von Krogh et al. [10] provide new ideas about how knowledge can be created in organizations and used for competitive advantage. However it is interesting to contrast many of the very tidy and mechanistic methodologies of Lean thinking with the very organic and colorful approaches of some companies that specialize in product design [11].The way in which information and knowledge is disseminated in an organization is very important especially in light of research quoted by Allen which shows an inverse relationship between contact of technologists with outside personal and technical performance [12]. The most effective model, he argues, is where the organization has key people or "technological gatekeepers" on which most people rely for information. These gatekeepers are mediators with the outside world in terms of relevant literature, links to academics and networks of practice. A number of paradigms in which innovation must be managed have been reviewed above; now the paper will examine ways in which ICT can support this process.

2.2 Knowledge Management and Innovation Networks

This section discusses how ICT supported networks can increase the innovative capacity of a firm and provide the capability to diffuse innovations across organizations. One of the underlying principles of the open innovation model, discussed in the previous section, is that not all the 'smart' people work in one team or one organization. They are distributed over multiple organizations and finding successful ways to work with them will lie at the heart of 'innovating' innovation [4]. ICT offers new tools to help in this meta-innovation by connecting 'smart' people and enabling them to exchange their knowledge. Individual learning and new knowledge creation occur when people combine and exchange their personal knowledge with others [13, 14]. Thus, ICT can contribute to the innovative capacity of a firm by connecting people from different disciplines and different institutions and allowing them to exchange their knowledge.

Knowledge management has been identified as critical for organizational success and sustained competitive advantage while ICT is proposed as the key to success for managing organizational knowledge. However, there is evidence that these knowledge management systems fail to recognize the importance of knowledge workers in their context. For example, Kelly et al. [15] investigated the use of a Lotus Notes-based application to share knowledge in a large professional services organization. The investment enjoyed limited success partly due to the formal nature in which the system was used. Many employees expressed anxiety about the way in which colleagues in other offices might interpret and use publicly available information about work activities. As a result, the information posted was usually uncontroversial, sanitized accounts of their work. Studies indicate that 50-70% of KM projects fail to meet objectives and this has been attributed to over reliance on IT [16]. Consequently, Kelly et al. [15] argue that management need to cultivate communities that share a social context within which they can interpret the contributions of others; norms and behaviors that reduce feelings of vulnerability and promote trust; and a sense of mutual solidarity as a motivation for participation.

Research on work practices has consistently shown that social relationships (or lack of them) are an important factor in knowledge flows. However, the paradox is that individuals generally form these relationships not according to what the formal organization dictates, but based on personal biases and preferences for collaborators who may be either inside or outside the firm. These relationships are the basis for informal, naturally occurring networks or "networks of practice" (NoPs). For this reason, employees are exploiting interactive communication technologies to develop networks of people with expertise and interest around a specific area of practice. Brown et al. [17] have distinguished between two types of networks, communities of practice (CoPs) and networks of practice (NoPs). CoPs consist of people who are informally, as well as contextually, bound by a shared interest in learning and applying a common practice. NoPs are a similar concept in that people have practice and knowledge in common but unlike CoPs, they are mostly unknown to each other. NoP members may be geographically distributed but use technologies such as listservs, bulletin boards and blogs to access another person and identify what expertise they have. Electronic ties are loosening the constraints of organizational structure and physical proximity to allow connectivity between individuals who would otherwise find it difficult to identify and sustain contact with others who share similar interests [18]. The ability to reach everyone in a NoP contrasts with the localized tight-knit relationships in a CoP. Consequently, NoPs act as a forum where participants can tell stories of personal experiences and discuss and debate issues relevant to their practice. Recent empirical findings by Teigland et al. [19] highlight the contribution NoPs can make to the innovative capacity of an organization. They examined whether individual creativity and performance were related to participation in various NoPs and found that high reliance by individuals on internal CoPs as sources of help results in lower levels of creativity. This suggests that participation in intra-organizational distributed NoPs enhances creativity as evidenced by the positive relationship between internal knowledge trading and both creativity and efficient performance. Individuals participating in internal distributed networks of practice are able to act as bridges between local CoPs, accessing non-redundant knowledge from other locations and integrating it with knowledge of their own [19].

There is increasing evidence that knowledge workers are utilizing ICT to create dynamic on-line discussions forums, rather than utilizing and contributing to static knowledge repositories. The BBC is one organization which has seen success from its

NoP initiatives. Rather than investing in an IT-based KM system, they installed simple bulletin board technology in order to move knowledge around rapidly. The site currently gets 450,000 page views a month from 8,000 unique users - startling in an organization of 25,000. The primary role of the forum is to ask questions and get answers but according to the BBC's Chief Knowledge Officer; "the board wouldn't have taken off if it were restricted to the dry discussions of pure business. The board has entered the daily life of BBC employees because it's fun and interesting as well as useful" [20]. This section has provided an overview of the role and opportunity of ICT for knowledge and innovation management. Now the paper will provide the context in which the research is being carried out and the initial approach to the work.

3 Research Method

3.1 Research Context: The Case

The case study is based in APC, Ireland a subsidiary of the American Power Conversion (APC) Corporation. APC designs, manufactures and markets back-up products and services that protect hardware and data from power disturbances. The explosive growth of the Internet has resulted in the company broadening its product offerings from uninterruptible power supplies (UPS) to the high-end InfraStruXure^{TM} architecture in order to meet the critical availability requirements of internet service providers (ISP) and data-centers. This modular design integrates power, cooling, rack, security, management and services, which allows customers to select standardized modular components using a web-based configuration tool. APC reported sales of $2 billion in 2005, globally employs approximately 7,000 people and is a Fortune 1000 company. The Corporation aims to set itself apart from the competition in three areas: financial strength, innovative product offerings and efficient manufacturing [21]. However, APC's president and chief executive officer Rodger B. Dowdell, Jr. had indicated, in recent financial reports, that the company needs to implement significant improvements in manufacturing and the supply chain [22]. According to the CEO, the company must work to develop a "lean, customer-centric, ambidextrous organization" in order to reach "optimal efficiencies in our processes" [23].

APC has two locations in the West of Ireland that serve the European, Middle East and Africa (EMEA) region. The company announced a streamlining of its operations in Ireland in June 2006. The Manufacturing Operations site, based in Castlebar, employs approximately 150 people and a number of functions including sales, information technology, business support and R&D are situated in Galway with a workforce of approximately 300. The widening of focus from the manufacturing of discrete products, such as UPS, to the delivery of customized InfraStruXure^{TM} solutions provides both challenges and opportunities for the Operations function. Responding to the challenge set by the CEO, a Lean Transformation project was set-up in the Castlebar campus in February 2006 with a cross-functional team of twelve members drawn from Management, Engineering, Manufacturing, Materials Planning, Quality, and Logistics functions. One objective of the Lean project team is to quickly deliver the message that Ireland is responding to, and leading, the corporate initiative and to provide a platform for the Irish subsidiary to obtain a reputation as an innovative location. Initial corporate feedback is that this project is "ahead of the curve" in terms of the other regions. In Ireland, a "Knowledge Exchange Forum" is promoted by the Platform Engineering group as part of the EMEA initiative to educate peer groups. This forum provides the opportunity for engineers to meet with customers and academics to trade knowledge concerning latest product development and topics of mutual interest. In a related program, APC run a User Group community for InfraStruXure™ customers.

3.2 Theoretical Considerations

This section will review the characteristics and assumptions, presented in the literature, of two possible research approaches to the study of innovation management: the case study and action research. Case study is an exploration of a case over time through detailed, in-depth data collection involving multiple sources of information rich in context [24]. Dube and Pare divide case study attributes into three main areas: research design (which includes experimental design and research questions), data collection (data collection methods and tactics for enhancing reliability and validity), and data analysis (process description and data analysis) [25]. According to these researchers, 87% case studies are done from a positivist philosophical perspective, with 12% being interpretive, and 1% critical. This positivist perspective is accompanied by a broad commitment to the idea that the social sciences should emulate the natural sciences [26]. Here "the researcher is seen to play a passive, neutral role, and does not intervene in the phenomenon of interest." An interpretive perspective addresses meaning, understanding, and interpretation, in a systemic and methodical way, and in the process "yields much of the desire to predict and control upon which positivist science rests its claims" [27]. Critical science seeks to recall both the positivistic potential to support emancipation as well as the capacity to develop mutual understanding through the use of language enabling people to cooperate more effectively [28]. In sharp contrast to the positivistic case study, action research is a post-positive approach [29], where the researcher is directly involved in planned organizational change. "Unlike the case study researcher, who seeks to study organizational phenomena but not to change them, the action researcher is concerned to create organizational change and simultaneously to study the process" [30]. Action research is empirical, yet interpretive; experimental yet multivariate; observational yet interventionist. To a positivistic perspective this seems very unscientific [29]. Dube and Pare note that while case study and action research can share many characteristics (such as a natural setting as source of data, the researcher's central role in data collection, and a focus on participant perspectives), their objectives and inherent challenges as well as the criteria by which to judge their quality are quite different. In contrast to a positivist's neutrality, an action researcher intends to help the system change by helping it to gather information it needs in order to change. Bentz and Shapiro continue to explain that this follows an "assumption that a system is more likely to change if it gathers its own information about its problems, potential, future direction, and so on" [31]. The researcher helps the system plan its actions and conduct fact-finding procedures so it can learn from them, become more skillful, set more realistic expectations, and discover better ways of organizing. This section has discussed two possible research approaches to study the management of innovation in APC, Ireland: the next section will explain why case study methodology has been initially chosen as the best fit.

3.3 Research Method: Case Study Design

Yin [32] defines a case study as an "empirical enquiry that investigates a contemporary phenomenon within its real-life context" and where a "how or why question is being asked about a contemporary set of events over which the investigator has little or no control". The initial research aim of this study was to consider the human and technological factors involved in the management and diffusion of innovation. To this end the following preliminary research questions were posed:

- RQ1: How well does APC Ireland Operations currently manage innovation?
- RQ2: How is a culture of innovation developed in APC Ireland Operations?
- RQ3: How is the diffusion of resulting innovations to the wider APC multinational corporation (MNC) facilitated?

This paper deals with the first of these research questions. It is considered that APC Ireland meets the case study criteria in the real-life context of both the need for the region to increase levels of innovation and APC's requirement for the transformation to a lean innovative a supply chain.

4 Innovation Audit

This section provides the results of an innovation audit carried out in APC Ireland Operations and a discussion of the emerging concepts from this initial study. Tidd et al. [33] propose that innovation must not be seen as a lottery but as a continuous improvement process and point out that based on recent research on innovation successes and failures, a number of models have been developed to help assess innovation management performance. Such self-assessment tools have been widely used in the area of total quality management (TQM) in order to benchmark an enterprise against best in class, for example, the Malcolm Baldrige National Quality Award [34]. In order provide some initial reference point on innovation management it was decided to use the self-assessment tool and audit framework developed by Tidd et al. [33] to obtain a response to the question "How well does APC Ireland Operations currently manage innovation?". As this was part of the exploratory phase of the research, the intention of the questionnaire was to discover ideas and insights and not test any hypothesized causal relationships [35]. The survey consists of forty questions using a Likert scale to score each statement between 1 (= untrue) and 7 (= very true) in order to gauge five dimensions of innovation management: Strategy, Processes, Organization, Linkages and Learning. As this was a small scale and focused survey, it was decided to limit the questionnaire to the population of management and engineering staff in APC Ireland Operations [36]. Questionnaires could be completed by email or

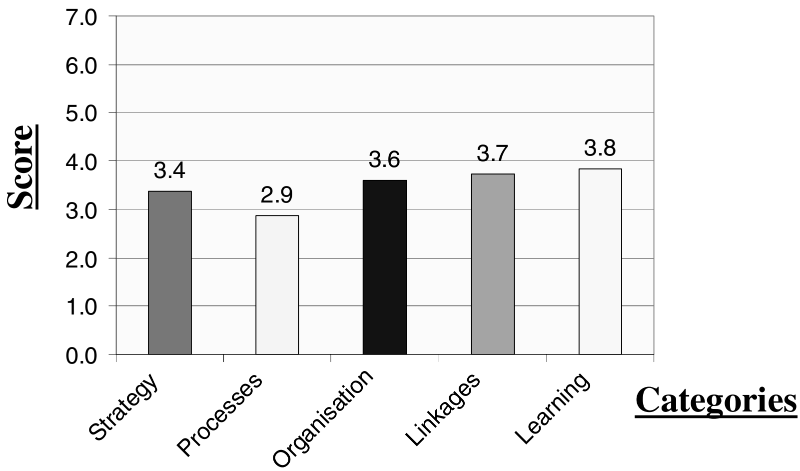

Fig. 1. Innovation Self-assessment Audit Results

on a hardcopy and anonymity was guaranteed. The analysis of the responses showed that out of a possible score of 7, the actual scores ranged between 2.9 and 3.8 for the five dimensions. Clearly it can be seen from the self-assessment indicator that there is considerable room for improvement in the management of innovation in the location.

The next step of the methodology was to represent the results of the audit in terms of four possible "archetypes" of innovation capability in order to provide a reference point for future continuous improvement. The taxonomy ranges from Type 1: where an organization doesn't know what or how to change, to Type 4: where a firm sees itself as having the capability to generate and absorb innovation. This "snapshot" is shown in figure 2 and the audit results indicate that APC Ireland Operations sees itself as being a Type 2 firm: know they need to change but unsure about how to bring this about.

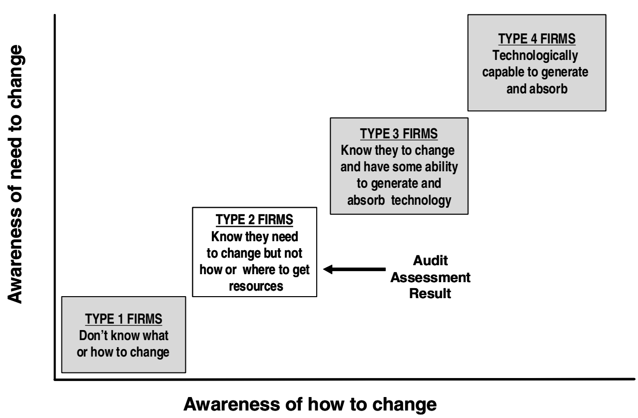

Fig. 2. Innovation Capability Self-assessment

The above section described the use of an innovation audit questionnaire to survey the Operations management and engineering personnel on innovation management capability. The results indicate that the organization is willing to develop competence in this area but need a strategy, roadmap and resources to achieve this.

5 Discussion

5.1 Significance of Present Work

Using the four broad categories, the "4Ps" of innovation- proposed by Tidd et al. [33], APC can be regarded as being an innovative company in the area of product innovation (for example the success of InfraStruXure™) and in position innovation (the relatively new markets of data centers and server farms). But there is a need for improvement in process innovation (delivery of products and services) and paradigm innovation (organizational models). This context provides both challenges and opportunities for APC Ireland Operations function. The fact that APC Ireland undertook an audit, to establish benchmark data on innovation and knowledge management, indicates that the site is willing to embrace the challenge presented by the CEO. Also, the quick response in setting-up a local Lean Transformation project to support the corporate strategy which is globally "ahead of the curve" is very significant. Browne et al. stress the importance of quick-wins and projects to encourage a culture of innovative actions [37].

5.2 Implications for Knowledge Management Practice

Lotus Notes is the collaborative software system used by APC to manage its knowledge flows. It provides and environment for asynchronous group work: where collaborators have different or independent work patterns. The software solution provides access to scheduling, messaging, and the sharing of documents and data. The present corrective action process embedded in the Lotus Note databases is designed to facilitate tightly controlled feedback procedure but is unlikely to inspire the diffusion of innovations. In order to increase its innovative capacity APC Ireland could lead the creation of a globally distributed NoP focusing on lean manufacturing techniques. The role of the technology is to create a global NoP by connecting these CoPs together. This ensures a rich diversity of skills, abilities and cognition among network participants which should enhance the opportunities for innovation [38]. As in the BBC example, the technology used can be simple and low cost. But it most be borne in mind that the social infrastructure is very important when using technology to facilitate better communication and knowledge sharing within and between geographically distributed groups. Significantly, research has shown that NoPs will only thrive when it is free to decide its own agenda and free from managerial control. To further increase its innovative capacity, APC Ireland should consider establishing supply chain focused communities accessible through shared knowledge environments that mirror the company's product focused User Communities described earlier.

5.3 Implications for Research and Limitations of the Study

The benefits of case study include a depth of understanding that is beyond that available from large-scale survey research, and that a focus on one case reduces travel costs, eases access issues, and in general makes complex in-depth research doable. The risks of a case study include a prolonged engagement costing additional time and money, and that this investment may be lost if the project is ended before being completed. Finally, over time the desired level of objectivity of the experimenter can be compromised given a significant level of communication with the case under study. Action research is another approach, different in several important ways, which could be considered for structuring research beyond the current effort. This option is perhaps made more attractive when it is seen that the current case study would be a partial fulfillment of the initial step of identifying a problem as called for in the action research approach as noted in the above literature review. Many of the risks of action research are the same as the risks encountered in a case study. Another risk of action research is that the researcher will be swept up in the effort to solve the presenting problem and abandon the rigor required to produce new scientific findings.

The innovation audit presented in this study provides evidence that the use of such a framework can provide a quick snapshot for practitioners that provides a reference point to launch a continuous improvement program for innovation and knowledge management. The methodology of the innovation audit raises a number of questions concerning the reliability and validity of the findings. The framework would require additional work to develop it beyond its use here as a vehicle for emerging concepts and the creation of ideas. The increasing move to an "open innovation" model would suggest that this paradigm needs to be incorporated into innovation audit frameworks to fill the gap that presently exists in the literature.

6 Conclusions

This paper has provided some preliminary results, findings and reflections from an ongoing case study of innovation management in the Operations function of a subsidiary of the APC Corporation located in Ireland. Literature reviews of the current context of innovation management and the ICT enablers of knowledge management were presented. An overview of the subsidiary was then provided that placed it in the context of the global Lean Transformation program being undertaken by APC. The challenges of Operations management in turbulent times were then discussed as well as the opportunities for a motivated team to take the initiative and be seen as corporate role models for innovation. The importance of the long-term commitment to a location required by "lean thinking" was highlighted. The results of a self-assessment innovation audit were then presented with the conclusion that it provided a reference point for the next steps: the development of an innovation strategy, roadmap and the allocation of resources. The paper proposed that the creation of Networks of Practice (NoPs) can provide APC Ireland with the ability to enable innovation within the Operations functions and diffuse the resulting innovations throughout the corporation. Finally the significance of this research for the management of innovation in MNC subsidiaries and the associated implications for knowledge management were presented. Future work is suggested on the role of innovation audits both as an aid to practitioners and as a research method.

Acknowledgements

The authors would like to acknowledge the assistance of Bronagh Collins and Derek Brogan in carrying out the innovation audit.

References

1. Enterprise Strategy Group, *Ahead of the Curve, Ireland's Place in the Global Economy.* 2004, Forfás Secretariat: Dublin 2004.
2. Katz, R., *Introduction*, in *The Human Side of Managing Technological Innovation: A Collection of Readings*, R. Katz, Editor. 2004, Oxford University Press: Oxford.
3. von Hippel, E., *Democratizing Innovation*. 2005, Massachusetts: The MIT Press.
4. Chesbrough, H.W., *Open innovation : the new imperative for creating and profiting from technology* 2003, Boston: Harvard Business School.
5. Cooper, R.G. and E.J. Kleinschmidt, *Stage Gate Systems for New Product Sucess.* Marketing Management, 1993. **1**(4): p. 20-29.
6. Womack, J.P. and D.T. Jones, *Lean thinking : banish waste and create wealth in your corporation*. 2003, London: Free Press.
7. Lamming, R.C., P.D. Cousins, and D.M. Notman, *Beyond vendor assessment : Relationship assessment programmes.* European Journal of Purchasing & Supply Management, 1996. **2**(4): p. Pages 173-181.
8. Katz, R., *The Motivation of Professionals*, in *The Human Side of Managing Technological Innovation: A Collection of Readings*, R. Katz, Editor. 2004, Oxford University Press: Oxford.
9. Nemeth, C.J., *Managing innovation: When less is more*, in *The Human Side of Managing Technological Innovation: A Collection of Readings*, R. Katz, Editor. 2004, Oxford University Press: Oxford.
10. von Krogh, G., K. Ichijo, and I. Nonaka, *Enabling knowledge creation : how to unlock the mystery of tacit knowledge and release the power of innovation* 2000, Oxford: Oxford University.
11. Kelley, T., *The Art of Innovation: Lessons in creativity from IDEO, America's leading design firm* 2001, New York Doubleday.
12. Allen, T.J., *Communication Networks in R&D laboratories*, in *The Human Side of Managing Technological Innovation: A Collection of Readings*, R. Katz, Editor. 2004, Oxford University Press: Oxford.
13. Kogut, B. and U. Zander, *Knowledge of the firm, combinative capabilities, and the replication of technology.* Organization Science, 1992. **3**(3): p. 383-397.
14. Nahapiet, J. and S. Ghoshal, *Social capital, intellectual capital, and the organizational advantage.* Academy of Management Review, 1998. **23**(2): p. 242-266.
15. Kelly, S. and M. Jones, *Groupware and the social infrastructure of communication.* Communications of the ACM, 2001. **44**(12): p. 77-79.
16. Baxter, A., *Keeping the know-how of a retiring generation*, in *Financial Times*. 2006: London. p. 1.
17. Brown, J.S. and P. Duguid, *The social life of information*. 2000, Boston: Harvard Business School.

18. Van Baalen, P., J. Bloemhof-Ruwaard, and E. Van Heck, *Knowledge sharing in an emerging network of practice: the role of a knowledge portal.* European Management Journal, 2005. **23**(3): p. 300-314.
19. Teigland, R. and M. Wasko, *Integrating Knowledge Through Information Trading: Examining the Relationship Between Boundary Spanning Communication and Individual Performance.* Decision Sciences, 2003. **34**(2): p. 261-287.
20. Weinberger, D., *The BBC's low tech approach to knowledge management*. 2005, (available on-line through http://www.knowledgeboard.com/item/1627, last accessed August 2006)
21. APC, *American Power Conversion Corporation* 2006, (available on-line through http://www.apcc.com/).
22. APCC Results, *American Power Conversion Reports Record Revenue for the Fourth Quarter and Full Year 2005.* (available on-line through http://www.apcc.com/corporate/press_room/index.cfm).
23. APCC Results, *American Power Conversion Reports First Quarter 2006 Financial Results. 2006, (available on-line through http://www.apcc.com/corporate/press_room/index.cfm).*
24. Creswell, J.W., *Qualitative inquiry and research design: Choosing among five traditions.* 1998, Thousand Oaks, CA: SAGE Publications.
25. Dube, L. and G. Pare, *Rigor in information systems positivist case research: Current practices, trends, and recommendations.* MIS Quarterly, 2003. **27**(4): p. 597-635.
26. Lee, A., *A Scientific Methodology fro MIS Case Studies.* MIS Quarterly 1989. **March 1989**: p. 33-50
27. Hatch, M.J. and D. Yanow, *Organization theory as an interpretive science*, in *The Oxford handbook of organization theory: Meta-theoretical perspectives*, H. Tsoukas and C. Knudsen, Editors. 2003, Oxford University Press: New York. p. 63-87.
28. Willmott, H., *Organization theory as a critical science? Forms of analysis and 'new organizational forms'* in *The Oxford handbook of organization theory: Meta-theoretical perspectives*, H. Tsoukas and C. Knudsen, Editors. 2003, Oxford University Press: New York. p. 88-112.
29. Baskerville, R.L. and A.T. Wood-Harper, *A critical perspective on action research as a method for information systems research.* Journal of Information Technology, 1996. **11**: p. 235-246.
30. Baburoglu, O.N. and I. Ravn, *Normative action research.* Organizational Studies, 1992. **13**(1): p. 19-34.
31. Benz, V.M. and J.J. Shapiro, *Mindful Inquiry in Social Research*. 1998, CA: SAGE Publications: Thousand Oaks.
32. Yin, R.K., *Case study research : design and methods* 1994, London Sage Publications.
33. Tidd, J., J. Bessant, and K. Pavitt, *Managing innovation : integrating technological, market and organizational change /.* 2005, Chichester: John Wiley & Sons.
34. Baldridge, *Baldridge National Quality Program.* 2006, (available on-line through http://www.quality.nist.gov/index.html).
35. Ghauri, P. and K. Gronhaug, *Research Methods in Business Studies : A Practical Guide*. Second Edition ed. 2002, Harlow: Pearson Education Limited.
36. Gillham, B., *Developing a Questionnaire*. 2000, London: Continuum.
37. Browne, J., et al. *"Innovation Management for Product and Process Development". Proceedings of the International Conference of Information Technology in Business, August 2000, Beijing, China* 2000.
38. Justesen, S., *Innoversity in communities of practice*, in *Knowledge Networks: Innovation Through Communities of Practice*, P. Hildreth and C. Kimble, Editors. 2004, Idea Group Publishing: London. p. 79-95.

Ontology-Based Business Knowledge for Simulating Threats to Corporate Assets

Andreas Ekelhart, Stefan Fenz, Markus D. Klemen, A. Min Tjoa, and Edgar R. Weippl

Secure Business Austria – Security Research, A-1040 Vienna, Austria
{aekelhart, sfenz, mklemen, atjoa, eweippl}@securityresearch.at
http://www.securityresearch.at

Abstract. We propose a security ontology, to provide a solid base for an applicable and holistic IT-Security approach for SMEs, enabling low-cost threat analysis. Based on the taxonomy of computer security and dependability by Landwehr [ALRL04] and the threat classification according to Peltier [Pel01], a heavy-weight ontology can be used to organize and systematically structure knowledge on threats, safeguards, and assets. The ontology is used in an organization to capture business knowledge required for and created during a security risk analysis where instances of concepts are added to the ontology to allow the simulation of different attack and disaster scenarios. Each scenario can be replayed with a different protection profile as to evaluate the effectiveness and the cost/benefit ratio of individual safeguards.

1 Introduction

IT-Security is no longer limited to access control or preventing the classical virus attack; applied IT-Security also has to consider social engineering, acts of nature beyond human control, industrial espionage or physical attacks. With the need to implement IT-Security measures in almost every environment and faced with the growing application scope, it becomes increasingly difficult for experts of different domains to understand each other and to use a precisely defined terminology.

Nowadays, a well working IT-infrastructure is business critical for almost every enterprise and those who take IT-Security seriously spend a lot of money on maintaining and securing this infrastructure.

Compared to large companies, Small and Medium Sized Enterprises (SMEs) are usually not financially able to employ an own IT-division to plan, implement and monitor a holistic IT-Security concept. In a lot of cases only one or two employees are responsible for maintaining and securing the entire IT-infrastructure. Mostly, this circumstance leads to overwhelmed IT-administrators who are not able to guarantee the implementation of a holistic IT-Security concept. [Hau00] summarized the problems of SMEs regarding the IT-Security aspect:

– Smaller IT budget, relative to total budget as well as in absolute figures
– Less IT knowledge, information technology is often looked after by employees from other departments

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 37–48, 2006.

– IT is not considered as important as within larger enterprises although more and more core processes are processed by IT elements
– IT environments are not homogeneous

In many cases these points cause a very poor implemented IT-Security concept and a security ontology can help in a first step, to clarify the meaning and interdependence of IT-Security relevant terms [Don03]. Beside the term definition we have also to integrate the dependability of threats, countermeasures and resources. The integration of these dependabilities is necessary to model events which threaten existing resources and for each threat we have to model proper countermeasures which are able to secure the resources. Such an ontology combined with an user interface would enable SMEs, to run their risk management and threat analysis for low costs and without an expensive audit program like CobiT [COB06] or ISO17799 [ISO06].

2 Security Ontology

The entire security ontology consists of three parts: The first part is derived from the security and dependability taxonomy by Landwehr [ALRL04], the second part describes concepts of the (IT) infrastructure domain and the third part provides enterprises the option to map their persons and internal role models. The taxonomy is designed in a very general way, and so it may easily be extended with additional concepts.

The ontology is coded in OWL (Web Ontology Language [OWL04]) and the Protege Ontology Editor [Pro05] was used to edit and visualize the ontology and its corresponding instances. The following subsections describe the parts in more detail:

Attribute Ontology. Each instance in the *Threat* ontology impacts n security attributes but the actual impact is not expressed in numbers. In fact the ontology shows which threats influence certain security attributes which is useful if a company wants to prioritize the IT-Security strategy regarding specific attributes.

Threat Ontology. Figure 1 shows an excerpt of the most important parts of the *Threat* ontology and the relevant relations to the other ontologies. As already described above we derived the fundamental structure from [ALRL04] and [Pel01] respectively and extended it with own concepts and relations. The *Threat* ontology with its various relations, represents a central part of the entire security ontology and makes the mapping of threats including proper countermeasures, threatened infrastructure and proper evaluation methods possible. The most important relations in a nutshell: Any instance of concept *sec:Threat* or one of its sub-concepts affects n instances of concept *Attribute* (e.g. Availability, Integrity, Maintainability) and with *sec:preventedBy* and its inverse relation it is possible to map mitigating countermeasures to a certain threat. To enable the mapping of threatened infrastructure to a defined threat the relation *sec:threatens* was introduced, where each threat threatens n infrastructure elements. Last but not least

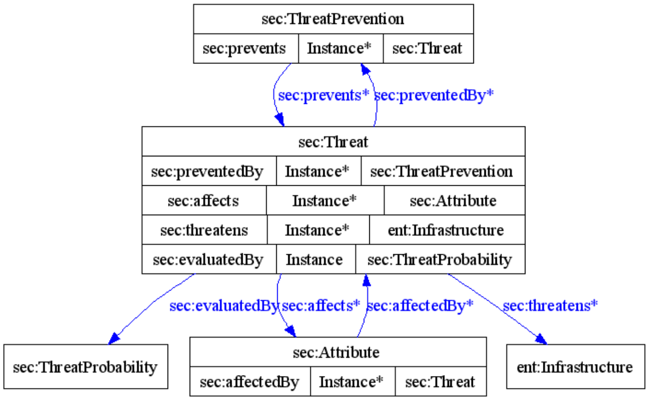

Fig. 1. Threat ontology - concept relations

the *sec:evaluatedBy* relation enables the user to map certain evaluation methods (e.g. probabilistic methods, qualitative methods, quantitative methods) to a defined threat class to support the user and the ontology-consuming application respectively at the risk analysis. Figure 2 shows a practical implementation of the computer virus instance *Tequila* and provides the reader with real world values regarding the computer virus concept. It can be seen that each network and computer device is described by its physical (in our example the development server and the switch are located in room R0202) and virtual (location within the network ⇒ development server is connected to the switch) location. If there would be no anti virus program installed on the development server and the affected operating system would be equal to the installed operating system the ontology reasoner would connect with the *sec:threatens* relation the instances *Tequila* and *DevelopmentServer* to indicate a potential threat. *sec:affects* and its inverse relation maps a certain threat to the corresponding security attributes and vice versa. In our example availability, reliability and integrity are directly affected by the computer virus instance *Tequila* and it has to be noted that the computer virus threat acts only as an example for further threats which can be modeled in a similar way.

Means Ontology. Figure 3 shows an excerpt of the *Means* ontology and we can see that relation *sec: requires* connects the concepts *sec:TechnicalThreatPrevention* and *sec:ThreatPrevention* to indicate that some technical threat prevention instances are requiring instances of the *sec: ThreatPrevention* concept. The security ontology needs this relation to determine countermeasures which consists of various elements such as a fire extinguisher system and a smoke detector were both elements are necessary to ensure an effective countermeasure.

Infrastructure Ontology. Figure 4 shows a fragment of the infrastructure part of the security ontology and due to the size of the infrastructure ontology

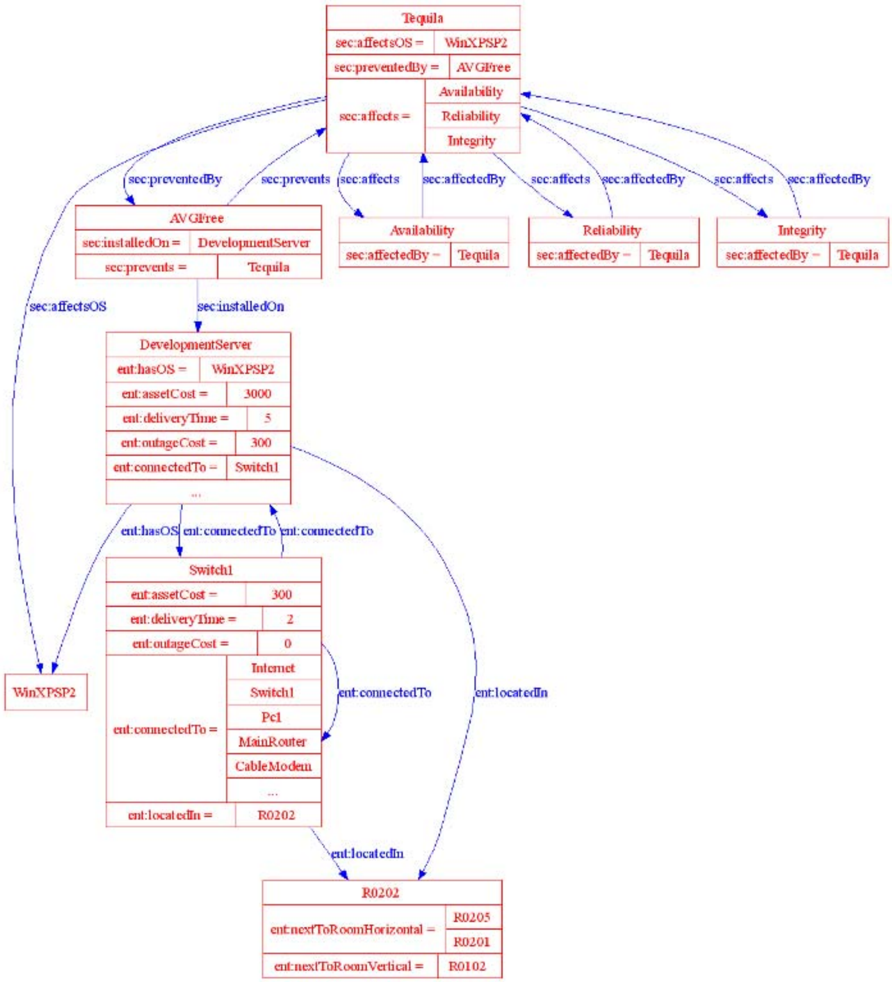

Fig. 2. Computer virus instance *Tequila* and corresponding relations

it is not possible to show the entire visualization and so we extracted the most relevant part. The company building with its corresponding floors and rooms can be described by using the infrastructure framework and to map the entire building plan exactly on the security ontology, each room is described by its position within the building. The ontology is able to reconstruct the entire building virtually by the following relations: (1) *ent:hasFloor* describes which instances of concept *Floor* can be found in a certain building (2) *ent:isUnderFloor* was designed to determine those floors which are located under a certain floor (3) *ent:isAboveFloor* defines the same type of knowledge as (2) but in the inverse direction (4) *ent:nextToRoomHorizontal* and *ent:nextToRoomVertical* define those rooms which are horizontal/vertical next to a certain room.

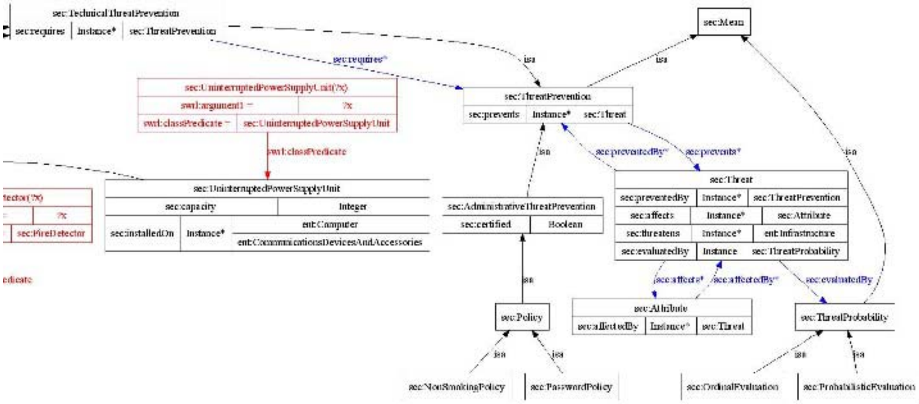

Fig. 3. Means ontology - concept relations

Role Ontology. The integration of roles is necessary to map the correct hierarchies to the ontology. The current role taxonomy includes the *ent:administers* attribute that is used to specify for which instances of *ent:ComponentForInformationTechnology* an administrator (instance of *ent:Administrator*) is responsible for. Additional relations depending on the further usage are thinkable and can be implemented without any effort if needed.

Person Ontology. This sub-ontology represents a simple listing of natural persons who are relevant for the modeling of certain security issues. Every person holds n roles and Figure 5 shows the most important concept relations of the *Person* taxonomy: the *ent:usedBy* relation maps computers, personal communication devices, etc. to its regular users and enables the ontology to relate those concepts for a further threat simulation (e.g. the calculation of non-productive personnel costs in the case of broken computers and/or communication devices).

2.1 Ontology Maintenance

The goal of this approach is to allow a detailed threat analysis, without requiring a lot time and too much expertise. Experts are only needed for the framework creation (at the first two levels). Three levels exist:

1. Defining the concepts of the ontology (including infrastructure, personnel, roles and disasters concepts), must be obviously done by experts.
2. Based on these concepts domain experts have to provide individuals (e.g. disasters, safeguards) and their attributes (e.g. spread time, damage, probability of occurrence).
3. At this stage a solid framework exists which can be utilized by companies. It is their task to model their company and this process will be done by the IT administrator or in further versions automatically based on building plans and the information extracted by automatic IT infrastructure recognition

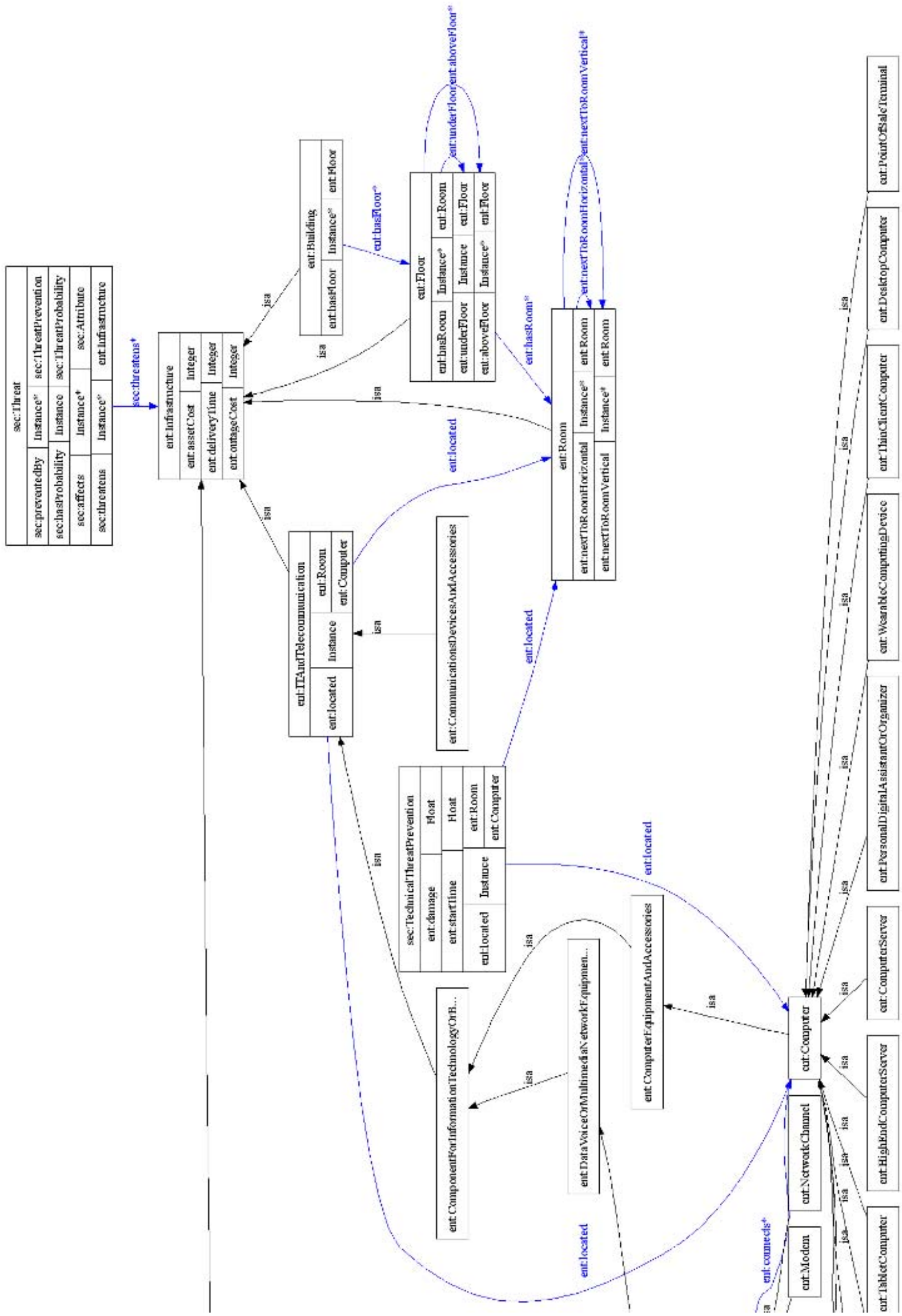

Fig. 4. Infrastructure ontology - concept relations

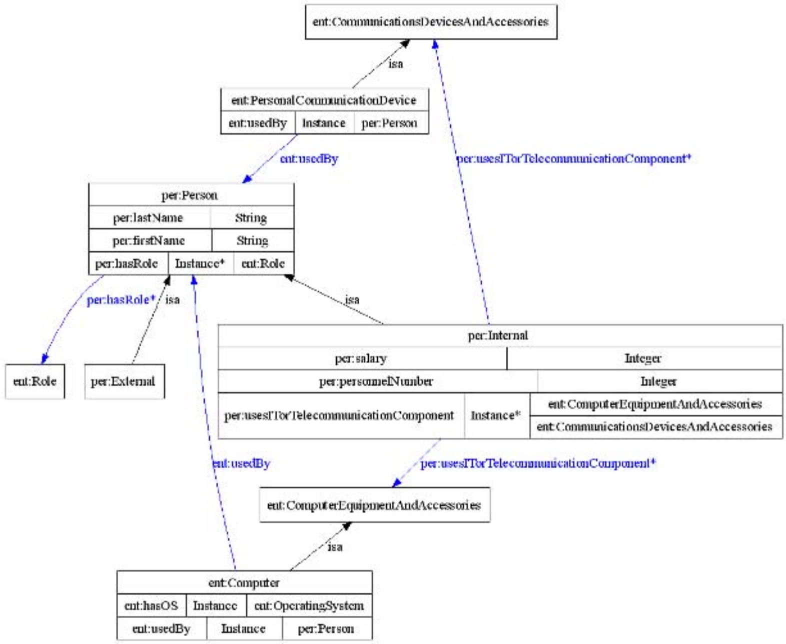

Fig. 5. Person ontology - concept relations

tools. Minor customization and extension can be done by company staff or external experts.

Filling the ontology is supported by user friendly and intuitive forms. The application itself, which accesses the ontology knowledge and executes the simulation to calculate risk management ratios, is provided by us. The next section presents example queries against the knowledge base.

2.2 Querying the Ontology

To take advantage of the developed ontology we must receive consistent answers as return on questions against the knowledge base. In the following we provide two easy examples, written in SPARQL (an RDF query language)[SPA06].

'What technical threats affect Integrity?', could be such a question from an administrator who wants to get to know unnoticed threats to check and update his security policy. The SPARQL representation is as follows:

```
SELECT ?threat
 WHERE { ?threatClass rdfs:subClassOf sec:TechnicalThreat .
     ?threat rdf:type ?threatClass .
     ?threat sec:affects <#Integrity> }
```

The query returns every instance of subclasses of *sec:TechnicalThreat* (see Subsection 2), e.g. 'Win32.Blackmal.F', 'Fire', etc. Now when the administrator knows possible threats he could ask for threatened infrastructure elements in his business, concerning the worm 'Win32.Blackmal.F'. The query can be defined like this:

```
SELECT ?threatenedAsset
 WHERE { <#Win32.Blackmal.F> sec:threatens ?threatenedAsset }
```

As result he gets a list of Windows-based computers in his company: 'Pc1, Pc2, FileServer, ...'. Obviously these are easy but already effective examples on how to use the developed ontology. Further queries could ask for possible countermeasures or the status of the anti-virus instances running on the PCs. In the following chapter we will introduce an advanced example on how to benefit on this approach.

3 Scenario

In this section we provide an example of how a company would use the aforementioned security ontology to model an IT infrastructure.

3.1 The Company

The company is an SME with six employees. Their main business is software sales and custom programming to modify their standard software. The company rents two floors (1st and 2nd floor) of a 5-floor building in the center of a small town. The following listing shows the allocation of relevant (IT) infrastructure elements:

– First floor - Office room (R0103): 2 PC's
– First floor - Storage room (R0104): data media (archived)
– Second floor - Server room (R0202): 4 Servers, 1 Router, 2 Switches, 1 Modem
– Second floor - Office room (R0203): 1 PC, 1 Notebook
– Second floor - Office room (R0204): 3 PC's

The infrastructure is mapped on the sub-tree *ent:Infrastructure* (compare Figure 4) and the following example gives an idea regarding the attributes of all *ent:Infrastructure* elements:

The concept *ent:PersonalComputer* with its concrete instance 'Pc4' has the attributes *ent:deliveryTime*, *ent:assetCost*, *ent:outageCost* and *ent:located*. If this or any other instance will be destroyed by a certain disaster, the ontology 'knows' how long it takes to get a new one, how much it costs, where it is located and the outage costs per day.

3.2 The Disaster

After describing the company with its infrastructure, the current subsection defines the disaster, which will hit our software company.

The event of fire, as a physical threat scenario, was chosen. The simulation should show the amount of damage in the course of time and in consideration of the fire source. A certain room can be defined as the fire source; the speed of propagation without any countermeasures will be 5 minutes per floor and 5 minutes per room. Every infrastructure element is assigned to a certain room. In the case of fire all infrastructure elements within a room will be destroyed completely. The outage costs per room correspond to the outage costs sum of all destroyed elements, which are located in the room. It is possible to assign countermeasures to any room. These safeguards can lower the probability of occurrence and the speed of propagation in the case of fire. The attribute *ent:damage* addresses the damage which results when the countermeasure is executed.

For instance: *ent:WaterFireExtinguisher* is located in room R0102 and will start, when switched on, immediately. Instance 'WaterFireExtinguisher0102' will extinguish the room within one minute. The attribute *ent:startTime* is important for countermeasures which are not activated automatically (e.g. hand fire extinguisher).

3.3 The Simulation

The framework for our threat analysis has been explained in the preceding sections, now we present a tool called *SecOntManger* which processes the ontology knowledge to simulate threats. This prototype handles IT costs and poses as proof of concept. Further threat effects as well as infrastructure components can be added easily due to the generic structure.

In our example the management wants to know what impact fire would have on the infrastructure, what countermeasures exist and what their benefits are. For this purpose we show two program runs, one against the unprotected company another including safeguards.

– The first program run without countermeasures: *SecOntManager* offers an intuitive graphical user interface, shown in Figure 6. A threat and a corresponding starting point have to be chosen before a simulation can be started. We decide for fire as threat and the server room (Room0202) as origin of fire. The program run produces a detailed log file which shows how the fire spreads from room to room and what damage it causes.

  Each room is processed completely before neighboring rooms are searched. At the end of the simulation all occurring costs are visualized in a line chart (see Figure 6). The time axis unit is set to minutes. Four curves, reflecting different cost categories, exist. The blue curve visualizes the damage: In the example the damage costs rise very fast due to the speed of fire - within 30 minutes every room was destroyed. By zooming in, displaying only the first 30 minutes, we can see how the damage evolves. After every room 'burned down' no further damage can occur. The red line represents the outage costs, taken from assigned outage costs of infrastructure components and employee's costs. Outage costs rise constantly in the simulation until

Fig. 6. SecOntManager: No countermeasures

recovery. The green curve shows recovery costs; replacement and setup times for destroyed components are taken into consideration. When components are available and paid connected outage costs decrease, visually spoken, the red line flattens. Additional installation costs lift the recovery costs upon damage. When every component is recovered, the pre-threat state is reached and outage costs do not rise anymore. Furthermore the total of all costs is reflected by the yellow curve. Fire costs 73605 Euros and it takes at minimum five days to recover the IT-infrastructure from the effects.

– Second program run with countermeasures enabled: We now concentrate on reducing the damage by installing safeguards. *SecOntManager* offers to install fire suppression systems in the building. We decide for pre-action pipes in the entire building. Necessary detectors and fire extinguishers are added to rooms in the OWL file. Their costs amount to 7200 Euros. Running the simulation produces the cost chart in Figure 7. As can be seen the total damage decreased drastically to 28329 Euros. After installing safeguards, the fire can not spread anymore; it is detected and extinguished shortly after breakout. Nevertheless costs and recovery times are still very high. The reason is that water extinguishers have a high damage factor concerning electronic devices and we have chosen the server room as place of fire origin. *SecOntManager* also offers CO2 fire extinguishers for locations with high electronically damages. Replacing the water extinguisher by a more expensive C02 extinguisher the total costs are reduced to 10934 Euros, which are mostly outage costs of

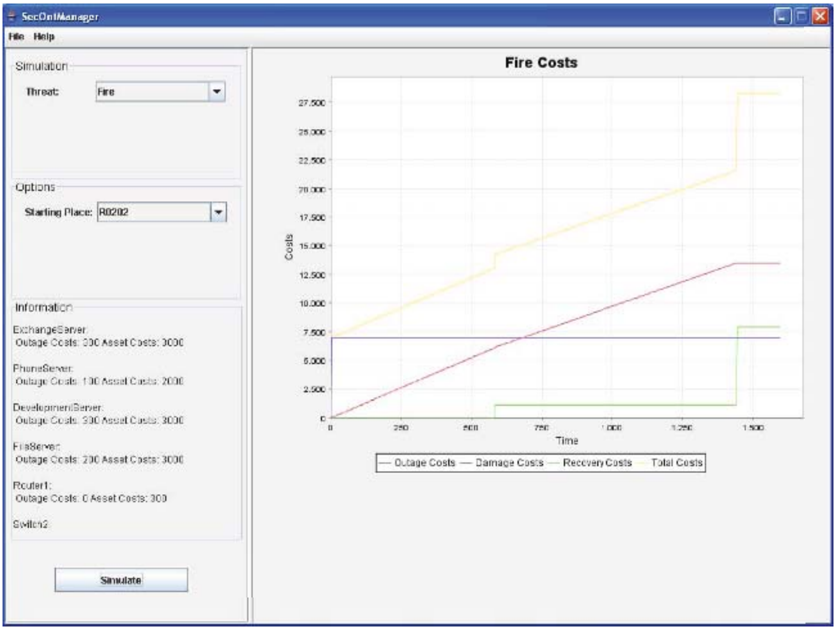

Fig. 7. SecOntManager: With countermeasures enabled

one server which caused the fire. By adding a redundant server the outage time and costs could be cut to zero.

4 Conclusion

We propose an ontology-based approach combining security- with business-domain knowledge to model companies. The ontology guarantees a shared and accurate terminology as well as portability. Knowledge of threats and corresponding countermeasures, derived from IT-security standards, is integrated into the ontology framework. Moreover, we implemented a prototype capable of simulating threats against the modeled company by processing the knowledge contained in the ontology. *SecOntManager* visualizes the damage caused by specific threats, outage costs and the recovery time. Running the program with added safeguards shows their benefits and offers objective data for decision making, which safeguards to implement and to avoid installing countermeasures that are not cost-effective. An enhanced prototype with advanced risk analysis and broader threat support will be developed in further research activities.

Acknowledgements

This work was performed at the Research Center Secure Business Austria funded by the Federal Ministry of Economics and Labor of the Republic of Austria (BMWA) and the federal province of Vienna.

References

[ALRL04] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell, and Carl E. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *IEEE Trans. Dependable Sec. Comput.*, 1(1):11–33, 2004.

[COB06] Cobit. http://www.isaca.org/, 2006.

[Don03] Marc Donner. Toward a security ontology. *IEEE Security and Privacy*, 1(3):6–7, May/June 2003.

[Hau00] Hans Eduard Hauser. Smes in germany, facts and figures 2000. Institut für Mittelstandsforschung, Bonn, 2000.

[ISO06] Iso17799. http://www.iso.org/, 2006.

[OWL04] Owl web ontology language. http://www.w3.org/TR/owl-features/, 2004.

[Pel01] T. R. Peltier. *Information Security Risk Analysis Boca Raton*. Auerbach Publications, Boca Raton, Florida, 2001.

[Pro05] The protege ontology editor and knowledge acquisition system. http://protege.stanford.edu/, 2005.

[SPA06] Sparql query language for rdf. http://www.w3.org/TR/rdf-sparql-query/, 2006.

Knowledge Work Productivity: Where to Start

Sebastian Eschenbach, Doris Riedl, and Bettina Schauer

Fachhochschul Studiengang Informationsberufe, Campus 1,
7000 Eisenstadt, Austria
{sebastian.eschenbach, doris.riedl,
bettina.schauer}@fh-burgenland.at
http://www.ib.fh-burgenland.at

Abstract. The paper proposes a new practical two step approach towards higher knowledge work productivity. Step one is based on a Knowledge Intensity Matrix adapted from Porter and Millar and allows for an assessment of the relevance of knowledge work for a particular company or strategic business unit. Step two identifies work effectiveness — rather than efficiency — as the decisive factor of knowledge work productivity and takes up Willke's systemic understanding of knowledge management to propose a process oriented analysis of knowledge work effectiveness, which results in suggestions of how to increase knowledge work productivity.

Keywords: Knowledge Work, Productivity, Knowledge Intensity.

1 Introduction

> "The most important contribution, and indeed the truly unique, contribution of management in the 20th century was the fifty fold increase in the productivity of the manual worker in manufacturing.
>
> The most important contribution management needs to make in the 21st century is similarly to increase the productivity of knowledge work and the knowledge worker.
>
> The most valuable assets of a 20th-century company were its production equipment. The most valuable asset of a 21st-century institution, whether business or nonbusiness, will be its knowledge workers and their productivity" [5 p 135].

In this paper we present a new method to identify potential for higher knowledge work productivity. This *Potential Analysis* was developed by a team of researchers and practitioners (see acknowledgements) at the University of Applied Sciences Burgenland (Fachhochschul-Studiengänge Burgenland) in Eisenstadt/Austria during the past three years and applied for the first time to a leading financial service provider during the academic year 2005/06. We present an overview of our theoretical basis, explain the method, report on the application and discuss options to further enhance the method.

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 49–60, 2006.

2 Knowledge Work Productivity

The term knowledge work was first used by Drucker thirty five years ago for people who deal with formal knowledge as a major part of their job [5]. Formal or organizational knowledge is used here in contrast to personal or tacit knowledge. The term "productivity" is generally used either in economics to describe input/output relations in the context of production or in engineering to describe production processes. When we use the term "knowledge work productivity", we simply refer to the results of knowledge work. Therefore higher knowledge work productivity just means higher levels of results.

While in economics labor productivity is frequently reduced to efficiency in the sense of an economic utilization of inputs[1], "[t]here is a consensus in the literature that in knowledge work, quality is one of the more important factors" [21 p 614]. The efficiency, for instance, of an executive might be measured by the number of decisions she takes per hour, the efficiency of a doctor by the number of patients he sees, the efficiency of a bank officer by the number of credit applications she processes. These three examples should be quite sufficient to demonstrate that data on efficiency by itself do not allow *any* reasonable conclusions about the productivity of knowledge work to be drawn. Quality needs to be taken into account too.

However, to the best of our knowledge, standardized approaches to assess the quality of the results of knowledge work are not available except for very special fields such as software engineering [2], [21]. Our Potential Analysis therefore concentrates on process quality instead. It tries to assess how *effectively* knowledge workers handle knowledge — their central resource — in the context of characteristic activities.

Work efficiency and the effective handling of knowledge seem to be two decisive factors of knowledge work productivity (which does not rule out other factors). If one of these two factors is reduced to zero — no results at all or totally unsystematic handling of knowledge — the productivity of knowledge work as a whole is destroyed. The following formula can be used to symbolize this important correlation:

Productivity of knowledge work
= work efficiency x effective handling of organizational knowledge ...[2]

Therefore meaningful attempts to improve knowledge work productivity, first of all call for quality standards on how to handle knowledge effectively. Without such standards, knowledge work productivity will continue to be an issue of chance rather than

---

[1] See e.g. the OECD Macrothesaurus, which lists "efficiency of labor" as a synonym for "labor productivity", http://www.uibk.ac.at/physics/info/oecd-macroth/en/1729.html, accessed 14.7.2006.

[2] A Google search showed that the software engineering expert Milton Bryce seems to have already used a similar formula „productivity = effectiveness x efficiency" in the early 1970s in the context of software projects. He too used the formula to point to the importance of effectiveness for the work productivity (http://www.phmainstreet.com/mba/ss050502.pdf, accessed 10.7.2006). It is also important to note that our formula refers to knowledge work and not knowledge work*er* productivity. To broaden the definition, other factors would need to be included. Drucker's six factors of knowledge worker productivity might serve as a valuable starting point [5].

of systematic management. Optimizing work efficiency might come as a second step, but is not part of the Potential Analysis presented here.

3 State of the Art

Approaches to measure and hence increase the productivity of knowledge work range from methods aiming at quantifying the intellectual capital of an organization to models stemming from ergonomics and process-oriented models — for an overview refer to [10]. The former include Intellectual Capital Reports, for example the Skandia Navigator [9], Sveiby's Intellectual Assets Monitor [23] and the well-known Balanced Score Card [12] with a focus on measuring the value of intellectual capital. These models thus provide a highly aggregated view on a company's intellectual assets. Neither an analysis of the efficiency nor of the effectiveness of knowledge work is presented by these approaches.

Ergonomic models, on the other hand, evaluate the performance of an individual knowledge worker, also with respect to the work environment. Higher productivity is mentioned as a desired goal but most methods do not provide precise means to capture the effectiveness of knowledge work. Ergonomic models comprise VERA-G, a method to identify potentials for rationalization with respect to different categories of knowledge work by Resch [22], TBS-GA(A), a task rating system of knowledge work by Hacker [11], Participative Productivity Management (PPM) introduced by Pritchard [20], the rating of performance types by Pfiffner and Stadelmann [18] and a classification of alternative knowledge work types by Davenport et al. [4].

The drawback of the approaches mentioned above is that they do not take into account the process oriented aspects of knowledge work. Process oriented models include OPUS developed by Matt, which is a method based on system theory to increase the productivity of indirect domains providing an efficiency as well as an effectiveness analysis of the domain under consideration [15]. Michaelis adopts the classical productivity model with an input output system for the creation of intellectual assets [16]. The emphasis is on the measurement of input and output but the model lacks a detailed analysis of the structure and the process of knowledge work.

Most questionnaires which are designed to evaluate the productivity of knowledge work question the interviewees directly about their opinion concerning knowledge management and the importance thereof, efficiency of knowledge transfer and the necessity for learning and professional training with respect to their work domains. In contrast, the method presented in this paper uses an indirect, process-oriented approach based on guided interviews in order to avoid socially expected answers.

4 Theoretical Basis

The approach presented in this paper draws upon the definition of the productivity of knowledge work discussed above, a strategic theory to determine the knowledge intensity of a strategic business unit combined with a systemic perspective on knowledge management.

4.1 Knowledge Intensity

Contrary to a frequently voiced belief among knowledge management experts, our approach is based on the explicit assumption that knowledge and knowledge management are of significantly different importance to different types of business. In fact even in well advanced knowledge economies there are many organizations for which knowledge is not and never will be the central resource.

We therefore start our Potential Analysis with an assessment of the relevance of knowledge for a particular organization or business unit. To do so, we re-utilize the Information Intensity Matrix developed by Porter and Millar [19], which despite its name actually shows the importance of knowledge to an organization's major activities, final products, and services [26] [see Fig. 1]. In accordance with the often used definitions of information and knowledge by Davenport and Prusak [3], we have renamed the matrix consequently "Knowledge Intensity Matrix".

Both factors — the knowledge intensity of processes and the knowledge intensity of products and services — are entered into a simple two by two matrix which should help to communicate if and why knowledge management should be high on the agenda of a specific organization.

Organizations have to face new challenges emerging in a knowledge economy and traditional management approaches are not capable of meeting the challenges of this kind of economy [26]. However this does not mean that knowledge management is per se important for every organization but only when sophisticated, knowledge-based products and services are critical success factors [26]. In fact, following Krey, we advise the selective employment of knowledge management with a specific focus in mind [13].

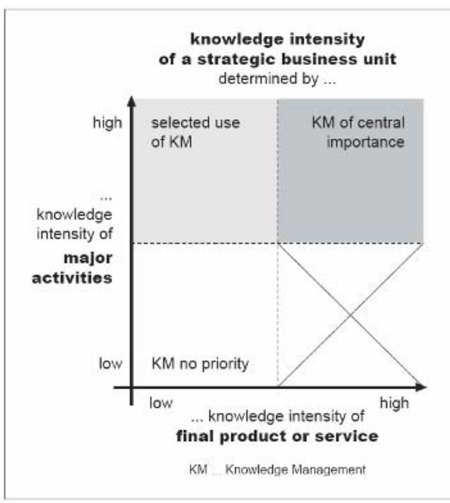

Fig. 1. Knowledge Intensity Matrix following Porter and Millar [19]

4.2 Standards for Effective Knowledge Work

Based on the assumption that not only individuals but social systems as a whole are able to learn, see [1], it follows that they are also capable of generating and storing knowledge as well as utilizing it within systemic processes [26]. According to Willke, organizational knowledge is characterised by the way a specific social system operates, which itself is defined by the context of rules made anonymous, independent of individuals [26]. This implies that in order to stay competitive in a knowledge economy it no longer suffices to equip the employees with the required expertise on an individual level, but also the organization has to integrate its organizational knowledge into the business processes.

In order to use knowledge in a systematic and effective way, an established system of standards and rules for the incorporation, storage, utilization and development of organizational knowledge is required. Such standards comprise standing operating procedures, guidelines, codifications, descriptions of operating processes, routines, traditions, databases, codified knowledge with respect to production and projects as well as the culture of an organization [26]. Thus, an organization's objective knowledge is a supra-individual structural entity of standards and rules.

Knowledge work comprises communication, transactions, interactions and decision making that require the continuous revising and improving of the resource knowledge. The productivity of knowledge work depends on an organization's ability to install a system of rules and an ICT as well as an organizational infrastructure for the exchange of information and knowledge transfer - including the enhancements and the experienced utilization thereof [26]. Hence, in order to find out whether an organization does effectively make use of its knowledge, the crucial criteria are the existence of and compliance with the standards described above.

5 A New Method to Identify Potential for Higher Knowledge Work Productivity

We start our Potential Analysis of knowledge work productivity with an assessment of the relevance of knowledge work and therefore knowledge work productivity for a particular organization or business unit [Fig. 2, Step (1)]. This assessment is based on how knowledge intensive a business unit's major activities and its most important products appear to be (the importance of products being measured e.g. in terms of current turnover, contribution or growth potential). To determine the knowledge intensity, we conduct a small number of expert interviews with the middle managers in charge of specific products or processes. As described above, the knowledge intensity of a product or process is measured by

- the time and effort that is needed for coordination (complexity),
- the number of options that characterize the respective product or process (complexity),
- the rate at which processes or products need to be adapted in order to keep a business competitive (innovation rate), and
- the necessary standard of formal education and training of the employees involved.

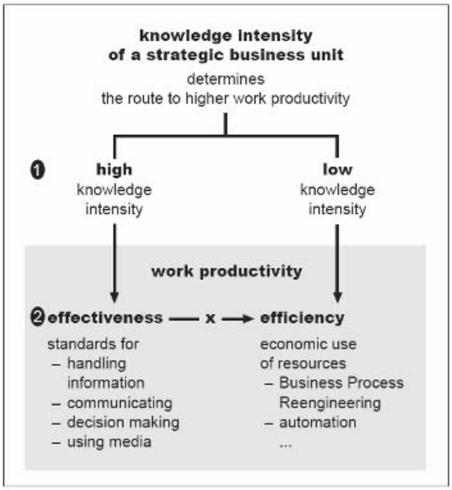

Fig. 2. Potential Analysis for knowledge work productivity

The higher these factors are, the higher is the knowledge intensity and thus the need for knowledge management.

The information gained from interviewing the relevant middle managers is entered into the Knowledge Intensity Matrix described above [see Fig. 1]. The matrix provides an analytical and transparent basis for discussing an organization's approach towards knowledge management. It might be of help in overcoming top management's prejudices' based approach towards knowledge management, which today still seems to prevail either as euphoria or total disapproval, see e.g. Malik [14].

According to our approach, the level of knowledge intensity determines the route to higher knowledge work productivity, which we define as the product of the two factors: (1) work efficiency and (2) effective handling of knowledge. In cases of relatively low knowledge intensity (e.g. a shop floor production of bolts), quality standards can be determined rather easily. Therefore one can focus directly on work efficiency, by making use of well established techniques like Business Process Reengineering or automation. But the higher the levels of knowledge intensity become (e.g. an investment bank, a hospital, a school, or even a research lab), the more the effectiveness of handling relevant knowledge becomes the decisive factor of knowledge work productivity as a whole (e.g. the productivity of a scientist is determined by the impact of her work and only marginally by the number of papers she produces during a specific year).

Therefore only after a particular business unit turns out to be characterized by high levels of knowledge intensity, does it make sense to move on to the second step of the Potential Analysis [Fig. 2, Step (2)]. This second step consists of an in depth analysis of how systematically knowledge is being handled in the context of selected business activities. Ideally this analysis should cover both primary activities involving an organization's external partners (e.g. a sales or purchasing activity) and internal coordinating activities (e.g. financial planning, conducting meetings, or acquainting employees with new tasks). Here our approach differs again significantly from other practices which aim at cross-organizational benchmarking and therefore need to focus on comparable activities. We, however, focus on activities which for one reason or another are of specific importance to the organization in question.

To be able to actually analyze knowledge work effectiveness in the context of a specific activity, we first draw up detailed process descriptions. These descriptions are then used to run a series of standardized interviews inquiring in minute steps whether employees can draw and actually do draw on adequate, explicit standards for handling knowledge during their every day work. These rules, regulations, checklists, routines, laws, plans etc. represent an organization's knowledge.

For this type of analysis, we have learned to differentiate between four types of subtasks:

(1) Systematic processing of information with adequate ICT support,
(2) Systematic communication with adequate ICT support,
(3) Systematic decision making,
(4) Systematic adaptation of rules and standards to changing circumstances.

This categorization allows us to distinguish between (a) a lack of effectiveness in connection with a very specific sub-activity — "a local weakness" so to say (e.g. an employee in a certain subsidiary is not yet familiar with a new guideline) and (b) a more general lack of effectiveness which might turn out to be somewhat typical for an organization's general approach towards handling knowledge (e.g. generally too unclear responsibilities in the context of decision making which might result in ineffective, widespread up-delegations).

In three empirical surveys on knowledge work productivity in Austria's medium size businesses which were also based on the theoretical assumptions presented here, we found that while the majority of companies seems to be quite good at handling bare information in a systematic way, many are not yet capable of organizing the more demanding activities like communication, decision making, or organizational learning in equally systematic ways [7],[8],[17].

6 Application for an International Financial Service Provider

After a number of empirical studies, the first major application of our Potential Analysis to a specific company was a project with a leading financial service provider's unit for international commercial activities which we conducted between October 2005 and June 2006.

6.1 Execution of the Potential Analysis

The first objective of this project was to establish if, why, and in connection with what type of activities knowledge management might be of relevance to the business unit in question. The second objective was to determine how effectively knowledge work was currently organized in the context of certain business processes.

The two processes selected for the Potential Analysis were

(1) a primary customer oriented business activity in the context of the approval processes for a specific type of commercial loans, and
(2) an internal support activity in the context of strategic budgeting for commercial key accounts.

1.b For how many of these new products does your organisation have to enhance the existing knowledge or develop new knowledge (i.e. check up, gather new information etc.)?

For every new solution we have to learn and enhance our knowledge.	
For most of the new solutions we have to learn and enhance our knowledge.	
In most cases we manage with our existing knowledge.	
We always manage with our existing knowledge (We combine it in new ways).	
Don´t know	

Indicator: Alternatives of the products

☐ OK ☐ Question not understood ☐ Question not relevant ☐ Answer does not fit into scheme

Fig. 3. Example of one of the questions for determining the knowledge intensity of a product (translated into English)

For the first step of the Potential Analysis two standardized interviews with the relevant process/product managers were conducted. Each interview consisted of 21 standardized questions [for an example see Fig. 3] covering complexity, innovation rate, and levels of employee education.

The analysis of the interviews showed that many aspects of the primary process are already highly standardized and thus of surprisingly little complexity. However, a high level of consulting with customers was required, which turned the customer interface of the activity (i.e. communication with the customer) into an area of high knowledge intensity. The internal support process which is about collecting, processing, analysing, and redistribution of information about international commercial customers is characterised by a high level of variability and needs a lot of coordination. Hence it was classified as a knowledge intensive activity. So both activities call for the employment of knowledge management, especially as far as the communication with customers and information about customers are concerned.

Moving on to Step two [see Fig. 2], we analysed a sample of six actual business transactions:

— three real cases of loan approvals to specific business customers and
— three real cases of strategic budgeting specific accounts.

The analysis was based on interviews with employees who were all involved with one of the two transactions as part of their every day work routines. All together we conducted eleven interviews, instead of just one for each activity, because, on the one hand, the knowledge intensive transactions we studied involved most of the time more than one employee, and on the other hand, because we tried to cover each transaction at least twice to check for personal biases of the interviewees. It also seems important to note that all interviews focused strictly on one particular transaction each (e.g. the approval of a loan to company X, two weeks prior to the interview). Basing the interviews on real events helped to get accurate descriptions and to avoid personal biases and false generalizations. The interviews were conducted over a period of two months and involved employees of three different national subsidiaries of the financial service provider. The working language was German.

During the interviews the transactions under examination were reconstructed in minute details to find out whether employees knew about and observed rules, standards, guidelines etc. about the handling of information (e.g. regulation on how a specific piece of information should be filed), communication, decision making, adaptation of business procedures, and the employment of ICT. So all together we asked 180 sets of questions [for an example see Fig 4.].

1-5.b Is there a rule in written form about how, where, when and by whom the request for an offer is filed and saved?

There are explicit written rules about how, where, when and by whom the request for an offer is filed and saved.	
There are indistinct informal rules about how, where, when and by whom the request for an offer is filed and saved.	
There are no rules – the filing and saving of the request for an offer occur as the case arises.	
Don´t know	

Indicator: Existence of rules

☐ OK ☐ Question not understood ☐ Question not relevant ☐ Answer does not fit into scheme

Fig. 4. Example of one of the questions for determining the existence of rules for a specific activity (translated into English)

Together the results from step one and step two point at two main potentials for more systematic and, according to our definition at the beginning of this paper, therefore more productive knowledge work.

6.2 Potential One: Knowledge Exchange About/with Customers

Despite the specifically high knowledge intensity of the customer interface (step one), the analysis of the interviews in step two of the Potential Analysis indicated precisely in this respect a remarkable weakness (i.e. ineffective or unsystematic handling of knowledge). It seems that many of the account managers who work with customers up front on a daily basis are simply not aware of some of the guidelines (e.g. for customer communication) — a fact that came as quite a surprise to some of the headquarter managers in charge of these processes.

So both the handling of knowledge about customers and the exchange of knowledge with customers leave plenty of room to further improve knowledge work productivity. This seems to be of special importance if a financial service provider tries to shift its focus from information transaction to far more knowledge intensive customer consulting activities.

6.3 Potential Two: Organizational Learning

The largest potential for a more systematic approach towards knowledge work was identified with respect to the way the company adapts its own organizational knowledge, represented as guidelines, rules, standards, regulation, plans etc. While up to this point we have been analyzing how systematically an organization employs its knowledge, the issue here is how systematically the company learns (i.e. creates and implements new knowledge).

In reaction to these findings, some employees of the client company questioned whether in times of rapid change management should rely on rules and other types of regulations any longer. To answer this question one needs to see what choices organizations in a fast changing knowledge economy actually have. It is certainly not the choice between a predictable and static or a changing environment, but between systematic adaptation of the organization's knowledge to new circumstances on the one hand, and arbitrary reaction on the other hand. In fact, the more knowledge intensive an organization is and the faster the environment changes, the higher is the call for clearly managed organizational learning on the basis of explicit knowledge.

7 Conclusions and Further Research

Generally speaking, knowledge work productivity is a decisive factor of economic growth and competitiveness [5]. However, this does by no means make knowledge work productivity the decisive success factor for every single organization. A sensible approach towards knowledge management should therefore start with a careful assessment of the knowledge intensity of an organization's major activities, products, and services. Without this first step, knowledge management runs the risk of being seen as little more than just another management fad.

Contrary to manual work, the productivity of knowledge work seems to be, first of all, a question of the effective handling of its most important resource rather than of efficiency. Starting with Willke's system theory based perspective on knowledge management [26], we proposed a Potential Analysis for knowledge work productivity which allows for an in depth analysis of how systematic knowledge work is managed in a specific organization.

The first application of this Potential Analysis — in the framework of an applied research project — seems to have yielded useful results. The client, a leading financial service provider's unit for international commercial activities, has gained a clear understanding of effective knowledge management geared to the specifics of its products and business activities and aimed at higher knowledge work productivity.

During this first application of our approach, we also gained some valuable insights into how to further improve our Potential Analysis: (1) While this time we just asked the client to name two of its business activities to be analysed, a more systematic selection seems to be advisable. (2) It also seems to be worthwhile to try to extend the analysis beyond an organization's boundaries including some interviews with customers on their perspective of the products' and services' knowledge intensity. (3) The structure of the interview questions in step two could also be complemented. While at the moment the interviewees are only asked (a) whether there are guidelines on how to handle information and knowledge in the context of a certain sub-activity and (b) whether the guidelines have essentially been obeyed in respect to a specific transaction, two more questions could be added. A third question could ask (c) whether a specific guideline — representing the organization's knowledge in respect to a specific business situation — actually led to an appropriate outcome. And finally, instead of just asking about guidelines in a relatively general way, the interviewee could be asked to name the specific regulation she or he applied in a particular situation.

Acknowledgements

This paper is the first result of a project funded by the FHplus Programme of the FFG. We would also like to thank Manfred della Schiava as well as Karl Enghauser for many important contributions and fruitful discussions and Veronica Dal-Bianco for proof reading.

References

1. Argyris, C., Schön, D.: Organizational Learning II. Theory, Method and Practice. Addison Wesley, Reading (1996)
2. Bok, H.S., Raman, K.S.: Software Engineering Productivity Measurement Using Functional Points: A Case Study. Journal of Information Technology, Vol. 15 1 (2000) 19-101.
3. Davenport, T.H, Prusak, L.: Working Knowledge — How Organizations Manage What They Know. 2nd edn. Harvard Business Scholl Press, Boston (2000)
4. Davenport, T.H., Thomas, R.J., Cantrell, S.: The Mysterious Art and Science of Knowledge-Worker Performance. MIT Sloan Management Review, Vol. 44 1 (2002) 23-30
5. Drucker, P.F.: Management for the 21st Century. HarperCollins, New York, 1999
6. Drucker, P.F.: The Age of Discontinuity — Guidelines to Our Changing Society. Harper & Row, New York (1969)
7. Ecker, K., et al.: Mittelbetriebe in der Wissensökonomie. Workingpaper des Fachhochschul-Studiengangs Informationsberufe, Nr. 4 (2004)
8. Ecker, K.: Antriebskräfte für und Hürden zum Wissensmanagement in Mittelbetrieben. Diplomarbeit am Fachhochschulstudiengang Informationsberufe, Eisenstadt, (2005)
9. Edvinsson, L., Malone, M.S.: Intellectual Capital: Realizing Your Company's True Value by Finding its Hidden Brainpower. HarperBusiness, New York (1997) http://www.fh-burgenland.at/Eisenstadt/IBBakk_AWMMag_Spez/v3/m3/index_fue.html, Accessed: 23.03.2006
10. Eschenbach, S., Geyer, B., et al.: Wissen & Management – 12 Konzepte für den Umgang mit Wissen in komplexen Organisationen. Linde, Wien (2004)
11. Hacker, W.: Tätigkeitsbewertungssystem (TBS). Verfahren zur Analyse und Gestaltung von Arbeitstätigkeiten. vdf, Zürich (1995)
12. Kaplan, R.S., Norton, D.P.: The Balanced Score Card: Translating Strategy into Action. Harvard Business School Press, Boston (1996)
13. Krey, G.: Wissensmanagement im Mittelstand – Wo steckt der Nutzen?. In: Bellmann, M., Krcmar H., Sommerlatte T. (eds.): Praxishandbuch Wissensmanagement. Strategien – Methoden – Fallbeispiele. Symposion, Düsseldorf (2002) 338-363
14. Malik, F.: Wissensmanagement – Auch dieser Kaiser ist nackt. Manager Magazin, 27.11.2001, http://www.managermagazin.de/koepfe/mzsg/0,2828,169723,00.html, Accessed: 23.03.2006
15. Matt, D.: OPUS Objektorientierte Prozess- und Strukturinnovation – Methode und Leitfaden zur Steigerung der Produktivität indirekter Leistungsbereiche. PhD Thesis, University of Karlsruhe, Karlsruhe (1998)

16. Michaelis, U.: Produktivitätsbestimmung in indirekten Bereichen. Springer Verlag, Berlin (1998)
17. Morawitz, J.: Umgang mit Wissen und Information in burgenländischen Mittelbetrieben. Diplomarbeit am Fachhochschulstudiengang Informationsberufe, Eisenstadt, (2006)
18. Pfiffner, M., Stadelmann, P.D.: Wissen wirksam machen. Wie Kopfarbeiter produktiv werden. Haupt, Wien (1998)
19. Porter, M., Millar, V.: How Information Gives You Competitive Advantage. Harvard Business Review, Vol. 63 4 (1985) 149-160
20. Pritchard, R.: Measuring Organizational Productivity. In: Dreuth, P., Seargeant, J., Takens, R. (eds.): European Perspectives in Psychology. John Wiley & Sons, Chichester Vol. 3 (1996) 79-87
21. Ramírez, Y., Nembhard, D.: Measuring Knowledge Worker Productivity — A Taxonomy. Journal of Intellectual Capital. Vol 5 4 (2004) 602-628.
22. Resch, M.: Die Handlungsregulation geistiger Arbeit. Bestimmung und Analyse geistiger Arbeitstätigkeiten in der industriellen Produktion. Huber, Bern (1988)
23. Sveiby, K.E.: The New Organizational Wealth: Managing and Measuring Knowledge Based Assets. Berret-Koehler, San Francisco (1997)
24. Strohmaier, M.: B-KIDE: A Framework and a Tool for Business Process Oriented Knowledge Infrastructure Development. PhD Thesis, University of Technology, Graz (2004)
25. Timbrell, G., Koller, S., Lindstaedt, S.N.: Improving Service Innovation through Structured Process-oriented Knowledge Infrastructure Design. Proc. I-KNOW`05 (5th International Conference on Knowledge Management), Springer Verlag, Graz (2005) 373-380
26. Willke, H.: Systemisches Wissensmanagement. 2nd edn. Lucius & Lucius, Stuttgart (2001)

Taba Workstation: Supporting Technical Solution Through Knowledge Management of Design Rationale

Sávio Figueiredo[1], Gleison Santos[1], Mariano Montoni[1], Ana Regina Rocha[1], Andréa Barreto[1], Ahilton Barreto[1], and Analia Ferreira [1,2]

[1] Federal University of Rio de Janeiro - COPPE Sistemas
Caixa Postal 68511 – CEP 21941-972– Rio de Janeiro, Brazil
Phone: +55-21-25628675; Fax: +55-21-25628676
{savio, gleison, mmontoni, darocha, ansoares, ahilton, analia}@cos.ufrj.br

[2] BL Informática Ltda
Av Visconde do Rio Branco 305 – 8º andar – Centro – Niterói CEP 24020-002– RJ, Brazil
Phone: +55-21-21132300; Fax: +55-21-2620-3644
analia@blnet.com

Abstract. The development and maintenance of software products are knowledge intensive tasks. Due to the high turnover of software industry, there is a great probability that the original designers are unlikely to be available when problems arise and modifications are needed. Therefore, having the reasons behind the decisions recorded can be invaluable as people leave and join the software team, because the knowledge about choices of those leaving would still be available to the newcomers. This work describes an approach to support the Technical Solution process through knowledge management in the context of a Process-centered Software Engineering Environment (PSEE) named TABA Workstation. It also presents some information related to a software process improvement initiative undertaken in a Brazilian organization that demonstrates the feasibility of the presented approach.

Keywords: technical solution, design rationale, knowledge management.

1 Introduction

The Technical Solution is a knowledge intensive process that aims to design, develop, and implement solutions to the requirements developed [3]. During its execution several choices have to be made, since a problem can be solved in many different ways. Therefore, it is important to preserve the organizational knowledge acquired during the process, recording the rationale which explains why a specific decision was made during design process. Moreover, the combination of a long life cycle and the typically high turnover in the software industry increases the probability that original designers will not be available to be consulted when problems arise and modifications in the design are necessary [2].

Another common problem that organizations have to undertake is the difficulty to make the newcomers take advantage of more experience personnel knowledge during

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 61 – 72, 2006.

training, since the dynamic environment of most software organizations does not allow more experienced people to spend time sharing their knowledge with newcomers. Thus, in decision making situations, newcomers tend to repeat the same mistakes made by other members of the team that have already experienced a similar situation [15].

Furthermore, very often the kind of knowledge necessary for maintenance involves the understanding of what design choices were considered, which alternative solutions were examined and rejected and which criteria and requirements were addressed under the deliberation process [4].

Design Rationale differs from other design documentation because it captures the reasoning behind design decisions, and not just a snapshot of a final decision. Design Rationale offers more: not only the decisions, but also the reasons behind each one of them, including its justification, other alternatives considered, and argumentation leading to the decision [2].

The Taba Workstation is an enterprise-oriented Process-centered Software Engineering Environment (PSEE); constituted of integrated tools to support software processes definition, deployment and enactment. It also integrates knowledge management activities within software processes aiming to preserve organizational knowledge, and to foster the institutionalization of a learning software organization. The Taba Workstation has been developed in the context of an academic project and it is not commercialized. Nevertheless, it is granted to small and medium size organizations of Brazil with no costs [17].

In order to support the capture, retrieval and use of this kind of knowledge called design rationale, a tool named TechSolution has been implemented in the context of Taba Workstation. Regarding knowledge management, this tool allows project members to consult the design decisions made during the project and design decisions made in past projects. Besides, this tool provides knowledge for a certain kind of problem: the choice of the architectural style. TechSolution is also integrated with a knowledge management tool named Acknowledge, which aims to acquire, filter and package organization members' tacit and explicit knowledge [17].

This paper presents the technical solution support based in knowledge through the capture, storage and retrieval of design rationale offered by the Taba Workstation. Next section presents some basic concepts regarding design rationale. The section 3 discusses knowledge management in the context of software organizations. The Taba Workstation infrastructure is presented in section 4. The section 5 discusses the approach adopted in Taba Workstation to support design rationale. Section 6 presents some information related to a software process improvement initiative undertaken in a Brazilian organization that demonstrates the feasibility of the presented approach. Section 7 presents some conclusions and points out future work.

2 Design Rationale

One indicator of a good design process is that the design was chosen after comparing and evaluating it against alternative solutions [3]. Design rationale (DR) is information that explains why an artifact, or piece of it, is designed the way it is [12]. Souza et al., [18] define DR as the design decisions documentation with its respective justifications, options considered, evaluations and the argumentations that lead to the decision made.

DR is particularly useful for maintenance, having in mind that very often, the type of knowledge necessary for maintenance depends on the understanding of what design decisions were considered, which assumptions were made, which alternative solutions were rejected, and which criteria and requirements were addressed by the deliberation process [2].

Moreover, the combination of a long life cycle and the typically high turnover in the software industry increases the probability that original designers will not be available for consultation when problems arise and modifications are necessary [2].

The human memory limitation motivates the DR storage because this storage enables that, after a period of time, information about decisions made, why they were made and alternatives investigated, can be remembered, avoiding that mistakes made in the past be performed again and the human memory limitation doesn't be the main reason [8].

[2] points out the main benefits and possible uses of DR: (i) *Design Verification*: rationale can help to verify that the design meets requirements and designer's intent; (ii) *Design Evaluation*: rationale can help to evaluate design and design choices relative to one another to detect nonconformities that may affect quality; (iii) *Design Maintenance*: rationale can help to locate sources of design problems, to indicate where changes need to be made in order to modify the design and to ensure that rejected options are not inadvertently re-implemented; (iv) *Design Reuse*: rationale can help to specify which portions of the design can be reused and, in some cases, suggest where and how it should be modified to meet a new set of requirements; (v) *Design Teaching*: rationale can help to teach new personnel about the design; (vi) *Design Communication*: rationale can help to communicate the reasons for decisions to other members of the design team; (vii) *Design Assistance*: rationale can help to clarify discussion, check impact of design modifications, perform consistency checking and assist in conflict mitigation by looking for constraint violations between multiple designers; and (viii) *Design Documentation*: rationale can help to document the design by offering a picture of the history of the design and reasons for the design choices as well as a view of the final product.

Despite all these benefits described, DR is not in widespread use [2]. The main obstacles that explain the designer's resistance to use DR are: (i) there is a great likelihood that people who spend time recording the DR probably will not be those who will take advantages from it; (ii) documenting decisions can impede the design process if decision recording is viewed as a separated process from constructing the artifact; (iii) there is a risk that the overhead of capturing the rationale may impact the project schedule enough to make the difference between a project that meets its deadlines and is completed, versus one where the failure to meet deadlines results in cancellation; and (iv) difficulties faced by developers to retrieve the DR.

Like lessons learned, best practices, software processes models and so on, DR is a kind of knowledge that learning software organizations should manage in order to preserve knowledge inside the organization.

3 Knowledge Management in Software Organizations

Software engineering is a wide knowledge area made up of various sub areas [6]. Besides, the activity of software development requires computing knowledge, knowledge of the intended application domain as well as knowledge of the application itself. Another aspect to be taken into account is the experimental, evolutionary and non-repetitive characteristics of the software engineering area [13], which means that there are approaches that work best in certain situations and it is necessary to tailor them in order to deal with new situations. Moreover, unforeseen events may occur despite careful software project planning. This implies making constant choices from among the many feasible options throughout the software life cycle [16]. As a result, many software companies have recognized the importance of administrating knowledge effectively, productively and creatively at both the individual and organizational levels [11].

The identification, maintenance and dissemination of different types of knowledge related to software processes (e.g. software processes models, best practices and lessons learned) from one project to another are important to develop software with high quality and enhance software processes [9]. Software processes models, for instance, explicitly represent knowledge about software development activities, but also the software products, necessary resources and tools, and best practices related to software processes execution [10]. Therefore, efficient management of such knowledge supports organizational learning and initiatives for software process measurement and improvement [14].

The fact that most software development organizations are process-centered provides many benefits (e.g. process-centered knowledge management systems can be designed to explicitly associate software process activities with knowledge necessary to execute it) [14]. Moreover, tacit and explicit member's knowledge related to software processes are valuable individual assets that must be captured and converted into the organizational level. The collected knowledge represents indicators of problems concerning the software process definition or the environment in which the software is being developed. This important knowledge can be used to learn about the software process and to provide the means for implementing organizational changes aimed to enhance business performance [5]. In order to acquire such knowledge efficiently, it is necessary to transform arbitrary experiences declarations in structured explicit representations through the execution of activities for knowledge acquisition, packaging, dissemination and utilization [1].

4 The TABA Workstation

The Taba Workstation has been developed in the context of an academic project and it is not commercialized. Nevertheless, it is granted to small and medium size organizations of Brazil with no costs. During the last years, the Taba Workstation evolved to comply with the software organizations capability maturity models different levels. It is constituted of integrated tools to support software processes definition, deployment and enactment. The functionalities of other tools to support Knowledge Management activities are integrated into the environment to facilitate the organizational knowledge preservation and support activities execution [17].

The Taba Workstation evolved during the last years to support knowledge management activities integrated to the software processes aiming to preserve organizational knowledge and foster the institutionalization of a learning software organization. Therefore, the main objectives of Taba Workstation are: (i) to support the configuration of process-centered software development environments for different organizations (Configured SDE); (ii) to support the automatic generation (i.e., instantiation) of software development environments for specific projects (Enterprise-Oriented SDE); (iii) to support software development using the instantiated environment; and (iv) to support the management of organizational knowledge related to software processes [17].

The CASE tools integrated in the environments offer automated support to: (i) definition of the organizational set of standard processes; (ii) execution of pilot project aiming process improvement; (iii) adaptation of the organization standard processes for a specific project; (iv) definition of the organizational structure; (v) acquisition, filtering, packaging and dissemination of organizational knowledge; (vi) planning the organization of specific projects; (vii) time, costs, risks, human resources planning, monitoring and control; (viii) planning and execution of Configuration Management activities; (ix) identification of software product quality requirements; (x) documentation planning; (xi) supporting the planning and monitoring of corrective actions; (xii) supporting measurement and analysis activities based on the GQM method; (xiii) project monitoring through the generation of periodic reports and measures; (xiv) controlling of the activities executed during a specific project; (xv) requirements management; (xvi) supporting software technical solutions through the use of design rationale; (xv) supporting software verification and validation planning and execution; and (xvi) post mortem analysis [17].

The next section presents the proposed approach, describing the decision making process defined ant the tool implemented in order to support the process deployment.

5 Technical Solution Supporting Through Design Rationale

In order to assist the Technical Solution process in software projects, a process aiming to aid design decision making was defined. Through the deployment of this process, the organization can preserve the knowledge concerning the decisions made and the reasons behind them.

Regarding the DR representation, the proposed approach uses argumentation and takes advantage of the following nodes:

- Evaluation: Problem being investigated which motivates the decision making process execution. An evaluation has goals, a set of feasible alternative solutions, criteria for assessing these alternative solutions and selecting the most appropriate one.
- Goal: Goals of the evaluation. This node can comprise information concerning possible barriers, for instance, time and restrictions. The goals can also comprise the requirements that must be met by the solution that will be selected at the end of the evaluation.

- Criteria: The identified alternative solutions are going to be evaluated against the defined criteria list. Each criterion has a weight that reflects the criteria importance for the final solution. Criteria can have a list of sub-criteria when the designer perceives that it is necessary to detail it more. In this case, only the sub-criteria will be assessed.
- Alternative Solution: Alternative Solutions for the problem being evaluated. These solutions will be assessed using the criteria in order to select the most appropriated solution.
- Criteria Appraisal: The criteria will be assessed regarding the alternative solutions. This appraisal comprises a value assignment that indicates how much the alternative solution meets the criteria. Besides, the designer must explain why he assigned a certain value to a particular criterion during the evaluation of an alternative solution. In addition, to increase objectivity, the organization can, institutionalize a list of criteria and their weight, according to business needs.

In addition to the process, a tool named TechSolution has been implemented in order to aid the process deployment. This tool, as said before, is available in the Taba Workstation and will be presented later in this section. The process and the tool are also being utilized by some Brazilian organizations engaged in enhancing their processes maturity and capacity. The following sections describe the process defined, the tool implemented and present some results from the use of the approach by a Brazilian organization throughout its process improvement initiative.

5.1 The Alternative Selection Process

The alternative selection process proposed comprises the following activities:

- ***Define Selection Goals and Constraints:*** Define goals and constraints regarding the evaluation and selection that will be accomplished in the following steps. It should be identified, for instance, schedule, resource and cost constraints. During this activity, the designer can look for a past appraisal that could help him during this appraisal. Moreover, the designer can consult the tool knowledge base in order to check if there is knowledge about this kind of appraisal available. For instance, if the designer needs to judge if the database tables primary keys are going to be created by the database itself or if the database table primary keys are going to be generated by the software, and if in a past project this decision has already been made, the designer would retrieve the DR available. In another case, if the designer must decide what architectural style to use, he would consult the knowledge provided by the tool that gives him a list of alternative solutions, criteria for evaluate these alternative solutions and the assessment of these alternatives solutions regarding the criteria provided. The examples described above are useful for development projects and also to maintenance projects. But, it is during this activity that the designer can look for a particular DR in order to realize the reasons behind a decision made.
- ***Establish Selection Criteria:*** Based on goals and expectations developed, selection criteria must be established. These selection criteria are going to be used to assess the alternative solutions. The selection criteria must be pondered in a way that the weigh reflects the criteria importance, in other words, a weight that reveal the importance in satisfying that criteria regarding the project must be assigned. It is also

necessary to define ranges and scales for each criteria, so that, later, during the assessment of a alternative solution regarding the criteria, a value can be selected in the range or scale established. In the case of a problem where there isn't a DR in a past project that could help the designer to solve it or that the tool doesn't have built-in knowledge about it, a set of general criteria will be suggested, then, the designer will be able to refine this set, adding new criteria and sub-criteria, removing criteria or modifying the criteria suggested. Therefore, when people in maintenance projects execute this activity, they will realize what criteria have been used during evaluation.

- ***Develop Alternative Solutions:*** During this activity, the designer will develop detailed alternative solutions. A wide range of alternatives appears asking to the maximum number of stakeholders for alternative solutions. Brainstorming sessions can stimulate new alternatives by fast iteration and feedback. [3]. The alternative solutions considered must be documented. At this moment, requirements must be mapped concerning the alternative solutions that will be investigated. Thus, it will be possible to track requirements regarding alternative solutions so that the designer will be able to see what requirements an alternative solutions is meeting and how the alternative solution is satisfying these requirements. This step assures that all the alternatives solutions meets costumer needs, meaning that wrong alternatives cannot be chosen. The deployment of this activity is very important for maintenance, when the designer will retrieve this knowledge that explains what alternative solutions have been investigated during the appraisal and why a particular solution has been selected among the others. In the event that the tool already has built-in knowledge about the appraisal or if the appraisal has already occurred in a past project, a list of alternative solutions will be suggested for the designer.
- ***Evolve Operational Concepts and Scenarios:*** During this activity, operational concepts, scenarios and environments to describe the conditions, operations modes and states specifics to each product component are evolved. Operational concepts and scenarios are evolved to ease the solution selection for the product components. Operational concepts and scenarios document the interaction of the product components with the environment, users and other product components, regardless of engineering discipline. The execution of this activity is necessary to provide more information to the designer, so that the designer can assess rightly the alternative solutions. As a result, the alternative solution that best satisfies the pre-established criteria can be selected.
- ***Evaluate Alternative Solutions:*** The evaluation aims to produce a technical report that will be the input for the selection activity. Each alternative solution must be assessed regarding the criteria established. All the assessment activity steps must be documented and this documentation must comprise, among other things, the reason why a particular alternative solution has been assigned a certain value during its evaluation among a particular criteria. At the ending of the evaluation, a report will be produced describing the alternative selection process performed, the criteria assessed, the alternative solutions considered, the requirements mapping concerning the alternative solutions and the assessment of the alternative solutions regarding the criteria. This report should describe the activities in a way that the reader can understand also the scope and the depth of the evaluation and that it can be reproduced. Again, in situations where the problem being analyzed has already been solved before, the designer will be able to retrieve the alternative solutions assessment regarding the

criteria of the past appraisal. In analogous way, if the tool has built-in knowledge about the appraisal, the designer will be also presented the assessment of the alternative solutions regarding the criteria.

- ***Select Solution:*** During the last process activity, an algorithm must be applied getting as input the results of the alternative solutions assessment activity. The Selection Solution begins when the assessment report is complete. After the execution of the selection algorithm, an alternative solution is suggested as output. The evaluations results can be evaluated and, then, reveal the necessity of additional information, resulting in the re-execution of some of the process activities that have already been executed before.

Regarding the graphical representation of the process, it uses the modeling language developed by VILLELA [19]. Figure 1 shows process defined.

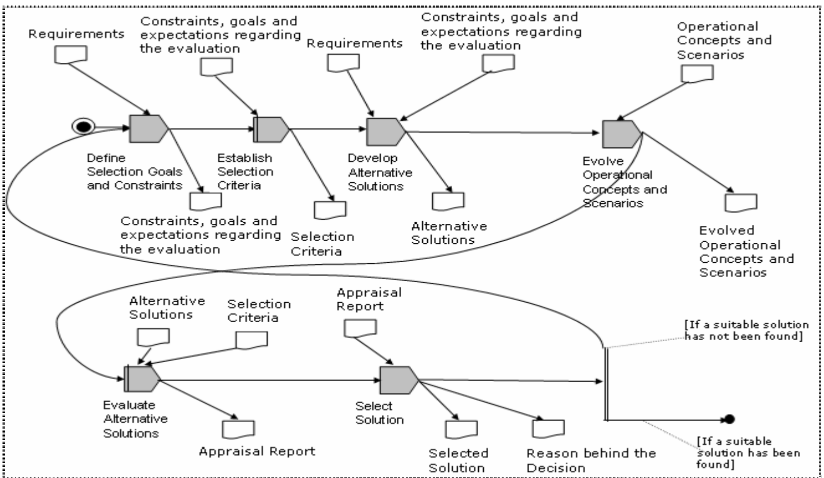

Fig. 1. Design Alternatives Selection Process

5.2 TechSolution: Supporting the Technical Solution Process Through DR

In order to support the designer through the deployment of the process activities a tool named TechSolution has been implemented. Since the beginning, the authors realized that a relevant requirement was to do not impact very much in the designer job, or the approach would fail. Therefore, the tool is available at Taba Workstation and designers don't need to leave the environment to register or retrieve the DR. Besides, other benefits as the integration of TechSolution with other tools could be enjoyed.

The Technical Solution tool provides Knowledge to solve a particular type of problem, the choice of the architectural style to be used. In this situation, TechSolution provides automatically for the technical solution evaluators specific set of criteria to be evaluated. Moreover, it also presents an alternative solutions list and indicates to the designer how best the solutions satisfy each selection criteria. Therefore, the alternative solutions given during the architectural style appraisal are different architectural styles. Besides, the tool is able to evaluate the relevance of each architectural style to the criteria established that has been proposed by the tool. All these tasks are done automatically, making very easy to the designer to deploy the process. The organization can input new knowledge about kinds of evaluations when necessary.

Even if the problem is unknown, in other words the tool has not knowledge about which specific set of criteria to use during the evaluation and selection, which alternative solutions could solve the problem and how best each of these alternative solutions satisfy the criteria established; the tool will help the user, among others ways, by presenting a list of selection criteria that he/she can use during the evaluation and selection of the alternative solutions and by supplying the designer with knowledge about how to perform the tasks.

Figure 2 presents the main interface of the TechSolution tool. On the left side of Figure 2, the system presents the activities that guide the execution of the tool. On the right side of the figure, the system presents another screen to support the execution of the selected activity; in this case, it is presented the screen that supports the Evaluation of Alternative Solutions Regarding the Selection Criteria as part of the evaluation activity.

Acknowledge supports the acquisition and retrieval of knowledge concerning development and maintenance process activities in order to allow the organizational knowledge reuse [17]. Acknowledge can be accessed from two icons located under the title bar of Taba Workstation. The integration of Acknowledge to TechSolution avoids interruption of organizations members' normal routine during knowledge capture and reuse [7].

The button ( ) allows the user to register a knowledge regarding the tool activity that is being performed at the moment. After that, the knowledge is going to be filtered and packaged by members of the knowledge committee. Then, this knowledge will be available for consulting during the tool activity execution by clicking on the icon ( ). Figure 2 also presents the integration between TechSolution and Acknowledge.

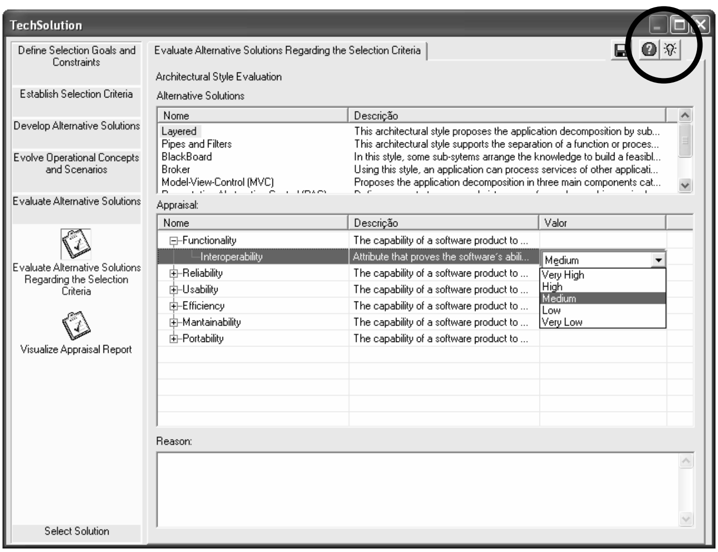

Fig. 2. TechSolution Main Interface

6 Results of the Approach Deployment by a Brazilian Organization

BL Informática, founded in 1987, is a Brazilian organization concerned with software development, maintenance, deployment, integration and factory. Its main objective, defined by the company quality policies, is to focus on customers, collaborators and stockholders satisfaction through solutions implementation in information technology, developed by defined, controlled and continuously improved processes which ensure the requirements achievement.

BL Informática had its processes evaluated against CMMI (Capability Maturity Model Integration) [3] Level 3 and 2 processes areas and is now a CMMI Level 3 adherent organization. During its process improvement initiative, in order to support the deployment of the Technical Solution and Decision Analysis and Resolution processes which should be adherent CMMI, BL Informática employed TechSolution and the process proposed.

Until now, the process has been performed for eight evaluations in 5 projects and during CMMI Level 3 appraisal, which evaluated 4 of these projects, the appraisal team considered TechSolution as a organizational strength to facilitate the success of Technical Solution and Decision Analysis and Resolution processes deployments initiatives and to overcome the inherent difficulties.

The suggested solution appointed by the use of the approach at BL Informática has been considered the most suitable in 7 evaluations. Merely one time, the customer claimed for another solution which was not the solution suggested by the decision making process executed. Besides, it has been perceived at BL Informática that developers tend to use design solutions that they have used in the past, even if the projects don't have the same requirements. The approach supports designer to take an objectively decision.

7 Conclusions

This paper presented an approach to support Knowledge Management in the Technical Solution Process through Design Rationale. Also the process defined and the tool implemented, which is named TechSolution and is integrated with Taba Workstation, have been described. Moreover, some information regarding the approach usage by a Brazilian Organization during its process improvement initiative aiming to be adherent to CMMI level 3 have been pointed out.

The main approach goal is to improve software quality by avoiding repeating errors made in past projects, supporting verification, knowledge management, documentation and maintenance.

A significant benefit perceived has been to reduce the effort needed to deploy the Technical Solution process by organizations that aim to be adherent to CMMI Level 3 and/or which seek the benefits of knowledge management through DR.

Acknowledgement

The authors wish to thank CNPq and CAPES for the financial support granted to the project Enterprise-Oriented Software Development Environments.

References

1. Birk, A., Tautz, C., Knowledge Management of Software Engineering Lessons Learned, IESE-Report 002.98/E, Jan (1998)
2. Burge, J. E., 2005, "Software Engineering Using design RATionale", PhD Dissertation, CS Dept., WPI, May.
3. Chrissis, M.B.., Konrad, M., Shrum, S., 2003, *CMMI: Guidelines for Process Integration and Product Improvement*, Addison-Wesley Publishing Company, Boston, USA.
4. Conklin, J., 1989, Design Rationale and Maintainability. Vol. II: Software Track, Proceedings of the Twenty-Second Annual Hawaii International Conference on Volume 2, 3-6 Jan. Page(s): 533-539. Vol. 2 Digital Object Identifier 10.11 09/HICSS. 1989.48049.
5. Decker, B., Althoff, K.-D., Nick, M., Tautz, C., Integrating Business Process Descriptions and Lessons Learned with an Experience Factory, In: Professionelles Wissensmanagement – Erfahrungen und Visionen (Beitrage der 1. Konferenz fur Professioneles Wissensmanagement), eds. Hans-Peter Schnurr, Steffen Staab, Rudi Studer, Gerd Stumme, York Sure. Baden-Baden, Germany, Shaker Verlag, Aachen, Mar (2001)
6. Desouza, K. C.: Barriers to Effective Use of Knowledge Management Systems in Software Engineering. Communications of ACM, Vol. 46(1) (2003) 99-101
7. Figueiredo, S.M., 2006, *Supporting Decision Making in the Technical Solution Process in Enterprise Oriented Software Development Environments*, M.Sc. Dissertation, COPPE/UFRJ, Rio de Janeiro, Brazil.
8. Franscisco, S.D., 2004, *DocRationale – a tool supporting Software Artifacts Design Rationale*, M.Sc. Dissertation, São Carlos Computing and Mathematic Science Institute, São Paulo University, Brazil.
9. Houdek, F., Bunse, C., Transfering Experience: A Practical Approach and its Application on Software Inspections, In: Proceedings of the 16th International Conference on Software Engineering and Knowledge Engineering, SEKE´04, pp. 427-430, Banff, Canada, June.
10. Holz, H., Konnecker, A., Maurer, F., Task Specific Knowledge Management in a Process Centered SEE, K-D Althoff, R. L. Feldmann, and W. Muller (Eds): LSO, LNCS 2176, (2001) 163-177
11. Kucza, T., Nattinen, M., Parviainen, P.: Improving Knowledge Management in Software Reuse Process. Lecture Notes in Computer Science, Vol. 2188. Springer-Verlag (2001) 141-152
12. Lee, J., Lai, K., What's in design rationale. Human-Computer Interaction, 6(3-4): 251-280 (1991).
13. Lindvall, M., Frey, M., Costa, P., *et al.*: Lessons Learned about Structuring and Describing Experience for Three Experience Bases. Lecture Notes in Computer Science, Vol. 2176. Springer-Verlag (2001) 106-119
14. Maurer, F., Holz, H., Process-Centered Knowledge Organization for Software Engineering, In: Papers of the AAAI-99 Workshop on Exploring Synergies of Knowledge Management and Case-Based-Reasoning, Orlando, Florida, Jul: AAAI Press, (1999)

15. Montoni, M., Santos, G., Villela, K., Miranda, R., Rocha, A.R., Travassos, G.H., Figueiredo, S., Mafra, S.: Knowledge Management in an Enterprise-Oriented Software Development Environment. In.: LNCS, ISBN 3-540-24088-8, pp. 117–128, presented at the 5th Int. Conf of Practical Aspects of Knowledge Management, Vienna, Austria, (2004)
16. Oh, E., Hoek, A.: Adapting Game Technology to Support Individual and Organizational Learning. Proceedings of SEKE'2001, Buenos Aires, Jun (2001) 347-362
17. Santos G., Montoni M., Rocha A. R., Figueiredo S., Mafra S., Albuquerque A., Paret B. D., Amaral M.: Using a Software Development Environment with Knowledge Management to Support Deploying Software Processes in Small and Medium Size Companies. In.: LNIA, ISBN 3-00-016020-5, pp 72-76, presented at the 3rd Conference Professional Knowledge Management Experiences and Visions, Kaiserslautern, Germany, April 10-13 (2005)
18. Souza, C, R. B., Wainer, J., Santos, D. B. e Dias, K. L. 1998, A model and tool for semi-automatic recording of design rationale in software diagrams. In: *Proceedings of the 6th String Processing and Information Retrieval Symposium & 5th International Workshop on Groupware*, Cancun, Mexico, 1998, p. 306-313.
19. Villela, K. V. C. (2004), Enterprise Oriented Software Development Environments Definition and Construction, D.Sc. Thesis, COPPE/UFRJ, Rio de Janeiro, Brazil.

Extraction and Analysis of Knowledge Worker Activities on Intranet

Peter Géczy, Noriaki Izumi, Shotaro Akaho, and Kôiti Hasida

National Institute of Advanced Industrial Science and Technology (AIST)
Tsukuba and Tokyo, Japan

Abstract. Knowledge regarding user browsing behavior on corporate Intranet may shed light on general behavioral principles of users in Intranet spaces, and assist organizations in making more informed decisions involving management, design, and use policies of Intranet resources. The study examines extraction and analysis of knowledge worker browsing behavior from WEB log data. Extraction of navigational primitives enabled us to identify common behavioral features of knowledge workers. Knowledge workers had a significant tendency to form behavioral patterns that were frequently repeated in Intranet environment. As they familiarized with the environment their navigation habituated.

1 Introduction

"Nobody has really looked at productivity in white collar work in a scientific way." (Peter Drucker) [1]. Absence of scientific evidence concerning knowledge worker productivity, efficiency, and their adequate measurement methods causes managerial difficulties [2]. Human dynamics [3] and observation of a behavior in electronic spaces [4] have been rapidly gaining importance in corporate sector. Corporations are eagerly exploiting ways to acquire more behavioral data about customers. Their interests are primarily commercially motivated [5].

Approaches to analyzing human behavior on world wide web rely on server-side data (WEB logs) generated by WEB servers, client-side data produced by script agents, behavioral studies, or their combinations. Conventionally, click-stream data is analyzed applying statistical modeling methods or empirical studies [6]. Markov models are frequently used due to their predictive accuracies [7], however, higher-order models become exceedingly complex and computationally expensive. Similar complications rise when utilizing adaptive learning strategies [8] at the processing stages where large data volumes still remain. Data mining only frequently occurring patterns can relax the computational complexity and improve speed of the methods [9]. Empirical investigations [10] provide only rule-based conclusions and are mostly unsuitable for predictive purposes.

In design of the next generation recommendation system improving knowledge worker effectiveness on Intranet we segment user behavior into elementary browsing patterns and their interlinks, and extract underlying behavioral abstractions. Linking these fundamentals enables formation of more complex observable behaviors and patterns. This study constitutes an exploratory analysis and provides relevant insights into behaviors of knowledge workers on Intranet.

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 73–85, 2006.

2 Approach Formulation

We present the basic approach to knowledge worker browsing behavior analysis together with definitions that help us better understand the concept at more formal level.

Click-stream sequences of visited pages are divided into sessions, and sessions are further divided into subsequences. Thus each session contains a train of subsequences. Division of sequences into subparts is done with respect to the user activity and inactivity. Consider the conventional time-stamp click-stream sequence of the following form: $\{(p_i, t_i)\}_i$, where p_i denotes the visited page URL_i at the time t_i. For the purpose of analysis this sequence is converted into the form: $\{(p_i, d_i)\}_i$ where d_i denotes a delay between the consecutive views $p_i \rightarrow p_{i+1}$. User browsing activity $\{(p_i, d_i)\}_i$ is divided into subelements according the periods of inactivity d_i satisfying certain criteria.

Definition 1. *(Browsing Session, Subsequence, Train, User Behavior)*
Let $\{(p_i, d_i)\}_i$ be a sequence of visited pages p_i with delays d_i between consecutive views $p_i \rightarrow p_{i+1}$.

Browsing session *is a sequence $BS = \{(p_i, d_i)\}_i$ where each $d_i \leq T_{BS}$. Browsing session is often in the further text referred to simply as a* **session***.*

Subsequence *of a browsing session BS is a sequence $S = \{(p_i, dp_i)\}_i$ where each delay $dp_i \leq T_S$, and $\{(p_i, dp_i)\}_i \subset BS$.*

A browsing session BS thus consists of a **train** *of subsequences S_i separated by inactivity delays ds_i, $BS = \{(S_i, ds_i)\}_i$.*

User behavior *is a sequence $UB = \{(BS_i, db_i)\}_i$ of user browsing sessions BS_i separated by delays db_i between consecutive sessions $BS_i \rightarrow BS_{i+1}$. For the purpose of simplicity we often consider user behavior to be a set of browsing sessions, $UB = \{BS_i\}$.*

Important issue is to determine the appropriate values of T_{BS} and T_S that split the user activity into sessions and subsequences. The former research [11] indicated that student browsing sessions last on average 25.5 minutes. However, we adopt the average maximum attention span of 1 hour as a value for T_{BS}. If the user's browsing activity was followed by a period of inactivity greater than 1 hour, it is considered a single session, and the following activity comprises the next session. Value of T_S is determined dynamically and computed as an average delay in a browsing session: $T_S = \frac{1}{N}\sum_{i=1}^{N} d_i$. If the delays between page vies are short, it is useful to bound the value of T_S from below. This is preferable in environments with frame-based and/or script generated pages. Since our situation contained both cases, we adjusted the value of T_S by bounding it from below by 30 seconds:

$$T_S = max\left(30, \frac{1}{N}\sum_{i=1}^{N} d_i\right) . \tag{1}$$

This approach allows detection and identification of specific user actions during a session. For example subsequence S_1 corresponds to login, subsequence S_2 is a download of a document, subsequence S_3 is a submission of a form, etc.

Another important aspect is to observe where the user actions are initiated and terminated, and what connects them. That is, to identify the starting and ending points of subsequences, and the transitions between subsequences.

Definition 2. *(SE Elements, Connectors)*
Let $BS = \{(S_i, ds_i)\}_i$ *be a browsing session and* $S_i = \{(p_{ik}, dp_{ik})\}_k^N$ *and* $S_{i+1} = \{(p_{i+1l}, dp_{i+1l})\}_l^M$ *be consecutive subsequences* $S_i \to S_{i+1}$ *of* BS*.*
SE element *(start-end element) of a subsequence* S_i *is a pair* $SE_i = (p_{i1}, p_{iN})$*.*
Connector *of subsequences* S_i *and* S_{i+1} *is a pair* $C_i = (p_{iN}, p_{i+1,1})$*.*

Each user's browsing behavior is then further associated with sets of SE elements and connectors. This enables us to distinguish frequent browsing patterns and in the future work categorize users.

3 Intranet and Data

Data used in this work was a one year period Intranet WEB log data of The National Institute of Advanced Industrial Science and Technology. The majority of users are skilled knowledge workers. Intranet WEB portal had load balancing architecture comprising of 6 servers providing extensive range of WEB services and documents vital to the organization. Intranet services support managerial, administration and accounting processes, research cooperation with industry and other institutes, databases of research achievements, resource localization and search, attendance verification, and also numerous bulletin boards and document downloads. The institution has a number of branches at various locations throughout the country, thus certain services are decentralized. The size of visible WEB space was approximately 1 GB. Invisible WEB size was considerably larger, but difficult to estimate due to the distributed architecture and constantly changing back-end data.

Table 1. Basic information about data used in the study

Data Volume	~60 GB
Average Daily Data Volume	~54 MB
Number of Servers	6
Number of Log Files	6814
Average File Size	~9 MB
Time Period	March 2005 — April 2006

Daily traffic was substantial and so was the data volume. It is important to note that the data was incomplete. Although some days were completely represented, every month there were missing logs from specific servers. Server side logs also suffered data loss due to caching and proxing. However, because of the large data volume, missing data only marginally affected the analysis. WEB servers run open source Apache server software and the WEB log data was in the combined log format without referrer.

4 Knowledge Worker Browsing Behavior on Intranet

Starting with the setup description we present the data preprocessing and cleaning, followed by session and subsequence extraction, elimination of machine generated traffic, and detection of SE elements and connectors. Analysis of knowledge worker browsing behavior based on the extracted features is provided in the specific subsection.

Setup. Computing environment consisted of Linux (FC5) setup with MySQL database as a data storage engine. Analytic and processing routines were implemented in various programming languages and optimized for high performance. Processing of large data volumes was computationally and time demanding.

4.1 Extraction of Knowledge Worker Browsing Features

Preprocessing. Data fusion of WEB logs from 6 servers of a load balanced Intranet architecture was performed at the preprocessing level. Data was largely contaminated by logs from automatic monitoring software and required filtering. During the initial filtering phase logs from software monitors, invalid requests, WEB graphics, style sheets, and client-side scripts were eliminated. Access logs from scripts, downloadable and syndicated resources, and documents in various formats were preserved. The information was structured according to the originating IP address, complete URL, base URL, script parameters, date-time stamp, source identification, and basic statistics. Clean raw data was logged into database and appropriately linked.

Table 2. Primary WEB log data statistics

Number of Log Records	315 005 952
Number of Clean Log Records	126 483 295
Number of Unique IP Addresses	22 077
Number of Unique URLs	3 015 848
Scripts	2 855 549
HTML Documents	35 532
PDF Documents	33 305
DOC Documents	4 385
Others	87 077

Approximately 40.15% of the original log records remined after initial filtering (see Table 2). Major access to Intranet resources was via scripts (94.68%). Only relatively minor portions of accessible resources were HTML documents (1.18%), PDF documents (1.1%), DOC documents (0.15%), and others (2.89%), such as downloadable software, updates, spreadsheets, syndicated resources, etc. Detected IP address space (22077 unique IPs) consisted of both statically and dynamically assigned IP addresses. Smaller portion of IP addresses were static, and relatively uniquely associable with users.

Session Extraction. Preprocessed and databased Apache WEB logs (in combined log format) did not contain referrer information. Click-stream sequences were reconstructed by ordering logs originating from unique IP addresses according to time-stamp information. Ordered log sequences from the specific IP addresses were divided into the browsing sessions as described in Definition 1. Divisor between sessions was the user inactivity period ds_i greater than $T_{BS} = 1\ hour$.

Table 3. Observed basic session data statistics

Number of Sessions	3 454 243
Number of Unique Sessions	2 704 067
Average Number of Sessions per Day	9 464
Average Session Length	36 [URL transitions]
Average Session Duration	2 912.23 [s] (48 min 32 sec)
Average Page Transition Delay per Session	81.55 [s] (1 min 22 sec)
Average Number of Sessions per IP Address	156
Maximum	1 553
Minimum	1

It is noticeable that user sessions on the corporate Intranet are on average longer (appx. 48.5 minutes) than those of students (appx. 25.5 minutes) reported in [11]. Average number of 156 sessions per IP address, and large variation in maximum and minimum number of sequences from distinct IP addresses, indicate that association of particular users with distinct IP addresses is relevant only for registered static IP addresses. Large number (3492) of single sessions only originated from distinct IP addresses due to wide DHCP use. It is possible to employ clustering techniques to identify reasonably diverse groups of users.

Subsequence Extraction. Each detected session was analyzed for subsequences as defined in Definition 1. Segmenting element dividing sessions into subsequences was the delay between page transitions $dp_i > T_S$, where T_S was determined according to (1). Lower bound of 30 seconds for the separating inactivity period dp_i was proper.

It has been observed that sessions contained machine generated subsequences. Periodic machine traffic with inactivity time less than the session separator delay could result in long session sequences. As seen in the histogram of average delays between subsequences (Figure 1), there was a disproportionally large number of sessions with average delays between subsequences around 30 minutes and 1 hour. This is indicated by spikes in the main chart of Figure 1. Detailed view (subcharts of Figure 1) revealed that the variation in the average delay between subsequences is approximately ± 3 seconds. It well corresponds to the peak in the histogram of average subsequence duration (Figure 2-a). It is highly unlikely that human generated traffic would produce this precision (although certain subsequences were legitimate).

Machine generated traffic contaminates the data and should be filtered, since we target primarily human behavior on the Intranet. We filtered two main groups of machine generated subsequences: login subsequences and subsequences with delay periodicity around 30 minutes and 1 hour.

Every user is required to login into Intranet in order to access the services and resources. Login procedure involves validation and generates several log records with 0 delays. Records vary depending on whether the login was successful or unsuccessful. In both cases the log records and login related subsequences can be clearly identified and filtered.

The second group of machine generated traffic are subsequences with periodicity of 30 minutes and 1 hour. Direct way of identifying these subsequences is to search for sessions with only two subsequences having less than 1 second (or 0 second) duration (machines can generate requests fast and local Intranet servers are capable of responding within milliseconds) and delay ds_i between subsequences within the intervals: 1800 and 3600 ± 3 seconds. It has been discovered that substantial number of such sessions contained relatively small number (170) of unique subsequences. Furthermore, these subsequences contained only 120 unique URLs. Identified subsequences and URLs were considered to be machine generated and filtered from further analysis. Moreover, the subsequences with SE elements containing identified URLs were also filtered.

Table 4. Observed basic subsequence data statistics

Number of Subsequences	7 335 577
Number of Valid Subsequences	3 156 310
Number of Filtered Subsequences	4 179 267
Number of Unique Subsequences	3 547 170
Number of Unique Valid Subsequences	1 644 848
Average Number of Subsequences per Session	3
Average Subsequence Length	4.52 [URL transitions]
Average Subsequence Duration	30.68 [s]
Average Delay between Subsequences	388.46 [s] (6 min 28 sec)

Filtering of detected machine generated subsequences and their URLs significantly reduced the total number of subsequences - by 56.97% (from 7335577 to 3156310), as well as the number of unique subsequences - by 46.37% (from 3547170 to 1644848). Since the login sequences were also filtered, the number of subsequences per session decreased at least by 1. Reduction also occurred in the session lengths due to filtering of identified invalid URLs. Filtering did not significantly affect the duration of subsequences because the logs of machine

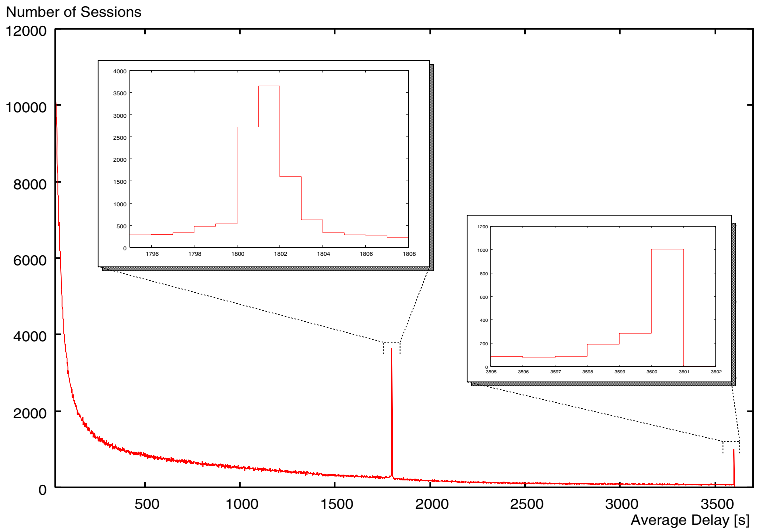

Fig. 1. Histogram of the average delay between subsequences in sessions. There are noticeable spikes around 1800 seconds (30 minutes) and 3600 seconds (1 hour). The detailed view is displayed in subcharts. The spikes with relatively accurate delays between subsequences are due to machine generated rather than human traffic.

generated subsequences occurred in rapid transitions with almost 0 durations and delays. It is noticeable that the average subsequence duration (30.68 seconds) is approximately equal to the chosen lower bound for ds_i (30 seconds). This empirically justifies the right choice of lower bound for T_S.

SE Elements and Connectors. Extraction of SE elements of subsequences and connectors between subsequences is relatively straightforward. SE elements and connectors also undergone filtering. If invalid URLs were present in at least one element of a pair, the respective SE element and/or connector was marked as invalid.

Table 5. Observed basic statistics for SE elements and connectors

SE Elements	7 335 577
Valid SE Elements	3 156 310
Filtered SE Elements	4 179 267
Unique SE Elements	1 540 093
Unique Valid SE Elements	1 072 340
Connectors	3 952 429
Valid Connectors	2 346 438
Filtered Connectors	1 605 991
Unique Connectors	1 142 700
Unique Valid Connectors	898 896

There is a noticeable reduction of SE elements and connectors due to the filtering. Number of SE elements decreased by 56.97% (from 7335577 to 3156310) and connectors by 40.63% (from 3952429 to 2346438). Similarly, reduction is evident in the number of unique SE elements (30.37%: from 1540093 to 1072340) and connectors (21.34%: from 1142700 to 898896).

4.2 Analysis of Knowledge Worker Browsing Behavior

The aim of analysis is to elucidate common browsing features shared by knowledge workers. Analysis of processed Intranet WEB log data provides numerous useful insights into their browsing and navigation behavior and can be utilized for improving the front-end as well as back-end Intranet design in various ways.

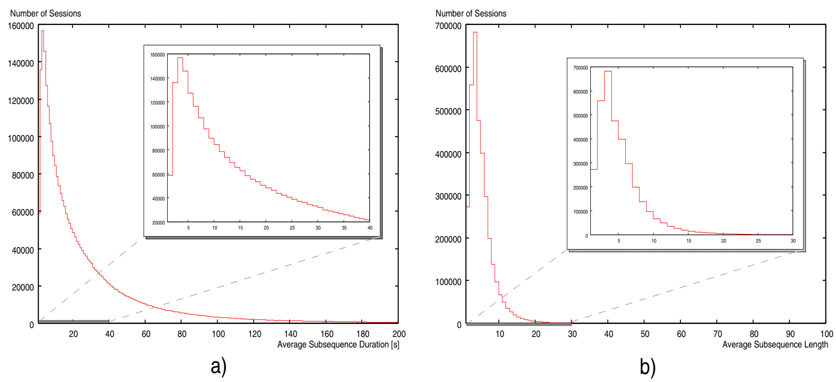

Fig. 2. Histograms: **a)** average subsequence durations in sessions, **b)** average subsequence lengths in sessions. The peak of both distribution curves is at the rounded average value equal to 3.

Frequent users knew their targets and navigational paths to reach them. Duration of subsequences in sessions was short - with peak in the interval of two to five seconds (see histogram in Figure 2-a). During such short period users were able to navigate through four pages on average (see Table 4 and Figure 2-b) in order to reach the target. Since there was approximately one second per page transition, there was virtually no time to thoroughly scan the page. Therefore it is reasonable to assume knowledge workers knew where the next navigational point was located on the given page and proceed directly there. There was little exploratory behavior.

Session objective was accomplished via few subgoals. Average session (after filtering) contained three subsequences (see Table 4) where each subsequence can be considered a separate action and/or subgoal. Average knowledge worker spent about 30 seconds to reach the subgoal/resource, and additional 6.5 minutes before taking another action. Considering the number of unique valid subsequences

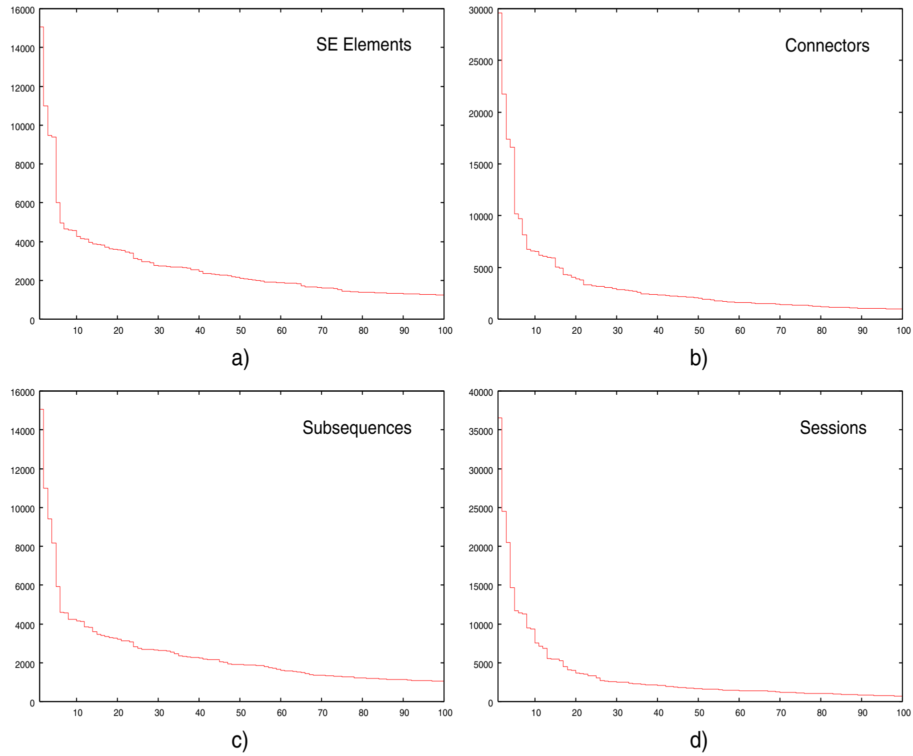

Fig. 3. Histograms of the most frequent 100 unique: **a)** SE elements, **b)** connectors, **c)** subsequences, and **d)** sessions. All histograms have long tails with low frequencies of elements.

(about 1.6 million) the complete population of users had relatively wide spectrum of browsing patterns. However, the narrow explored Intranet space of a single user suggests large diversification.

Knowledge workers form browsing behavioral patterns and often repeat them. Formation of behavioral browsing patterns is strongly evident from the repetition of SE elements (66.03%), connectors (61.69%), subsequences (47.89%), and also the complete sessions (21.72%). Histograms in Figure 3 depict the frequencies of one hundred most re-occurring SE elements, connectors, subsequences, and sessions. It can be seen that relatively small number of them were highly repetitive. This evidence is further supported in percentile analysis (see Table 6). Approximately 0.2% of unique SE elements, connectors, and subsequences were occurring frequently enough to account for 30% of total valid observations. Furthermore, less than 6% of them accounted for 50% of valid observations.

Since SE elements represent starting and ending points of subsequences (or corresponding subgoals), their re-occurrence suggests that knowledge workers utilized the same basic browsing patterns. Connectors delineate transitions between subgoals, and their repetition suggests formation of more complex patterns. Difference

Table 6. Percentile analysis of valid SE elements, connectors, subsequences, and sessions. The values represent what percentage of unique valid elements accounts for a given percentage of total valid elements (e.g. SE element value 0.01 indicates that 0.01% of unique valid SE elements accounts for 10% of total valid SE elements.)

	Percentiles [%]								
	10	20	30	40	50	60	70	80	90
SE Elements	0.01	0.06	0.2	0.61	1.67	4.72	14.77	41.13	70.57
Connectors	0.004	0.04	0.19	0.82	2.86	8.51	21.69	47.79	73.9
Subsequences	0.01	0.05	0.24	1.06	5.79	23.24	42.43	61.62	80.81
Sessions	0.004	0.75	10.58	23.35	36.12	48.9	61.68	74.45	87.22

in repetitive frequencies between SE elements and subsequences highlights the fact that although the users were aiming at the same subgoal, they were taking different paths to reach it.

Formation of behavioral browsing patterns positively correlates with short peak average duration of subsequences (3 seconds). Knowledge workers with formed browsing patterns exhibited relatively fast page transitions. They also displayed shorter delays between subsequences.

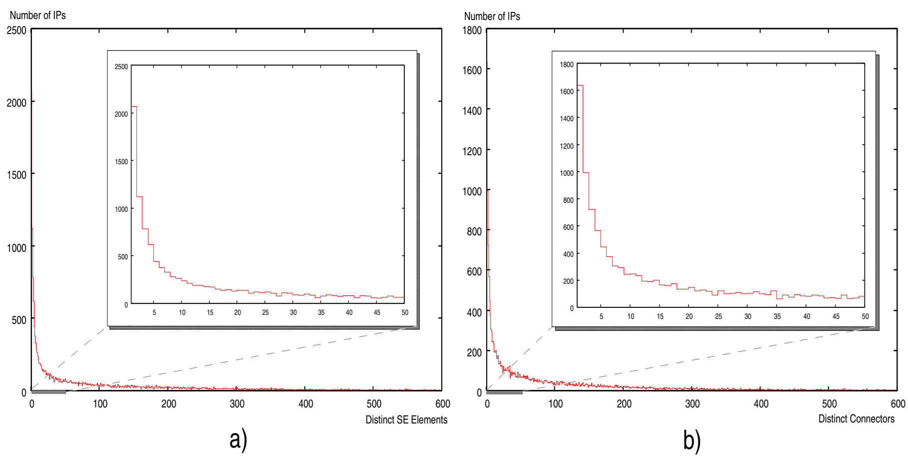

Fig. 4. Histograms: **a)** number of distinct SE elements in relation to the number of distinct IPs; **b)** number of distinct connectors in relation to the number of distinct IPs

Observation of subgoal patterns, and formation of more complex patterns of user browsing behavior gives rise to the possible user classification. Users with significantly overlapping behavioral patterns may have related working habits (and/or job descriptions) despite their physical location. Information about knowledge worker browsing patterns can also be utilized for development of personalized interfaces, services, usability analysis, and search functionalities.

Knowledge workers had focused interests and explored only narrow spectrum of Intranet resources. Histograms in Figure 4 indicate that unique SE elements and connectors per distinct IP addresses were in the range of hundreds. That is, during the one year period knowledge workers targeted only few hundred resources out of approximately 3 million available (see Table 2). They performed routine tasks on Intranet (submitted forms, accessed local bulletin boards and news, downloaded internal documents, searched for very specific resources, etc.). This could be attributed to the focused interests, work description, and role within the organization.

Large number of subsequences or even sessions contained only single element. This suggests either frequent use of hotlists such as bookmarks and history [10], or substantial period of inactivity between single actions (such as reloading the page, accessing another resource from hotlist, etc.). In the working environment, it is a common practice that the priority is given to the immediate internal events rather than to Intranet tasks. There is possibility that after resolving the internal issues knowledge workers returned to complete their browsing tasks, however, after substantial delay. Thus their following activity was detected as a new subsequence or session.

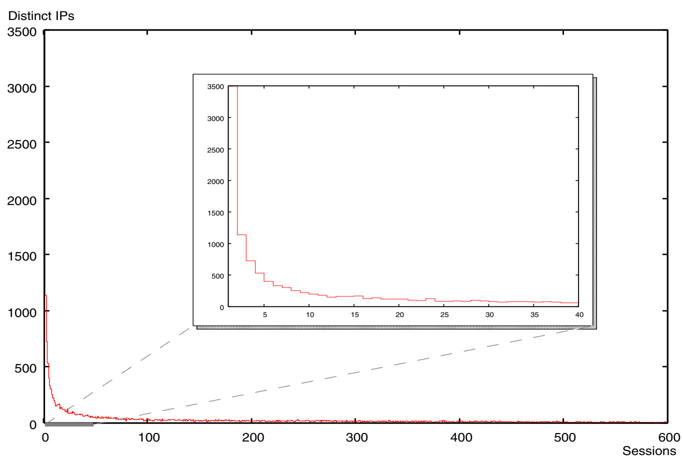

Fig. 5. Histogram depicting the number of sessions with respect to distinct IPs

Management and support of knowledge workers Intranet activities can be provided at various levels. *Browsing Management:* assistance tools that optimize knowledge worker browsing behavior and access in alignment with organizational goals and structure can be implemented. *Attention Management:* adequate information selection and filtering that is brought to relevant knowledge worker attention in a timely and appropriate manner can be performed. *Personalization Management:* personalized support for frequent routine tasks according to organizational policies can be provided, so the knowledge workers can concentrate on creative tasks that have the potential to generate new value.

Association of users with unique IP addresses is not appropriate in Intranets largely utilizing dynamically assigned IP addresses. Static addresses may provide direct identification of users. It can be noticed from Figure 5 that use of DHCP shifted the higher frequency of unique IP addresses toward smaller number of sessions originating from them. This is in a contrary to desirable situation allowing user identification.

5 Conclusions and Future Work

Extraction of navigational primitives and analysis of knowledge worker browsing behavior was performed. Knowledge workers navigational activities on corporate Intranet were segmented into sessions and subsequences depending on the periods of inactivity between actions. Higher abstraction components: SE elements and connectors, were extracted from subsequences and sessions. Analysis of these fundamentals revealed important behavioral features. Session objectives were accomplished via relatively small number of subgoals. Knowledge workers had well defined targets and knew how to achieve them. Utilization of Intranet platform led to formation of elemental and complex browsing patterns that were repeatedly applied. Knowledge workers had focused interests and explored diminutive range of Intranet resources. Partial limitation of the study is reliance on WEB log data only. Additional information sources regarding knowledge worker browsing behavior should be incorporated.

Aim of the future work is classification of knowledge workers according to their behavioral patterns in connection with content based analysis. Results should form the basis for the next generation of recommender systems and usability analysis.

Acknowledgment

The authors would like to thank Tsukuba Advanced Computing Center (TACC) for providing raw WEB log data.

References

1. B. Schlender. Peter Drucker sets us straight. *Fortune*, (December 29, 2003), http://www.fortune.com.
2. T.H. Davenport. *Thinking for a Living - How to Get Better Performance and Results from Knowledge Workers*. Harvard Business School Press, Boston, 2005.
3. A.L. Barabasi. The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207–211, 2005.
4. Y-H. Park and P.S. Fader. Modeling browsing behavior at multiple websites. *Marketing Science*, 23:280–303, 2004.
5. W.W. Moe. Buying, searching, or browsing: Differentiating between online shoppers using in-store navigational clickstream. *Journal of Consumer Psychology*, 13:29–39, 2003.
6. R.E. Bucklin and C. Sismeiro. A model of web site browsing behavior estimated on clickstream data. *Journal of Marketing Research*, 40:249–267, 2003.

7. M. Deshpande and G. Karypis. Selective markov models for predicting web page accesses. *ACM Transactions on Internet Technology*, 4:163–184, 2004.
8. I. Zukerman and D.W. Albrecht. Predictive statistical models for user modeling. *User Modeling and User-Adapted Interaction*, 11:5–18, 2001.
9. J. Jozefowska, A. Lawrynowicz, and T. Lukaszewski. Faster frequent pattern mining from the semantic web. *Intelligent Information Processing and Web Mining, Advances in Soft Computing*, pp. 121–130, 2006.
10. M.V. Thakor, W. Borsuk, and M. Kalamas. Hotlists and web browsing behavior–an empirical investigation. *Journal of Business Research*, 57:776–786, 2004.
11. L. Catledge and J. Pitkow. Characterizing browsing strategies in the world wide web. *Computer Networks and ISDN Systems*, 27:1065–1073, 1995.

Knowledge Sharing to Support Collaborative Engineering at PLM Environment

David Guerra-Zubiaga[1], Laurent Donato[2], Ricardo Ramírez[1], and Manuel Contero[3]

[1] Centro de Innovación en Diseño y Tecnología
Tecnológico de Monterrey, Campus Monterrey, Avenida Eugenio Garza Sada 2501 Sur, Monterrey, N.L., 64849, México
{david.guerra, ricardo.ramirez}@itesm.mx
[2] Ecole d´ingénieurs et d´architectes de Fribourg, Laboratoire de Développement de Produits, CH-1705 Fribourg, Switzerland
laurent.donato@hefr.ch
[3] Universidad Politécnica de Valencia, Escuela Técnica Superior de Ingenieros Industriales, 46022 Valencia, España
mcontero@degi.upv.es

Abstract. Nowadays, the success of process design relies on an effective communication between collaborative design team members, among others. Product Lifecycle Management (PLM) tools represent a key platform to support knowledge sharing at multi-disciplinary design teams working under collaborative engineering concepts. However, an important issue is to capture valuable expertise at collaborative work using the PLM platform reusing key knowledge. This paper contributes to knowledge management showing how the usage of knowledge structures at a PLM environment is suitable for assembly engineering applications. The present research encompasses multidisciplinary engineering work teams defining the assembly process of an airplane part. This paper argues that knowledge structures enable the capture and exchange of expertise at collaborative work increasing value added between collaborators. A case study is presented to validate this idea.

Keywords: Knowledge Structures, PLM, Collaborative Engineering, Knowledge sharing, Assembly process, Aerospace Industry.

1 Introduction

Design is an important process for manufacturing companies to gain competitiveness [1]. For this reason, design processes have gained more attention nowadays [2]. The new key resource is inside the heads of people: knowledge. What organizations know and the way they use it, is essential for success [3]. An important manner of supporting a design process is through information and knowledge sharing. Knowledge structures can be built to store, manage, classify and use information and knowledge, in order to facilitate the process design [4].

The critical points generated at process development are closely related with a reciprocal demand between costs, quality and time. Collaborative engineering works towards the reduction and elimination of conflicts between them. Additionally,

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 86 – 96, 2006.

collaborative engineering teams allow parallel work minimizing design time, cycles and redundancy work [4]. Rose et al. [5] proposed a network structure involving actors, resources and activities, describing the relationships between them under collaborative engineering environments. In other words, when multi-disciplinary work is being performed, each team member has access to the entire knowledge base and shares it; knowledge and information structures organize the knowledge base, enabling its organization, acquisition, access, management, share and reuse. The correct use of the structures increases the collective expertise.

Product Lifecycle Management (PLM) strategy promises to integrate all the information related to a product lifecycle and make it available to every person involved in its development. PLM tools are powerful resources for creating a collaborative engineering environment [7]. They support engineering solutions for design in virtually every branch of manufacture. Process planning, assembly design, layout evaluations and other resources are examples of the broad band of applications available at a PLM environment [8]. In the present work, the application of these tools on a specific case study for a company of the aerospace industry (ICKTAR) will be discussed. This was done working in a collaborative engineering environment making multi-disciplinary efforts together with an academic team from México (Tecnológico de Monterrey).

The aim of this paper is to discuss the use of knowledge structures to support the assembly process design through collaborative engineering teams at a PLM environment.

2 Knowledge Structures for Manufacturing Knowledge

The concept of manufacturing modeling has evolved thanks to the efforts of various research groups which have analyzed processes and resources in manufacturing facilities, as the work presented by [9, 11, and 12]. Additionally, the advance of the research has already been expanded on information and knowledge modeling used to design reconfigurable-design methodologies. Information and knowledge structures were used to classify processes, resources and knowledge to develop a reconfigurable manufacturing facility [13]. However, knowledge structures can be applied to other activities of the product lifecycle, such as assembly design.

A knowledge structure is the representation of a knowledge model, utilized in this research to define the classification of a manufacturing facility. It is important to broaden the field of application for the knowledge structure concept. It could be applied to a wide range of businesses, industries or any other system in which value-adding procedures are developed, such as financial institutions, non-profit organizations, etc.

Fig. 1 shows a structure in UML notation that classifies the information and knowledge in a facility, where a shop, a cell or a station can be considered as "facility". The information classes are in the top part of the structure, and the knowledge classes are in the bottom part of it. The facility is described by the main categories, which are process and resource. Facility knowledge is classified in process knowledge and in resource knowledge; the three types of knowledge used are explicit, tacit and implicit.

At this point, it is important to define data, information and knowledge. "Data relates simply to words or numbers the meaning of which may vary and is dependent upon the context in which the data is used". "Information is data, which is structured or titled in some way so that it has a particular meaning". "Knowledge is information with added detail relating to how it may be used or applied" [10].

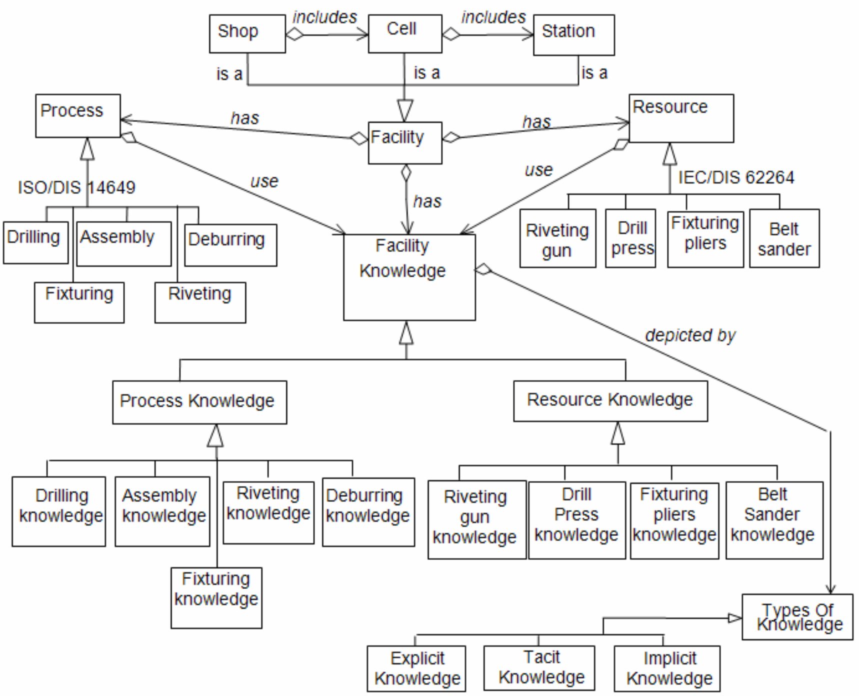

Fig. 1. Knowledge, Process and Resources (KPR) structure for assembly process design (adapted from [9])

This structure is an important tool when storing, classifying and accessing knowledge is required, especially in process planning and assembly design operations, allowing the implementation of knowledge sharing techniques [9].

When using the KPR structure, the first thing to do is to choose the facility under study (a manufacturing cell, for instance); then, what is known about the processes and resources must be defined in order to build the knowledge structures to support the design process [1]. The following step is to classify the data, information and knowledge related to assembly processes and resources within the KPR structure. This classified assembly knowledge is relevant for PLM concepts, and a mutual support between a PLM environment and a collaborative engineering effort is achieved using this structure as a link.

The application of the KPR structure provides a knowledge and information structure for assembly design performed by collaborative engineering teams. This concept will be analyzed more thoroughly on section 4; furthermore, section 4 presents as Case Study an application of the KPR structure to five workstations of an assembly shop, to demonstrate its ability to support knowledge classification and sharing.

3 Project Collaboration at PLM Environment

PLM tools provide solutions for a wide range of design and manufacturing applications [13]. For example, reconfigurable design and assembly design. A PLM environment allows multi-disciplinary teams to work and share experiences at a linked information platform. However, there is a need to develop additional tools to organize the knowledge to be used at the PLM platform [13]. The KPR structure presented in Fig. 1 is proposed as an additional tool to solve this problem. The KPR structure is not only used to organize knowledge to facilitate multi-disciplinary teams to work at PLM environment, but also to support PLM engineers' decisions by sharing their experiences. This fact increases the efficiency of design process by reducing the time that is usually lost when knowledge is not properly classified, stored and accessed. On the other hand, using only one commercial PLM brand at a time enables communication between the different modules (tools) of the software. These tools support the design work, consider communication and coordination features, and intend to create collaboration links among actors.

The concept of collaborative engineering was developed with the intention of improving design processes in terms of efficiency, time consumption and innovation [14]. Design in engineering needs to be supported by different viewpoints, but each of these viewpoints must be related to each other. In a project collaboration environment, even though each member has specific tasks to accomplish, all the members are related to a common goal [15]. This approach, aligned with proper knowledge and information structures facilitates the definition of a clear work path.

The idea is that every member involved in the assembly process design already has a common knowledge and expertise that is going to be enriched with the contribution of all the team. The exchanged knowledge needs to be accurate, especially between designers. This fact will support the decision-making process. Common knowledge brought by designers is not enough to perform a collaborative engineering task. Each collaborative part must work on bridging the gaps between them and focus on mutual understanding of the problem to be solved. Design process is related to the creation of innovating concepts that must be understood by collaborators.

Fig. 2 depicts the concept of collaborative engineering environment supporting assembly process design, where each element plays a role in information/knowledge sharing. With this approach, every advance accomplished by any element is *visible* to the rest of the group, therefore allowing true collective expertise growth; these elements have a common connection when the KPR structure is being used. According to figure 2, contributions from research and documentation (2), as well as experts (3) of the assembly process design (6) can be absorbed by the designers (4), some of whom use PLM digital tools (5).

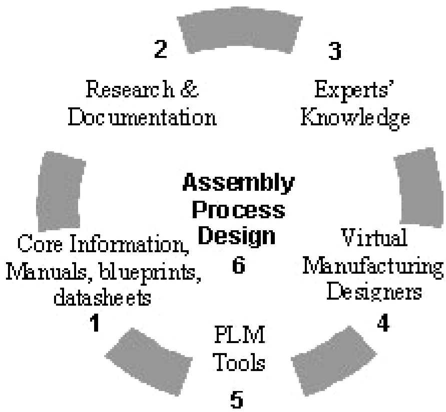

Fig. 2. Collaborative engineering environment in assembly process design

The five elements shown in Fig. 2 are strongly related to a knowledge sharing environment to support assembly process design under a collaborative engineering framework. The main elements of the framework are:

Core information (#1). It involves all the information obtained from the resources of the assembly: product design information, assembly manuals, blueprints & technical drawings, specification sheets. This information is the basic source to define the design problem.

Research and documentation (#2). This element refers to an investigation about similar processes performed in the past, and in the present time. This investigation involves the documentation of patents, procedures and other processes that can be used to understand the factors affecting the assembly process.

Expert knowledge (#3). The contribution from experts in the specific disciplines involved in the product or process to be designed is vital. They can provide valuable recommendations regarding new trends and different approaches. A feedback link can be created with the rest of the team to extract all the expertise they carry with themselves.

Virtual Manufacturing Designers (#4). They are the team responsible for conceiving and evaluating assembly processes by using the knowledge and tools from the rest of the elements. The main intellectual work is performed by them, and they receive support from the rest of the elements.

PLM tools (#5). PLM tools are virtual digital tools that support technical analysis scenarios in product development processes within PLM environments. In this research work, the different DELMIA® modules are called PLM tools. The particular, DELMIA® modules used in the present work are: "part design", "layout planning", "factory flow simulation", "ergonomics" and "DMU kinematics".

While multi-disciplinary work is being performed, a common goal is achieved by approaching from different angles, allowing parallel working with a shared direction. Therefore, the central focus is receiving, storing, sharing and using a large amount of information and knowledge; these tasks could be accomplished either manually or automatically using PLM digital tools.

This knowledge and information integration is enhanced when a PLM working environment is present, because not only the different people working in their respective teams have a direct connection with the goal, but also their computational equipment and software. The latter enhances the designer's abilities to perform their tasks more efficiently.

4 Case Study

The assembly design of true-scale aircraft RV-10 is represented as case study in this paper. This is a low wing airplane with a fixed tricycle landing gear and uses slotted flaps and mass-balanced control surfaces. The primary structure is made of aluminum with a composite cabin top and doors. The RV-10 is a metal aircraft designed to accept engines in the 200-260 hp range. It is intended as a moderately fast touring airplane that can carry up to four real people, a reasonable amount of baggage and about 60 gallons of fuel. It is not aerobatic, but it is reported to have an excellent low-speed handling and control. Its assembly is carried out by very small aluminum rivets, nearly 70 000 of them. The aircraft is sold as a kit with all the necessary parts disassembled, and it takes about 2 000 man-hours of work to assembly. This aircraft was thoroughly studied by applying reverse engineering techniques [16].

For this case study, one of the objectives was to provide a design solution for layout planning to assist the assembly process. Due to the extension of the design task, it was divided in sub-projects; this case study regards a specific part of the project focused on manufacturing engineering. It was called Virtual Manufacturing, and was assigned to a collaborative team formed by mechanical engineering students at their senior year. An important fact is that these students were from two different universities, ITESM (Monterrey, Mexico) and EIA (Fribourg, Switzerland), and they had advisors from both universities, as well as from the company staff itself. ICKTAR also contacted an expert on the aerospace mechanic industry to complement the support. They all worked on the assembly process design, using PLM digital manufacturing tools.

Before applying collaborative work, it is important to define the knowledge structures to be used for information and knowledge sharing. Figure 1 depicts the knowledge structure used to support the collaborative work, which in turn is based on Figure 2. Members of the collaborative design team classified data, information and knowledge according to KPR structure by hand on a notebook, then not only the structure but also the data, information and knowledge classified in this structure were used. For example different teams defined the riveting process name and description as data. The riveting definition was presented as information. In a similar manner, the riveting knowledge to be applied at assembly process was identified and classified. The riveting data, information and knowledge were used by the collaborative members at the assembly process design. For example**,** the knowledge of the riveting, fixturing and drilling processes were gathered, studied and finally used at the airplane's empennage assembly.

It is important to clarify that the next three issues are not considered in the present research work:

a. The development of an AI mechanism to generate new knowledge
b. The development of an AI mechanism to retrieve the knowledge from the knowledge-base, and
c. The maintenance of the knowledge captured.

The knowledge-maintenance is already being explored in other papers of the authors. On the other hand, the structure provided a manner to gather, classify and share the knowledge and expertise of all the team members. Since the team members were geographically separated, this approach to knowledge-management made possible the standardization of the concepts and processes that were going to be used along the project. However, the structures could be used as a blueprint to create the AI mechanism mentioned above (a, b).

As the airplane assembly process evolved, the knowledge structure was improved according to the airplane assembly process knowledge. The different assembly teams used the classified knowledge to support collaborative engineering. For example, the riveting process knowledge was shared by the team members that were directly involved in this process.

PLM tools played a crucial role in this project, since they are meant to be information and data managing tools. This case study validated the usefulness of merging KPR structures and PLM tools in a collaborative engineering task. The next paragraphs explain how the research ideas were implemented.

Figure 3 depicts the elements of the research concept in three columns throughout this case study implementation. It shows an overview and links through the project idea. First column contains the KPR structure used at airplane assembly process part. This structure classifies the information and knowledge related to processes and resources at the assembly facility fully explained at figure 1. The KPR structure is used as a guide to the collaborative environment work showed at column two. The different work teams were aligned in the collaborative manner fully explained using figure 2. As a consequence, the PLM tools were used to design the assembly process as a result of column 3. Column 3 is fully explained using figure 4 and section 4.1 to 4.3 following the case study selected.

It is important to mention that the figure 3 is the link through figures 1, 2 and 4. The KPR structure represented in Fig. 1 is built under an information and knowledge sharing framework. This framework is the basis to link the elements presented in Fig. 2. In addition, Fig. 4 considers this same framework under the knowledge sharing and PLM concept approach. In this way figures 1, 2 and 4 are incorporated in figure 3 to explain the collaborative engineering concept applied to the PLM environment.

Fig. 4 depicts the PLM tools integration used at this research work, using some of the real simulation results. Product design, layout planning, ergonomics and factory flow simulation were performed using DELMIA® and CATIA®. Information and knowledge sharing under PLM environment is shown supporting collaborative engineering design teams. In the next sections figure 4 is discussed in full detail.

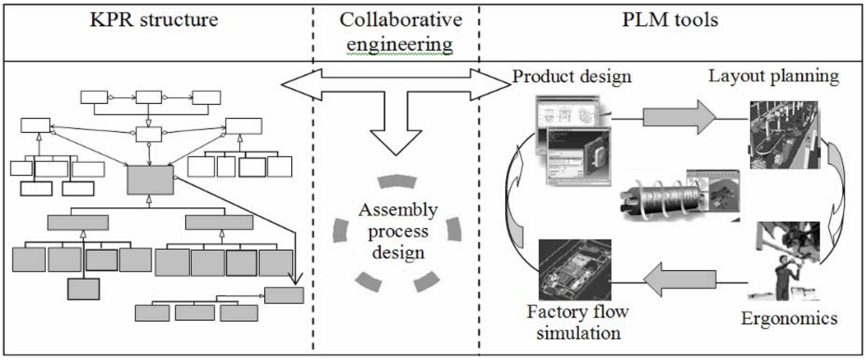

Fig. 3. KPR structures to support collaborative engineering at PLM environment (adapted from [13])

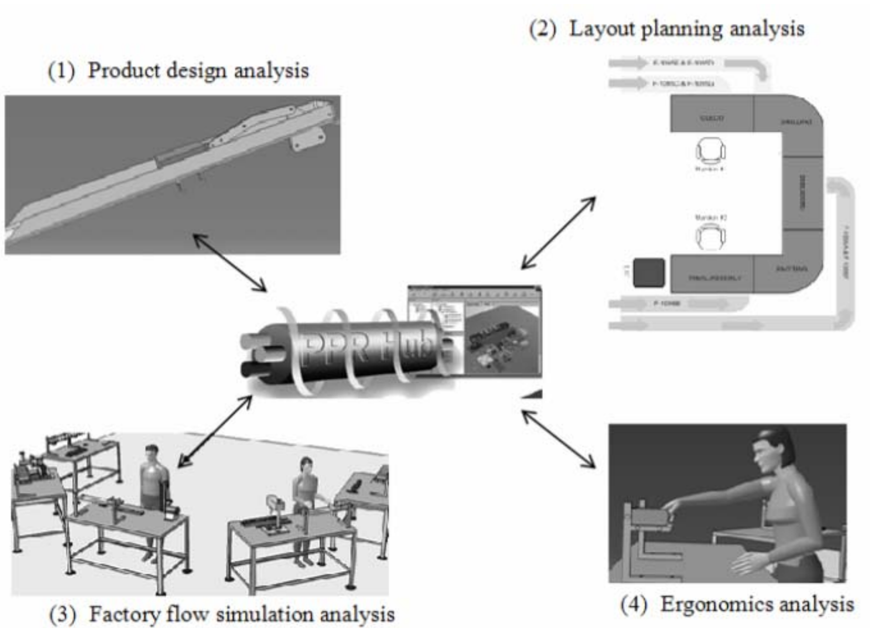

Fig. 4. Information and knowledge sharing at PLM environment

4.1 Product Design Analysis (#1 from Fig. 4)

The product design operations under collaborative work environments use and share a significant amount of information and knowledge that may not be validated, complete or certain [17]. This ambiguous information and knowledge at preliminary stages may imply a risk of misunderstanding if a proper information sharing network is not present. In this case study, for the Virtual Manufacturing team, the objective of the project was to design and simulate an assembly process for a specific empennage part. The preliminary work consisted in subdividing and grouping the required steps for the assembly into five different workstations. These stations were divided according to the main processes that were needed: riveting, drilling, de-burring, fixturing and assembly. After that, resources, processes and knowledge were defined for each station using the KPR structure from Fig. 1, through the collaborative work of experts consulted as well as the previous experiences of the members. These workstation classifications represent the main reference frame for both building a consistent knowledge infrastructure, and segregating the different tasks to perform parallel work and collaborative engineering.

4.2 Layout Planning Analysis (#2 from Fig. 4)

The next step was to propose different layouts to arrange the different workstations, considering the work sequence, the timings, the material flows and the value-adding processes. For this task, the PLM digital tools (Factory Flow simulation, for instance) were significantly useful since some of the modules are designed to perform these specific activities, automatically generating and sharing explicit knowledge and information i.e. the results of the simulations. The layout alternatives were then analyzed by the team members to decide which one was the optimal choice, using the time cycle as a primary criterion.

4.3 Ergonomic Analysis

The final effort of the project was to use virtual manufacturing tools to build a simulation for the assembly process with the layout planned before. The simulation was not only intended to serve as a data-show, but also to provide value-added contained in the information and knowledge generated during the project, regarding the collective expertise created along the project realization. Due to the complexity of the traditional assembly procedure, the main contributions of the simulation part of the project for ICKTAR were: time studies, process optimization and special focus on critical steps of the assembly process for quality considerations. The company had special interest in obtaining this valuable information and knowledge related to the assembly process. A snapshot of this simulation is presented in Fig. 4 (#3 and #4). Since two operators were working simultaneously, line balancing procedures had to be performed. Industrial engineering concepts, such as line balancing and work sequencing, were also applied by the members for optimization procedures.

5 Discussion, Conclusions and Further Research

The collaboration between ICKTAR and ITESM continues to produce significant results both academically and professionally. Nowadays, new design teams have joined the effort under the same structure stated when the project started. Relevant conclusions arise from the contribution of the Virtual Manufacturing team in terms of facilities planning and assembly design. This case study is still producing enough research issues regarding novel concepts in manufacture and design.

This multi-disciplinary environment, together with a collaborative engineering effort enhanced considerably the knowledge flow between coworkers. Teamwork and commitment represent a key factor in developing multi-disciplinary tasks, due to the main purpose is to avoid work repetition and focus on parallel work.

Important advantages of using knowledge and information structures under PLM environment to support collaborative engineering efforts have been presented. These advantages point to increasing the efficiency of the design process by enhancing information and knowledge flows between multi-disciplinary elements, hence allowing collective expertise growth. The case presented is an example of the application of the KPR structure to support an assembly process design work under collaborative engineering strategy. The application of this structure is wide and not limited to a manufacturing environment.

During the realization of this project, the lack of a knowledge management system oriented to all the phases of the product development became evident. PLM tools provides good collaborative work environment using product data management application dealing mainly with information management. Then it is required a new Knowledge Management (KM) methods and techniques to support KM throughout PLM. This paper presented a KPR structures as KM technique to solve this problem.

This paper presented a KPR structure supporting a collaborative work environment by hand using a notebook (KM conceptual model application). However, a KM software will be produced as future work and connected at PLM using the conceptual model application. The development of such system is subject of further work for the research team.

Acknowledgements. The authors acknowledge the support received by the Instituto Tecnológico y de Estudios Superiores de Monterrey (ITESM) through the Research Chairs in Autotronics. Additional support for this work has been provided by the IBM SUR GRANT. The authors acknowledge the support that ITESM manufacturing engineering team students have provided, especially Luc Vuichar and Javier Garcia participation at ICKTAR project. The authors also acknowledge the valuable support of: Samantha Rodriguez, Eduardo Gonzalez Mendivil, Pedro Orta, and Hugo Elizalde.

References

1. Gao, J.X., Aziz, H., Maropoulos, P.G, Cheung, W.M. Application of product data management technologies for enterprise integration. International Journal of Computer Integrated Manufacturing. (2003) 16: 491-500.
2. Young, R.I.M., Cutting-Decelle, A., Guerra-Zubiaga, D., Gunendran, G., Das, B., Cochran, S. Sharing Manufacturing Information and Knowledge in Design Decision Support. Advanced Integrated Design and Manufacturing in Mechanical Engineering. (2005), 173 – 185.
3. Karagiannis, D., Apostolou, A. Knowledge Management in Eco-tourism: a case study. Lecture Notes in Artificial Intelligence 3336. PAKM 2004. (2004) 508 – 521.
4. Guerra-Zubiaga, D., Young, R. A Manufacturing Model to Enable Knowledge Maintenance in Decision Support Systems, 33th Annual Conference of North American Manufacturing Research Institution of Society of Manufacturing Engineering (NAMRI/SME).(2005) 33-1: 203-210.

5. Olsen, R., Cutkosky, M., Tenenbaum, J.M., Gruber, T.R. Collaborative Engineering based on Knowledge Sharing Agreements. Proceedings of the 1994 ASME Database Symposium. (1994).
6. Rose, B., Robin, V., Lombard, M., Girard, P. Use of collaborative knowledge and formalized exchanges to manage collaborative design environment. PLM: Emerging solutions and challenges for Global Networked Enterprise. (2005), 67-77.
7. Bouras, A., Gurumoorthy, B., Sudarsan, R. Editorial. In: Bouras, A., Gurumoorthy, B., Sudarsan, R. (eds): Product Life Cycle Management, Emerging Solutions and Challenges for Global Networked Enterprise. Inderscience Enterprises Limited. (2005)221-230ï
8. Peñaranda, N., Molina, A., Aca, J., GuerraZubiaga, D. Experiences in implementing ALM Tools: An action Research Approach. ïïIn: Bouras, A., Gurumoorthy, B., Sudarsan, R. (eds): Product Life Cycle Management, Emerging Solutions and Challenges for Global Networked Enterprise. Inderscience Enterprises Limited. (2005)221-230ï
9. Guerra-Zubiaga, D. A manufacturing model to enable knowledge maintenance in decision support systems. Ph.D. Thesis, Loughborough University. (2004)
10. Harding, J. A. A knowledge representation to support concurrent engineering team working by Jennifer Anne Harding. Thesis, Loughborough University. (1996)
11. Zhao, J., Cheung, W.M., Young, R.I.M. The influence of manufacturing information models on product development systems. Engineering Design Conference, UK. (2000).
12. Molina, A., Bell, R. A manufacturing model representation of a flexible manufacturing facility. Journal of Engineering Manufacture. (1999), 213(B3): 225.
13. Guerra-Zubiaga, D., Rosas, R., Camacho, R., Molina, A. Information Models to Support Reconfigurable Manufacturing System Design. In: Bouras, A., Gurumoorthy, B., Sudarsan, R. (eds): Product Life Cycle Management, Emerging Solutions and Challenges for Global Networked Enterprise. Inderscience Enterprises Limited. (2005) 55 – 63.
14. Ozawa Masanori., Cutkosky Mark R and J. Howely Brian. Model Sharing among Agents in a Concurrent Product Development Team. (1998).
15. Young, R.I.M., Dorador, J.M., Zhao, J., Wang, S., Costa, C.A. A Shared Information/ Knowledge Environment to Support Concurrent Engineering. International Conference on Concurrent Engineering Research and Applications, USA. (2001).
16. Elizalde, H., Ramirez, R., Orta, P., Guerra, D., Perez, Y. An Educational Framework for Learning Engineering Design through Reverse Engineering, Engineering, Sixth international workshop on Active Learning in Engineering Education, Mexico. (2006), 344 - 365.
17. Grebici, K., Blanco, E., Rieu, D. Framework for Managing Preliminary Information in Collaborative Design Processes. In: Bouras, A., Gurumoorthy, B., Sudarsan, R. (eds): Product Life Cycle Management, Emerging Solutions and Challenges for Global Networked Enterprise. Inderscience Enterprises Limited. (2005) 90-100.

Knowledge Management Systems and Organizational Change Management: The Case of Siemens ShareNet

Hauke Heier[1] and Susanne Strahringer[2]

[1] Accenture, Strategic IT Effectiveness (SITE)
Kaistrasse 20, 40221 Duesseldorf, Germany
hauke.heier@accenture.com
[2] European Business School, Information Systems 1
Schloss Reichartshausen, 65375 Oestrich-Winkel, Germany
susanne.strahringer@ebs.edu

Abstract. Earlier research has shown the interdependence of Knowledge Management Systems' (KMS) implementations and corporate cultures. This study builds on and extends this research by exploring how knowledge-intensive corporate cultures can be established. Employing a case study approach, we analyze the (successful) implementation of Siemens Information and Communication Networks (ICN) group division's ShareNet, a global knowledge sharing network for the sales and marketing community. We bring together the overlapping disciplines and practices of (culture) change management and knowledge management (KM) by testing the explanatory power of Lewin-Schein's well-known and widely cited change paradigm [18, 28, 29] in the new setting of KMS. Our case analysis demonstrates the linear, staged theoretical framework's strength in understanding and explaining aspects of success and failure in KMS implementation projects.

Keywords: Knowledge Management, Knowledge Management Systems, Change Management, Corporate Culture, Organizational Change, Information Systems Implementation.

1 Introduction

Corporate culture is widely acknowledged as a make-or-break factor of Knowledge Management Systems (KMS) implementations, which differ from most information systems (IS) projects due to greater difficulties associated with managing human factors and effectively changing the corporate culture [4, 16, 27]. Academics and managers begin to recognize that knowledge processes, either deliberate or emergent, exist in duality with a context composed of strategy, structure, people/culture, and technology [8]. Up to now, most IS implementation studies focus on critical success factors such as top management support, commitment, user involvement, and user training, but do not explicitly acknowledge or incorporate the impact of corporate culture.

Following academic suggestions for combining change management/organizational behavior theories with IS research [6, 33] - as well as with knowledge management

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 97 – 105, 2006.

(KM) investigations [30, 35] - this study applies a linear, staged change paradigm to the new setting of KMS. Goals are to test the theory's explanatory power and to explore its helpfulness in establishing knowledge-intensive corporate cultures[1] [13]. We intend to learn about the timing and sequencing of suitable change interventions. This study is situated at the nexus of research on KMS, (culture) change management, and IS implementation projects; hence it acknowledges and builds on a variety of literatures.

Earlier process studies on success factors in IS and KMS implementation projects [e.g. 12, 30] typically focused on the well-known Lewin-Schein change paradigm [18, 28, 29]. We selected the theoretical framework for this study since it focuses on culture change, serves as a means of integrating and combining material from other process studies [1], and its application to the new setting of KMS has explicitly been suggested in earlier studies [30]. While this linear, staged change paradigm has attracted criticism for drawing a rather static picture of corporate culture [e.g. 21, 25] we are confident that the understanding of KMS implementation projects will benefit from its a priori explanations on an individual, group, and organizational level.

2 Conceptual Foundations

Lewin-Schein's theoretical framework, widely acknowledged both in descriptive studies [e.g. 12, 41] and prescriptive analysis [e.g. 20, 38], views organizations as moving from an old stage of quasi-stationary equilibrium to a new one. If change is to happen and be permanent, i.e. to achieve good implementation results, midlife or mature organizations have to pass through all three stages presented in Table 1. The corresponding sub stages highlight a sequence of critical success factors and prior explanations of success and failure for IS and KMS implementation projects [12]. The model's predictions will be used for later theory-testing or time-series analysis [40, 15].

In the first stage - "unfreezing" - people become aware of the need for culture change and see the necessity to deviate from prior behavior patterns and mind-sets [32]. They must continuously refine their organization's knowledge-based core capabilities which would otherwise result in core rigidities or cultural inertia [17, 36]. Three intertwined processes create motivation to change: first, disconfirming data causing serious discomfort and disequilibrium to the organization; second, a connection between disconfirming data and important ideals leading to anxiety or guilt; and third, sufficient psychological safety, i.e. sense of identity and integrity to explore new opportunities [28].

Disconfirmation alone does not produce motivation to change, as long as it is not linked to important objectives and ideals. The same holds true for expensive KMS implementation projects: these get support when they are somehow related to business objectives, e.g. cost savings, product/service innovations, and time to market [4, 39]. Our case discussion will explore disconfirmation and business purpose linkage factors (respectively denoted by "DIS" and "BPL") and their application to create survival

---

[1] We define knowledge-intensive corporate cultures broadly as those cultures where employees value learning and show a positive knowledge-orientation: they are free and willing to explore, are proactively seeking and offering knowledge, and their activities are given credence by managers [4, 23].

Table 1. A model of transformative change

Stage One Unfreezing: creating the motivation to change	• Disconfirmation • Creation of survival anxiety or guilt • Creation of psychological safety to overcome learning anxiety
Stage Two Cognitive restructuring: learning new concepts and new meanings for old concepts	• Imitation of and identification with role models • Scanning for solutions and trial-and-error learning
Stage Three Refreezing: internalizing new concepts and meanings	• Incorporation into self-concept and identity • Incorporation into ongoing relationships

anxiety. Further emphasis is placed on the predictions for fostering psychological safety which match well with success factors commonly established by the innovation, organizational behavior, and IS implementation literature (e.g. [7, 14, 34]: top management support, commitment, user involvement, and user training (respectively denoted by "TMS", "COM", "UIN", and "UTR").

The second stage - "cognitive restructuring" - implies moving the organization to a new quasi-stationary equilibrium. It involves the gathering of new information, adopting new behavior, and experiencing the new behavior's contribution to improved performance. At this point, the examination of existing IS and the development of new IS get underway [32]. There are two different paths to proceed: "example setting", i.e. individuals, organizational units, and pilot studies, is an effective means for communication and building stakeholder commitment [e.g. 31]; "trial-and-error learning" promises a better fit with the learner's personality [29]. For KM McElroy suggests interventions which give rise to desired behaviors instead of commanding them [22]. The case analysis will examine the change path taken (respectively labeled by "ESE" and "TAE").

Finally in the third stage - "refreezing" - the new behavior and set of underlying assumptions are solidified to bring back the organization to stability [14]. Change agents must work actively with the organization's members to ensure the new IS' installation, debugging, usage, measuring, and enhancements, i.e. embedding the KMS and a knowledge-intensive corporate culture into the organizational context. Users should finally assume ownership for maintenance and evolution [11]. "Refreezing" comprises the following major activities: first, the determination of evidence for behavioral and cultural shifts, i.e. technology's fit with people/culture; second, the IS' organizational integration, i.e. technology's fit with strategy and structure [8, 11]; third, the dissemination of confirming data to conclude the change process (the factors are respectively denoted by "BCD", "OIN", and "CON").

3 Research Methodology

We argue that an exploratory case study is appropriate to examine the change paradigms' explanatory power and goodness of fit, since the boundaries between phenomenon and context are not clearly evident in KMS implementations [40]. Case studies lend themselves well to IS research because the technology is relatively new and researchers begin to shift their interest to organizational rather than technical issues [2]. We chose a positivist epistemology, assuming that reality is objectively given and can be described by measurable properties which are independent of the researcher and the instruments [3].

While some authors recommend interpretive research for studying IS implementations and corporate cultures - especially for the deeper levels of values and assumptions [9, 24] - we believe in positivisms applicability to case study research [5, 15] and its suitability for our study which focuses on observable change interventions and on overt attributes of knowledge-intensive corporate cultures. We argue that a single case is appropriate: ShareNet represents a critical case with all necessary conditions for theory-testing; additionally, Siemens' Information and Communication Networks' (ICN) successful, culturally-sensitive rollout is a polar and unique case [3, 40]. The KMS implementation project combines KM and change initiatives and offers multiple embedded units of analysis, i.e. business units and regional sales organizations [see 10 for details].

3.1 Site Selection and Unit of Analysis

Our research site was the Siemens ICN group division, headquartered in Munich, Germany. Its activities include the development, manufacturing, and sale of public communication systems, private business communication systems and related software, and the provision of a variety of consulting and maintenance services. In a more and more tense market, characterized by deregulation, increasing complexity of the business, and disintegration of traditional value chains, Siemens ICN's sales of EUR 7.1 billion resulted in a negative EBIT of EUR 366 million in fiscal 2003. The group division has several business units with independent profit responsibility and regional sales organizations (local companies) around the globe.

To support the strategy of becoming a global solution provider where the business units provide the physical components of a sales project and regional sales organizations are responsible for the customization and integration, the importance of international cooperation and knowledge management was realized in 1998. A project team was established to facilitate a culture shift towards knowledge sharing vs. knowledge hoarding and proactively seeking and offering knowledge. Our case study's unit of analysis is the resulting implementation of ShareNet, a global knowledge sharing network for Siemens ICN's sales and marketing community with 18,000 registered users. The project analysis pays explicit attention to the factors introduced in the conceptual foundations.

For the sharing of explicit knowledge, ShareNet's aim was to provide structured solution objects (e.g. technical or functional solution knowledge) and environment objects (e.g. customer or market knowledge) in a knowledge repository. For the exchange of tacit knowledge, the global cultivation and support of personal collaboration was enhanced with urgent requests, discussion forums, news, and chat.

3.2 Data Collection

We collected empirical evidence for this study at different levels and through a variety of methods: semi-structured interviews (lasting for two hours on average), direct observations, and reviews of corporate documents. The triangulation of data collection techniques provided multiple perspectives for the issues under study and allowed for a cross-checking of existing and emerging concepts [26]. During two three-month field visits in the summers of 2002 and 2003, we interviewed a cross-section of the Siemens ICN employees responsible for ShareNet's implementation and maintenance: top management sponsors, internal consultants, IT support staff, global editors, user champions, and users. Of special interest was information about change interventions and a variety of qualitative and quantitative metrics that the firm uses to regularly assess the KMS' usage and financial viability. Success was further determined with accepted measures for non-mandatory IS: user satisfaction, objectives accomplished, and external recognition [4, 19].

4 Case Discussion

Data examination, i.e. time-series analysis [40], confirmed the theoretical framework's goodness of fit with the case study [see 10 for details]. The application of the Lewin-Schein change paradigm highlights a sequence of critical success factors/change interventions which influence the outcome of KMS implementations. It also confirms earlier findings from IS implementation research: top management support ("TMS"), user involvement ("UIN"), and user training ("UTR") remain critical success factors in the setting of KMS. While though those factors matched well with Schein's [28, 29] and Zand & Sorensen's [41] predictions for fostering psychological safety, they were not restricted to "unfreezing" but ran across most implementation stages. For commitment ("COM"), only limited indications were found. Case study data suggests that commitment factors are no true interventions, but rather the result of successful "unfreezing". They result from a "felt need" or deliberate actions but are no direct change management measures.

Throughout ShareNet's implementation, top management provided project supervision, allocated funding for an adequate (technology) infrastructure, and communicated the need for a long-term change management initiative [12, 13]. A lack of executive support at headquarters and at some regional sales organizations hampered local rollouts. To maintain a high level of user involvement, 40 sales representatives augmented the ShareNet project team. Key activities were mapping Siemens ICN's sales process with the required knowledge for each step, defining a data model and quality assurance process, prototype testing, and proposing a consistent organizational structure. As a result, sense of ownership and the assessment of system requirement's quality increased.

Since KMS use is generally not mandatory, we also postulate that the communication of disconfirmation ("DIS") - combined with business objectives ("BPL") - provides the necessary impetus for change [comp. 4, 34]. Both factors were mainly used for the project's start, congruent with the theory. Since most sales and marketing knowledge resided in regional sales organizations, those employees were the first to experience disconfirmation through a changing business environment, to recognize the need for a KM initiative. Their suggestions were taken up by the KMS project team. Several brainstorming sessions, a formalized training concept, and a vision statement made the interrelationship between KM and economic objectives clear: "ICN ShareNet is the global knowledge sharing network. We leverage our local innovations globally. We are a community committed to increase value for our customers and ICN by creating and re-applying leading-edge solutions."

For "cognitive restructuring", the Lewin-Schein interventions provided less IS implementation-specific guidance. We postulate that "trial-and-error" learning ("TAE") has minor applicability for KMS implementation projects. Considering IS implementation a deliberate, planned process of "preparing an organization for a new system and introducing the system in such a way as to assure its successful use" [37] defeats the very nature of trial-and-error learning. Interventions to set positive examples ("ESE") were more widely used throughout all implementation stages: e.g. pilot studies, success stories, and user champions. Dedicated executives and success stories describing successful cases of collaboration created KM demand throughout the regional sales organizations and first commitment to change. A user satisfaction survey in 2000 confirmed the user champion's status as role models: users remarked that they were "vital in promoting ShareNet and its workshops". Example setting factors therefore qualify as significant change enablers for combined KM and change initiatives or communities-of-practice (CoPs).

The main "refreezing" activities, i.e. the determination of behavioral und cultural shifts ("BCD"), the KMS' organizational integration ("OIN"), and the dissemination of confirming data ("CON"), were all found at the research site. On conclusion of the first three project stages we saw a transfer of ShareNet's ownership from the project team to a regular line organization and even a further delegation of maintenance and evolution responsibilities to smaller CoPs. In parallel, the ShareNet team continually monitored behavioral and cultural change based on predefined measures and personal experiences, e.g. with user satisfaction surveys and an observation of voluntary actions. After some years, the majority of employees acknowledged the need for global knowledge leverage, exchanged content without ever making personal contact, and policed the network for low quality postings. When those first results from culture change showed, top executives started to dispense confirmatory statements and feedback. Heinrich von Pierer - Siemens CEO and chairman - particularly approved of the accomplishments achieved when he declared ShareNet's technical platform and phased rollout approach a Siemens-wide KMS best practice.

5 Conclusions and Implications for Further Research

Our exploratory design allowed for a combination of discovery and testing [3]: time-series analysis [40] confirmed the applicability of Lewin-Schein's theoretical framework while further causal relations and interferences emerged. Case study data reveals indications for the isolated conceptual variables/factors, introduced in the conceptual foundations. With the exception of user involvement and factors from the change paradigm's second stage, all show the predicted, different pattern over time. In the "unfreezing" stage, disconfirmation and business purpose linkage factors built a foundation for the "traditional" critical success factors top management support, commitment, user involvement, and user training. The "cognitive restructuring" stage provided less explanations for IS implementations: trial-and-error factors had limited significance, while (personal and impersonal) example setting factors were only vaguely prescriptive. Finally, in the “refreezing” stage, organizational integration and behavioral and cultural diagnosis factors paved the ground for concluding confirmation factors.

However, an emerging area like KMS implementation research - potentially affecting the KM context elements strategy, structure, people/culture, and technology - is characterized by many unresolved issues. Our research is an initial attempt to identify and organize the most important ones [8, 13]. The conclusions set forth are not derived from a generally accepted theory, but are pieced together from research on culture change, IS implementation, and KMS. They integrate several widely-accepted change interventions and common practice. Of course, this study’s tentative results require additional substantiation and tests before they could gain general acceptance. Generalization from theory to different settings will need further confirmation from additional case studies [15].

References

1. Armenakis, A. A., Bedeian, A. G.: Organizational change: a review of theory and research in the 1990s. Journal of Management 25 (1999) 3 293-315
2. Benbasat, I., Goldstein, D. K., Mead, M.: The case research strategy in studies of information systems. MIS Quarterly 11 (1987) 3 369-386
3. Darke, P., Shanks, G., Broadbent, M.: Successfully completing case study research: combining rigour, relevance and pragmatism. Information Systems Journal 8 (1998) 4 273-289
4. Davenport, T. H., De Long, D. W., Beers, M. C.: Successful knowledge management projects. Sloan Management Review 39 (1998) 2 43-57
5. Dubé, L., Paré, G.: Rigor in information systems positivist case research. MIS Quarterly 27 (2003) 4 597-635
6. Galliers, R. D.: Towards the integration of e-business, knowledge management and policy considerations within an information systems strategy framework. Journal of Strategic Information Systems 8 (1999) 3 229-234
7. Ginzberg, M. J.: Key recurrent issues in the MIS implementation process. MIS Quarterly 5 (1981) 2 47-59
8. Grover, V., Davenport, T. H.: General perspectives on knowledge management: fostering a research agenda. Journal of Management Information Systems 18 (2001) 1 5-21
9. Hatch, M. J.: They dynamics of organizational culture. Academy of Management Review 18 (1993) 4 657-693
10. Heier, H., Borgman H. P., Manuth, A.: Siemens: expanding the knowledge management system ShareNet to research and development: Siemens: expanding the knowledge management system ShareNet to research & development. Journal of Cases on Information Technology 7 (2005) 1 92-110
11. Jin, K. G., Franz, C. R.: Obstacle coping during systems implementation. Information & Management 11 (1986) 2 65-75
12. Keen, P. G. W., Bronsema, G. S., Zuboff, S.: Implementing common systems: one organization's experience. Systems, Objectives, Solutions 2 (1982) 2 125-142
13. King, W. R., Marks, P. V., McCoy, S.: The most important issues in knowledge management: what can KM do for corporate memory, management thinking, and IS responsibility, as well as for overall business performance? Communications of the ACM 45 (2002) 9 93-97

14. Krovi, R.: Identifying the causes of resistance to IS implementation: a change theory perspective. Information & Management 25 (1993) 6 327-335
15. Lee, A. S.: A scientific methodology for MIS case studies. MIS Quarterly 13 (1989) 1 33-52
16. Leidner, D.: Information technology and organizational culture. In: Galliers, R. D., Leidner, D. E., Baker, B. S. H. (eds.): Strategic information management: challenges and strategies in managing information systems, 2nd ed. Butterworth-Heinemann, Oxford (1999) 523-550
17. Leonard, D.: Wellsprings of knowledge: building and sustaining the sources of innovation. Harvard Business School Press, Boston (1998)
18. Lewin, K.: Group decision and social change. In: Newcomb, T. N., Hartley, E. L. (eds.): Readings in social psychology. Holt, Rinehart & Winston, Troy (1947) 329-340
19. Lucas, H. C.: Implementation: the key to successful information systems. Columbia University Press, New York Guildford (1981)
20. Lucas, H. C., Plimpton, R. B.: Technological consulting in a grass roots action oriented organization. Sloan Management Review 14 (1972) 1 17-36
21. Macredie, R. D., Sandom, C.: IT-enabled change: evaluating an improvisational perspective. European Journal of Information Systems 8 (1999) 4 247-259
22. McElroy, M. W.: Using knowledge management to sustain innovation: moving toward second generation knowledge management. Knowledge Management Review 3 (2000) 4 34-37
23. Meso, P., Smith, R.: A resource-based view of organizational knowledge management systems. Journal of Knowledge Management 4 (2000) 3 224-234
24. Myers, M. D.: Qualitative research in information systems. MIS Quarterly 21 (1997) 2 241-242
25. Orlikowski, W. J., Hofman, J. D.: An improvisational model for change management: the case of groupware technologies. Sloan Management Review 38 (1997) 2 11-21
26. Pettigrew, A. M.: Longitudinal field research on change: theory and practices. Organization Science 1 (1990) 3 267-292
27. Ruggles, R.: The state of notion: knowledge management in practice. California Management Review 40 (1998) 3 80-89
28. Schein, E. H.: Organizational culture and leadership, 2nd ed. Jossey-Bass, San Francisco (1992)
29. Schein, E. H.: The corporate culture survival guide: sense and nonsense about culture change. Jossey-Bass, San Francisco (1999)
30. Seeley, C.: Change management: a base for knowledge-sharing. Knowledge Management Review 3 (2000) 4 24-29
31. Spector, B.: From bogged down to fired up: inspiring organizational change. Sloan Management Review 30 (1989) 4 29-34
32. Srinivasan, A., Davis, J. G.: A reassessment of implementation process models. Interfaces 17 (1987) 3 64-71
33. Stein, E. W., Zwass, V.: Actualizing organizational memory with information systems. Information Systems Research 6 (1995) 2 85-117
34. Storey, J., Barnett, E.: Knowledge management initiatives: learning from failure. Journal of Knowledge Management 4 (2000) 2 145-156
35. Thomas, J. C., Kellogg, W. A., Erickson, T.: The knowledge management puzzle: human and social factors in knowledge management. IBM Systems Journal 40 (2001) 4 863-884
36. Tushman, M. L., O'Reilly, C. A.: Ambidextrous organizations: managing evolutionary and revolutionary change. California Management Review 38 (1996) 4 8-17

37. Tyran, C. K., George, J. F.: The implementation of expert systems: a survey of successful implementations. Database 24 (1993) 1 5-15
38. Urban, G. L.: Building models for decision makers. Interfaces 4 (1974) 3 1-11
39. Wilson, R.: Don't make culture another item on the KM checklist. Knowledge Management Review 3 (2000) 4 8-9
40. Yin, R. K.: Case study research: design and methods, 2nd ed. Sage Publications, Thousand Oaks (1994)
41. Zand, D. E., Sorensen, R. E.: Theory of change and the effective use of management science. Administrative Science Quarterly 20 (1975) 4 532-545

Measuring Business Feedback Cycles as Enhancement of the Support Knowledge Engineering Process

Alexander Holland and Madjid Fathi

Institute of Knowledge Based Systems and Knowledge Management, Department of Electrical Engineering and Computer Science, Hoelderlinstraße 3, 57068 Siegen, Germany
{alex, fathi}@informatik.uni-siegen.de
http://www.ws.fb12.uni-siegen.de/

Abstract. Supporting business processes through knowledge management technologies is one of the key factors in the today's industry. The former technologically related organizations now have to orient themselves to the business strategy of the company, which they must support with their offered IT services. IT processes has to become more transparent and better manageable. It is fundamental to visualize service prosperities to satisfy customer needs. Different business processes and the potentials of supporting them need knowledge management measures. We illustrate the applicability and possibilities of IT service management and appropriate support services in business processes especial in the field of incident management requiring a novel feedback cycle and associated measures to assist such service support processes. A prototypical realization was implemented as manageable knowledge desk.

1 Introduction

New digital technology is improving business efficiency by radically increasing the quality and quantity of information available to knowledge workers. Support center knowledge management strategies will help customers and employees to decide which new knowledge management technologies are relevant to their business and show how to use them to build an integrated knowledge management solution. Response or customer support centers constitutes a set of resources (personnel, computers, telecom-munication equipment) which enable the delivery of services via the telephone, email or web portal access. Customers make support contracts with different service levels [3]. The highest urgency level for instance guarantees that the problem is solved within a special time frame based on the contract service level agreement. Some problems require not only the attention of experts in the support organization but replacement parts and field engineers. A support center [6] coordinates these different steps: understanding what the customer problem is about and lastly determining the necessary actions to solve the problem. As an interesting alternative to placing a service call to the support center, customers can also turn to the electronic customer support center which provides a single entry point to comprehensive, customized support information and tools. From the support center side it is also necessary to share best practices in the support process. Networks can be

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 106 – 118, 2006.

build if some sort of catalogue of knowledge (knowledge bank) exists that can be used to connect people [17]. Best practices were found in the four main areas support, sales, workflow and project management. This means in a condensed form a structured knowledge [1] and information creation [16] process project for customer support center. Effective knowledge creation, maintenance and sharing becomes more and more a crucial factor for support organizations and especially for the support business [14].

2 IT Service Management and Service Operations

IT Service Management (ITSM) focuses on delivering and supporting IT services that are appropriate to the business requirements of the organization, and it achieves this by leveraging IT Infrastructure Library (ITIL)-based best practices that promote business effectiveness and efficiency.

2.1 HP ITSM Reference Model

The ultimative goal of ITSM is to provide quality services to customers, HP has organized IT processes into five different groups that focus on different aspects of the service lifecycle. This approach allows users to follow a complete service lifecycle [7]. Starting at the upper left and proceeding clockwise around the model (see figure 1), users can follow the progress of an IT service from initial conception through delivery, eventual obsolescence, and updating or replacement by a new service.

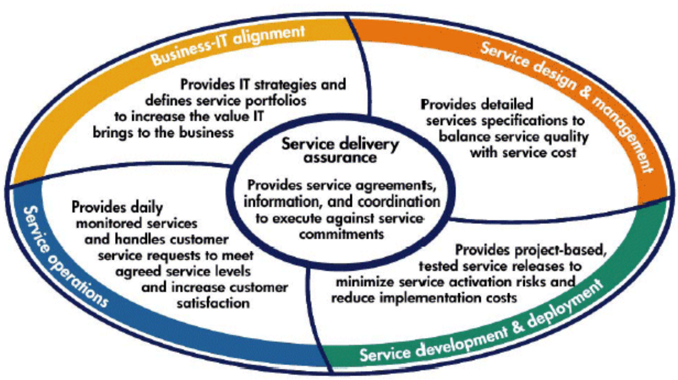

Fig. 1. HP ITSM reference model structure including five different groups

2.2 Customer Call Management Process

Incident and service request management is used by the support center to quickly restore service availability, minimize service disruptions and respond to customer needs. Reactive in nature, it's activities focus on handling incidents in the infrastructure or those reported by users via efficient first-, second- or third-level support services as well as responding to service related incidents. It also deals with requests for information and other types of administrative assistance. This process interacts frequently with change management and configuration management [15]. The following customer call management process within service operations processes describes the call flow inside organizational structures and the knowledge creation and retrieval action (see figure 2).

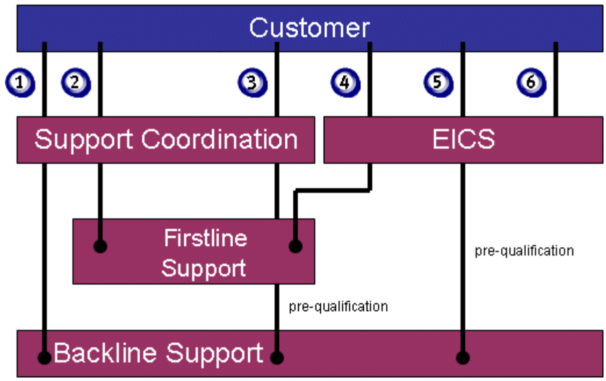

Fig. 2. Customer call flow and workflow with affected staff and pre-qualification steps

In the case of phone-in calls a call is logged in the support coordination (SC). Based upon the problem description and the contract type, the call is routed to the adequate support team Firstline (FL) or Backline (BL). The FL represents 1st level support and tries to solve most of the standard product requests (case 2) by using support knowledge tools such as knowledge databases, knowledge retrieval systems or by using knowledge trees, implemented in the knowledge desk solution explained below.

2.3 Domain Architecture and Involved Knowledge Persons

First of all and always in the center of the attention at support centers is the customer. For any project that is set up the first question will be: How does the customer profit from? Furthermore, there are the different support lines involved such as FL and BL domain architects, project managers and finally the support management. The ultimative goal is to increase the self-solving rate of a customer with the help of ElCS which is basically a knowledge desk solution. The FL is the least trained division. They represent the customer entry point the direct link to the customer. If they cannot help the customer with a problem they connect the pre-qualified call to the BL. They have two main tasks: The knowledge desk usage and the knowledge feedback illustrated below. The BL or often specialized product team consists of well trained specialists. In the context of knowledge desks a domain architect (DAR) setups the knowledge tree structure and creates the knowledge content. A DAR is also responsible for normal call handling. Each domain architect has his own domain for which he is liable. For a description of a domain see the next figure 3 below:

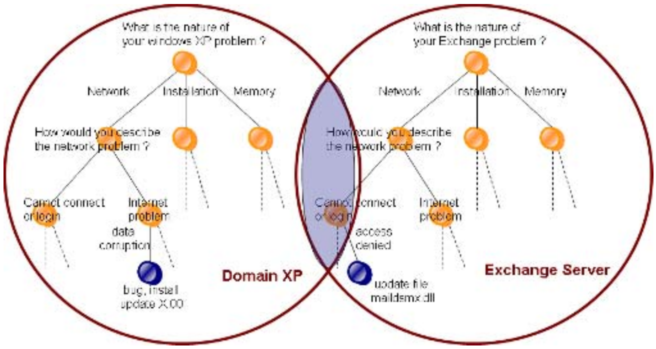

Fig. 3. Knowledge desk decision tree structure including two domains Windows XP and Exchange

Figure 3 shows the domains Windows XP and Exchange Server with the tree structure inside the circles. Yellow dots stands for questions and blue dot for corresponding answers. At the end of a tree should be an answer or at least a link to the solution documentation. Different domains can have common nodes or common answers expressed as intersection. Examples for a product team are black circles that represent different domains (see figure 4). The green circle represents the FL support.

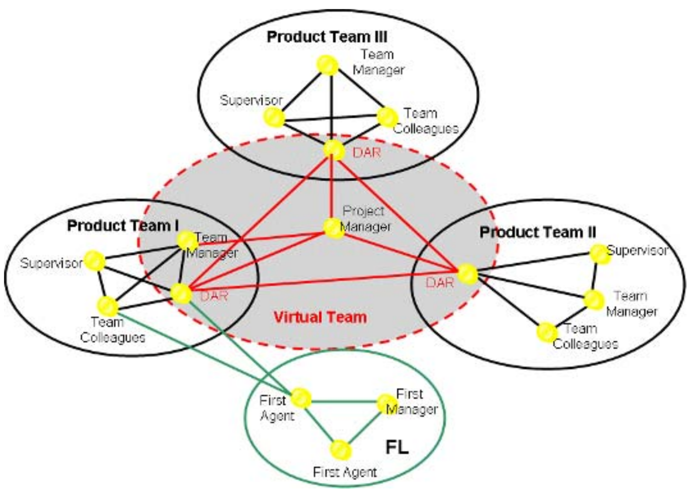

Fig. 4. Support team structure and domain architect integration as virtual team conception

3 Knowledge Desk

A Knowledge Desk (KD) is a system portal solution that can creates inside support organizations a cased-based reasoning and advisory system for guided retrieval, maintenance and creation of knowledge. KD is aimed to enable and encourage structured, seamless integration of process and solution information from a variety of data sources and legacy systems. It offers also a fast linking and learning decision tree system integration all users in the online process of content and structure maintenance. KD users are able to focus on handling the calls, while the knowledge tool takes care of the complexity of the systems, networks and processes for them. It is more than a knowledge tool. It is also a strategy for complete integration of the processes of knowledge retrieval and maintenance, and workflow management. It is an instrument for a fast learning organization as fast linking of knowledge about people, organizations, products, problems, customers and processes is key to success. The main target group for KD is the FL who are less trained and less educated than the product teams on the next level. They very often do not have a fundamental technical background which makes it so important that KD is easy to manage. Furthermore, it has a very fast response time and all necessary information and tools are integrated into one interface. It contains a powerful feedback process which transforms it to a learning and growing organism. KD also has the capability to add multiple attachments to an action or a question should the domain architect deem it necessary to provide additional information to the FL. These attachments can be executable and have the capability to launch external programs. These feedback process will be described in detail in chapter 4. The same knowledge and toolset can be directly used by external customers, enabling customer self-help and self-repair with the help of the ElCS. The key features of KD are knowledge retrieval, customer information knowledge bases, knowledge creation, online filter techniques, knowledge maintenance and online reporting to measure values. The most important advantages of KD summarized are inexpensive, better performance with a local server solution, built-in web capability, cut and paste possibilities, statistics on case usage and flexibility.

3.1 Knowledge Desk Architecture and Implementation

Figure 5 illustrates a screenshot of the knowledge desk. As can be seen top left, there I the tree structure with the last questions that where asked. The first question, the entry point is question for the experience level of the questioner. There the user has as basic three different possibilities, high, medium and low. The user can also visualize the tree structure using hyperbolic trees. A direct jump and link to a specific KD node is available via the syntax '*domain-name/question.x*'. If the customer clicks on a solution node he will get detailed solution information concerning the initial request and also several helpful links to other web pages or other tools. The FL uses the KD in the same way although there are different contents [10]. When he receives a call from a customer he asks for the problem and clicks on the relevant knowledge tree. Then, he scans through the questions and problems and asks the customer in the form of a dialogue for further information while clicking on the right answers and questions. At the end of the tree, there should be the answer to the request, a link to a cookbook as instruction set written by employees how to solve the problem or an or more attachments. These attachments are a method of providing the FL with further documentation. An executable attachment is capable of running an external application that can then provide the agent with additional information [5].

The knowledge desk implementation is based on Microsoft's .NET framework and the programming language C#.NET utilizing an object-oriented application and data structure [20]. The NET framework with its main components depicted in figure 6 is considered to be a set of software technologies for connecting information, people, systems and devices.

The NET.framework is based on a platform independent fundament (common language runtime). In addition to that, it provides a common language specification that

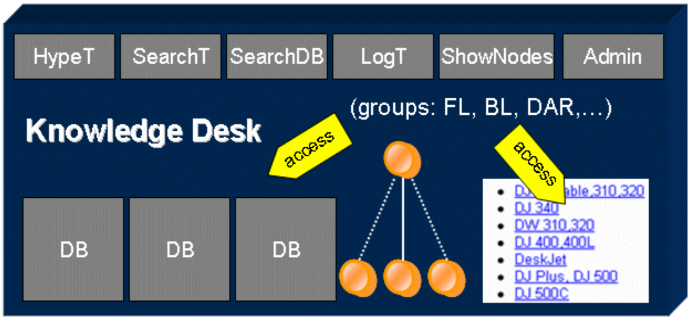

Fig. 5. Knowledge desk structure including the tree structure, user navigation panel, hyperbolic tree and node search

allows users for integrating existing or developing new programming languages on top of the .NET framework by implementing appropriate compilers. Thereby, programming concepts are available across .NET-based programming languages. A common base class library together with common services such as ADO.NET and concepts such as windows forms build a comprehensive infrastructure for the development of software applications. Because of the sophisticated requirements concerning the KD main user interface, the KD was implemented with C#.NET. KD

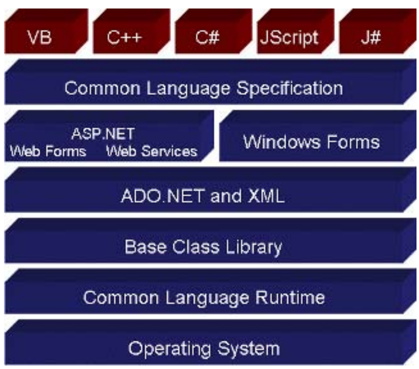

Fig. 6. The Microsoft .NET framwork

comprises a database or knowledge support repository server as enhancement for the BL users. It is a server infrastructure that provides simple access to available support knowledge with millions of documents such as cookbooks, instructions or comments. The support repository can provide an infrastructure to meet the changing business requirements for simple deployment of agents, technicians or 3rd party support providers as well as improve productivity of the existing labor force. It can also provides a secure server foundation to enable improved electronic customers access to solutions via electronic access to the ElCS. Consolidated the support knowledge repository provides a solid foundation for retrieval, management and storage of support knowledge, supports multiple languages, provides metrics and proposals to facilitate managing knowledge content, supports search and retrieval of rich content formats

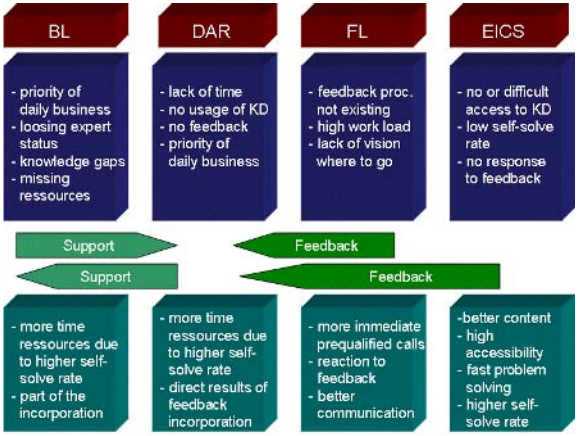

Fig. 7. Vision for the knowledge chain with observations and benefits for the customer support groups

and audio/video content in compound documents and captures metrics on content quality and usage by including user ratings and annotations.

We will next introduce a vision for the support knowledge chain including support and feedback processes to increase the benefit of involved support groups based on their observations.

For the BL figure 7 illustrates for example that they would loose their expert status if they put their knowledge into the trees. They are afraid to deliver wrong content and do simply not have the time resources to handle everything. The domain architects also have time problems and the daily business still priority. FL users complained about the too complicated feedback process and the high work load of the daily business. For the customer who is using the ElCS, it was sometimes difficult to access the KD and they were often older or without good content. In the bottom of figure 7 are the consequences from a regular usage of the KD and the feedback process. The BL teams would have more time to resolve the more difficult questions due to a higher self-solve rate of the customers themselves and of the FL.

4 Support Feedback Process

We will now describe the main tasks to implement the support feedback process. A feedback process is necessary to make a knowledge desk to a powerful tool. It also helps to grow together within a knowledge support community as support network. Furthermore, the feedback process has to be easy and simple to handle because otherwise no one will use it again [11]. Below, we will now describe the whole proceeding of the feedback process. A feedback process and its use have to be communicated to employees to improve acceptance and to be daily business. The access to feedback has to be more problem oriented [12], otherwise support users feel feedback as useless. A must is the common understanding of the various definitions, a clear communication flow, a fast processing of the feedback and fast reaction dependent on the priority, obvious and clear structure of the feedback sheet to support the easy handling, a well known owner of the feedback for further contact, definition of priorities and a mechanism to easy analyze feedback tables for statistic issues. It should also be possible to give general feedback which is not related to the KD or specific node IDs. In this case, it makes sense to develop a feedback process in a support organization n collaboration with the involved people: FL, domain architect, BL and project manager. To allocate feedback the user pre-classifies his feedback to facilitate the processing step. The user differentiates between feedback types, the rating and priority of the specific feedback request. A feedback type can be info, structure, semantic or syntax related. The rating can be good or bad, the priority normal or critical (Fig. 8).

Fig. 8. New support feedback form including type related routing

The feedback requester can initiate a call back for details from the associated support group to clarify some vagueness or demand an email to get the information that the feedback is established and in the closed status. This will be done automatically by the system. Every author of a critical feedback will automatically get this email when the status changes. This ensures that the feedback user himself gets an information about the ongoing process. In order to get feedback about the feedback process, a questionnaire was developed that can easily be accessed by the users via internet. The statistical result presents information concerning how often the users use KD and their satisfaction degree, how often they provide feedback, which other support tools they use (e.g. cookbooks, colleagues, own notes, miscellaneous), general potential of the KD-feedback process, commitment and collaboration of the support management, judge the handling of the feedback process or estimate the importance of the feedback process for their own benefit.

Figure 9 illustrates the support knowledge flow with the embedded customer feedback. After the user submits a new feedback in the support queue (status assigned) the support responsible [13] take over the case (status open) and follow up. Answered

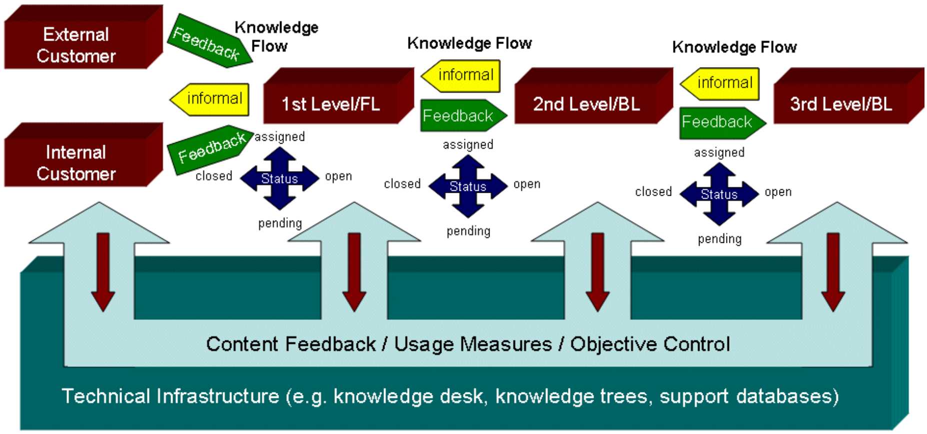

Fig. 9. Support knowledge flow with feedback states and the underlying technical infrastructure framework

requests have a specific waiting time (based on the priority) before closing. The user has the possibility to check the feedback status online everytime. The content feedback influences the technical infrastructure to grow to an underlying powerful support tool. After the feedback process was implemented it had to be communicated to the different groups and divisions to get their commitment.

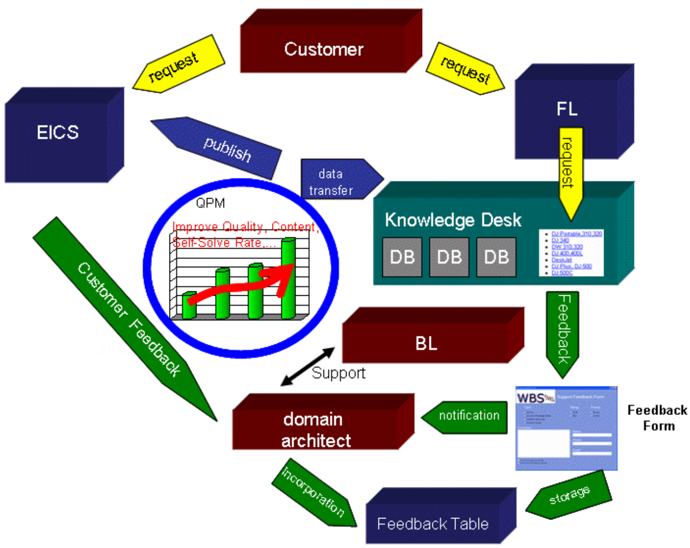

Fig. 10. Complete feedback cycle for content engineering and maintenance

The next figure 10 presents the whole feedback cycle to visualize the most important aspects of the process flow and their embedded components. The customer requests the support center via ElCS or FL. The FL agent tries to solve the user request with KD. If the FL agent detects an error, bad content or has new ideas for improvement steps he will give feedback. The agent fill out the feedback form and has the possibility to demand a telephone call for clarification or demand an email to get the information that the feedback is established. If the feedback is in the closed status, the system will send an email notification automatically. The given feedback is meanwhile stored in feedback tables for processing and later review [13]. The knowledge domain responsible verify the feedback, update content, makes status updates to aid feedback incorporation [19]. On the other side the ElCS customer feedback request is a possibility to improve the customer loyality, customer satisfaction and customer self-solve rate.

The knowledge desk solution trees and support databases are involved in the content maintenance process. A web interface is easy to use for content administration, publish preparation and content review.

5 QPM Metric for the Support Feedback Process

Incident Managements goal is to minimize the adverse impact of technology problems on business operations, ensuring that the highest levels of service quality and availability are maintained. One way that supports this process is by bringing together management data from across the infrastructure and giving IT [18] a single place to find and fix problems [21]. We will next consider the quality and performance management of the support feedback process to assist the support management in generating business support measurements for support innovation, risk and efficiency [4]. Support innovation touch e.g. growth in support market, percentage of revenue from new products and processes, profits resulting from new business operations or percentage of research and development in the support business field. Risk measures the number of customer complaints, percentage of repeat customers as of total, the employee turnover ratio or return on research and development spending [8]. Efficiency is an indicator for the profit per customer, value added per employee, usage of support tools and knowledge bases, time to market of new products and services or the revenues per customer. The quality and performance measures for the complete feedback cycle with embedded support groups and functionalities [9] like domain architects leads to a Quality & Performance Management (QPM) metric for knowledge creation and maintaining, management and support knowledge tools.

After the QPMs were described in a project team it had to be communicated to the different divisions and especially domain architects. How important is the QPM for the knowledge engineering process is the main question to answer by allocating values between 1 (low) and 3 (high). Based on the defined QPMs in table 1 a tasks list was created and clustered in topics education, knowledge usage, knowledge content increase, knowledge accuracy, quality of ongoing maintenance, knowledge tool availability and advertisement. We can determine the task-QPM influencing factor as value in the interval [0,3] ([no influence, direct high influence]). The corresponding QPM metric table illustrates figure 11 below:

Table 1. Definitions of QPMs for the support engineering process with DAR ratings

QPM's	average ranking	single ratings								standard deviation
1: % trained users of to be trained users (FL)	1,63	1	2	2	1	2	2	1	2	0,52
2: % trained authors (DAR)	2,13	3	1	3	2	2	2	2	2	0,64
3: # of scheduled author trainings	1,63	2	1	2	2	1	2	1	2	0,52
4: # of scheduled user trainings	1,63	1	2	2	1	1	3	1	2	0,74
5: # of new nodes per domain	1,00	1	1	1	1	1	1	1	1	0,00
6: # of new cookbooks per domain	1,00	1	1	1	1	1	1	1	1	0,00
7: % of nodes up-to-date and accurate (not expired, no pending feedback)	2,75	2	3	3	3	3	2	3	3	0,46
8: ElCS: # of ElCS user responses with YES to "Did this answer your question?"	2,75	3	3	3	3	3	3	2	2	0,46
9: ElCS: # of unique ElCS knowledge trees users	1,88	1	2	2	2	3	2	2	1	0,64
10: ElCS: # of total ElCS knowledge trees user sessions	1,38	1	1	2	2	1	1	1	2	0,52
11: ElCS knowledge trees traffic	1,25	1	1	1	2	1	1	1	2	0,46
12: ElCS: # of Customer feedback in ElCS related to user sessions	2,75	3	3	3	3	3	3	3	1	0,71
13: ElCS: avg. customer's rating of knowledge tree quality (5=good, 1=bad)	2,88	3	3	3	3	3	3	3	2	0,35
14: # of new FL feedback items given	2,38	3	3	3	2	2	2	2	2	0,52
15: pre-qualified calls in BL related to total calls in BL	1,88	2	1	2	2	2	1	2	3	0,64
16: ElCS: calls opened within knowledge trees related to calls opened directly	1,75	1	2	2	2	1	2	1	3	0,71
17: calls closed in FL with KD related to total closed calls in FL	2,38	1	2	2	3	3	3	2	3	0,74
18: # of feedback sessions with BL/DAR regarding usability and satisfaction	1,25	1	1	1	2	1	1	1	2	0,46
19: decrease of invested time to solve	1,75	1	1	1	2	2	2	2	3	0,71
20: total count of feedback items status AS/OP/PE	1,00	1	1	1	1	1	1	1	1	0,00
21: feedback items AS/(AS + OP + PE)	1,25	1	2	2	1	1	1	1	1	0,46
22: % of DAR resources related to total support head count	2,25	3	2	3	2	2	3	2	1	0,71
23: time to implement bugfixes and tool-enhancements	2,38	3	2	3	2	2	2	2	3	0,52
24: ElCS: Customer satisfaction with selfhelp in ElCS	1,63	1	1	1	2	1	2	2	3	0,74
25: ElCS: customer loyality with selfhelp in ElCS	1,50	2	1	1	1	1	1	2	3	0,76
26: Cost per call	1,50	3	1	1	1	1	2	1	2	0,76
27: support satisfaction with prequalified calls	1,88	2	1	3	2	1	1	3	2	0,83

The average in column *AC* is a significance value concerning each task over all summarized QPMs, the average in row line 32 signify the quality and performance influence concerning the sum of the listed tasks.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1	**Tasks / QPMs**	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	average
2	**Education**																												
3	training, user	3	1	1	3	0	0	2	0	0	0	0	0	0	2	1	0	1	0	1	1	0	1	0	3	3	2	1	0,96
4	training, author	0	3	3	3	0	0	2	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	1	1	1	0	0,67
5	**Knowledge usage**																												
6	knowledge usage externally	0	0	0	0	1	1	2	2	3	3	3	3	2	0	0	3	0	0	0	0	0	3	0	2	2	1	2	1,22
7	knowedge usage internally	3	3	3	3	3	3	3	2	0	0	0	2	0	2	0	2	3	1	0	2	2	3	0	1	2	1	2	1,70
8	**Knowledge content increase**																												
9	provide human resources for knowledge creation	0	0	0	0	3	3	3	2	2	2	2	2	2	2	2	2	2	3	2	3	3	0	0	3	2	2	1	1,78
10	content generation progress	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	0	2	1	1	1	2,52
11	**Knowledge accuracy**																												
12	external customer (EICS) satisfaction	0	3	3	3	3	3	3	0	3	3	3	0	0	0	3	3	0	0	0	0	0	3	0	3	2	1	2	1,63
13	internal customer (FL) satisfaction	3	3	3	3	3	3	3	0	0	0	0	0	0	2	3	2	3	3	3	2	2	3	0	2	2	2	3	1,96
14	**Quality of ongoing maintenance**																												
15	regular feedback sessions with FL/BL/DAR regarding tools	2	3	3	3	0	0	0	0	0	0	0	0	0	2	2	1	2	3	2	3	3	3	3	2	1	3	2	1,59
16	regular feedback sessions with FL/BL/DAR regarding processes	3	3	3	3	1	1	1	0	0	0	0	0	0	2	2	1	2	3	2	3	3	3	3	3	2	2	1	1,74
17	review and maintenance of knowledge tree structure	0	3	3	3	0	0	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	0	3	2	1	2	2.41
18	review of feedback to nodes	0	3	3	3	0	0	3	3	3	3	3	3	3	3	3	3	3	2	3	3	3	3	0	2	1	2	0	2,26
19	review of knowledge documents	0	3	3	3	0	0	3	3	3	3	3	3	3	3	3	3	3	1	3	3	3	3	0	2	1	1	0	2,19
20	close feedback loop (FL/BL) about knowledge	3	3	3	3	2	2	3	2	2	2	2	3	3	3	3	1	3	2	2	3	3	3	0	1	2	1	0	2,22
21	**Knowledge Tool Availability**																												
22	provide IT Supported tools and environment	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	3	2	3	2	1	1	0	0	3	2	2	0	1.11
23	tool enhancements	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	3	1	1	1	0	3	2	1	3	2	1.00
24	high availability of tools	0	0	0	0	1	1	1	3	3	3	3	2	2	2	3	3	3	3	2	0	0	0	2	2	1	2	2	1,63
25	provide a portfolio of tools for knowledge creation and maintaining	0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	3	3	3	3	3	0	3	2	1	2	2	1,81
26	regular feedback sessions with FL/BL/DAR regarding tools	2	3	3	3	0	0	0	0	0	0	0	0	0	2	2	1	2	3	2	3	3	3	3	1	1	1	2	1.48
27	provide functional and stable runtime environment	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	3	2	3	1	0	0	0	3	1	1	1	1	1,00
28	**Other and not used for sponser review**																												
29	advertisement for local tools	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	0	1	1	1	2	2	0	2	1	1	1	1	0,67
30	advertisement for services and processes EICS	0	0	0	0	0	0	0	2	2	2	2	2	2	0	3	3	0	0	0	0	0	0	2	2	2	1	2	1,00
31	advertisement for services and processes local	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	0	2	2	3	2	2	0	0	1	0	1	1	0,70
32	average	0,83	1,61	1,61	1,70	1,04	1,04	1,70	1,30	1,39	1,39	1,39	1,35	1,22	1,74	2,04	1,87	1,96	1,95	1,74	1,83	1,78	1,61	1,04	1,96	1,48	1,52	1,30	

Fig. 11. QPM metric with associated tasks

6 Conclusions

This paper has presented a new approach on support-organizational incident management by proposing a novel feedback concept and feedback process as workflow integration. This feedback process solution is a whole organizational concept, in which each participating support organization group may define and realize its relative support processes from their own perspective. Each support participant can access and give feedback. Feedback verification and content updates facilitates content maintenance, feedback incorporation and finally knowledge desk usage and process integration. In the future, we plan to integrate the quality and performance measurements in the online feedback cycle to adjust and refine support business measurements automatically.

References

1. Allweyer, T.: Modellbasiertes Wissensmanagement. IM Information Management 13(1) (1998) 37-45
2. Anton, J.: The Past, Present and Future of Customer Access Centers. Int. Journal of Service Industry Management 11 (2000) 120–130
3. Anupindi, R., Smythe, B.T.: Call Centers and Rapid Technology Change. Teach Node (1997)

4. Brewer, P., Speh, T.: Using the Balanced Scorecard to Measure Supply Chain Performance. Journal of Business Logistics 21(1) (2000) 75-93
5. Giaglis, G.M.: On the Integrated Design and Evaluation of Business Processes and Information Systems. Comm. of the Association for Information Systems, Vol. 1 (1999)
6. Grossman, T.A.: Call Centers. Encyclopedia of Operations Research and Management Science, 2nd Edition (1999)
7. Grütter, R.: IT Servicemanagement nach ITIL – Unterschiede zwischen Anbietern und Kunden. Diploma thesis, University of Zurich, Switzerland (2005)
8. Holland, A.: Modeling Uncertainty in Decision Support Systems for Customer Call Center. Int. Conf. 8th Fuzzy Days, Dortmund, Germany, Computational Intelligence in Theory and Practice, Springer Series Advances in Soft Computing (2004) 580-588
9. Holland, A., Peitzsch, K.M.: Learning Skills from Data Based on XML Structured Qualification Profiles. Proc. of I-KNOW '05, 5th Int. Conf. on Knowledge Management, Graz, Austria (2005) 170-178
10. Holland, A., Fathi, M.: Experience Fusion as Integration of Distributed Structured Knowledge. 6th Int. Symposium on Soft Computing for Industry. World Automation Congress, Budapest, Hungary, July 24-27 (2006)
11. Jablonski, S., Horn, S., Schlundt, M.: Process Oriented Knowledge Management. 11th Int. Workshop on Research Issues in Data Engineering, Heidelberg, Germany, IEEE Comp. Society (2001) 77-84
12. Kim, S., Hwang, H., Suh, E.: A Process-based Approach to Knowledge Flow Analysis. Knowledge and Process Management 10(4) (2003)
13. Koole, G., Mandelbaum, A.: Queueing Models of Call Centers. Annals of Operations Research 113 (2002) 41-59
14. Maier, R.: Knowledge Management Systems. Springer-Verlag Berlin (2002)
15. Maier, R., Remus, U.: Implementing Process-Oriented Knowledge Management Strategies. Journal of Knowledge Management 7(4) (2003) 62-74
16. Pinedo, M.: Call Centers in Financial Services. Strategies, Technologies and Operations. Kluwer Publisher (1999)
17. Raimann, J., Enkel, E., Seufert, A., von Krogh, G., Back, A.: Supporting Business Processes through Knowledge Management. A Technology-based Analysis. Technical report, University of St.Gallen (2000)
18. Thiadens, T.: Manage IT! Organizing IT Demand and IT Supply. Springer Publish (2005)
19. Weerakkody, V., Currie, W.: Integrating Business Process Reengineering with Information Systems Development. Proc. of BPM Conf., Eindhoven, Netherlands (2003)
20. Williams, M.: Microsoft Visual C#.NET. Microsoft MS Press (2002)
21. Zhao, X., Liu, C.: An Organisational Perspective on Collaborative Business Processes. 3rd International Conference on Business Process Management, Lecture Notes in Computer Science, Vol. 3649. Springer-Verlag, Berlin Heidelberg New York (2005) 17-31

DKOMP: A Peer-to-Peer Platform for Distributed Knowledge Management

Vikrant S. Kaulgud and Rahul Dolas

Software Engineering and Technology Labs (SETLabs), Infosys Technologies Ltd.,
Plot no. 1, Rajiv Gandhi Infotech Park, Hinjawadi, Pune – 411027
{vikrant_kaulgud, rahul_dolas}@infosys.com

Abstract. Distributed knowledge management (KM) is a key component of any enterprise's knowledge management strategy. However, issues related to implementing a robust distributed KM system deter their adoption in enterprises. We studied the requirements of a distributed KM system from an enterprise's perspective. Based on our study, we propose DKOMP; a P2P infrastructure for distributed KM. DKOMP integrates tightly in to enterprise network infrastructure and yet retains the benefits of P2P paradigm. DKOMP should provide a good vehicle for increasing adoption of distributed KM in enterprises.

Keywords: Distributed Knowledge Management, Peer-to-Peer, Architecture.

1 Introduction

An enterprise Knowledge Management System (KMS) implementation, often has a centralized architecture. Usually, there is a central repository of knowledge artifacts representing the explicit knowledge within the enterprise. Taxonomy provides a structure to the repository. Typically, users upload knowledge artifacts, search, and download artifacts via a web-portal. Often, a user-created expert profile captures and represents the tacit knowledge (knowledge held by humans). These profiles contain a list of areas in which the users has expertise. Users can search for relevant profiles through the portal. Centralized KMS has many benefits such as better manageability, high degree of control over the knowledge resources, easier deployment and higher accessibility.

While centralized KMS are effective for managing explicit knowledge (knowledge captured in artifacts), they are insufficient for tacit knowledge. The sheer reluctance of users to share knowledge via a central system is a dampener in capturing and disseminating tacit knowledge [3]. Often, users are unwilling to share knowledge via centralized KMS due to confidentiality issues. Interactions within communities of interest and / or practice create tacit knowledge. Mostly, centralized KMS fail to capture these interactions. Knowledge is contextual [22]. A centralized KMS acts as a repository and offers little support for a context-based usage of the resources. E.g. a user may wish to understand how a case study can be reused in a specific scenario. To achieve this, the user should be able to create an adhoc community of experts, which can facilitate the context-based usage of the case study. The output of such interactions itself becomes part of the knowledge pool. Furthermore, communities of interest

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 119 – 130, 2006.

are informal and cut across organizational structure. Typically, centralized KMS do not facilitate evolution of such communities.

An analysis of these representative cases leads to the necessity of enabling a social approach to knowledge management. The social approach harnesses knowledge creation and consumption happening at the organizational edges adding significant value to enterprise knowledge. Social approaches e.g. social networking imposes concepts of reputation and relevance, leading to more effective knowledge retrieval [20]. Tiwana [23] and Wellman [26] have emphasized the need of for a community approach to knowledge management and the value social approaches bring to knowledge community. Since, the social approach is essentially distributed, it mandates use of an effective distributed KMS. It is pertinent to understand that both centralized and distributed KMS, have their own place within the enterprise. They are complementary rather than competitive.

In this paper, we present DKOMP (Distributed KnOwledge Managenent Platform); a comprehensive platform for Peer-to-Peer (P2P) distributed KMS. It is a modular infrastructure architecture and can be adapted to suit specific KMS requirements. As it provides an infrastructure, various algorithms (e.g. for expert profiling) can be easily plugged to match specific requirements. DKOMP is Java-based and built on mature open-source technologies making it open and portable across multiple platforms. The DKOMP deployment architecture also allows extending the distributed KMS to mobile handhelds such as PDAs. It is easy to integrate DKOMP with enterprise systems such as authentication systems, allowing a P2P application to leverage existing enterprise resources.

The paper's structure is as follows: Section 2 discusses the architecture details and deployment. In section 3, we present a real-life application we are building on DKOMP. In Section 4, we highlight existing works. Finally, we conclude the paper in Section 5.

2 DKOMP Architecture

2.1 DKOMP Architecture

The DKOMP architecture is presented in Figure 1. The basic P2P facilities mentioned in Figure 1 are provided by the generic P2P platform. The generic P2P platform provides the communication substrate for all DKOMP functionalities. The Content Management Platform provides the required modules for implementing distributed content repositories. Each peer running DKOMP has a local content repository containing knowledge artifacts shared by the user. The Content Management Platform manages this repository. The basic functions that it offers are content sharing, downloading and notifications. Apart from this, the Content Management Platform has modules for attaching additional and richer metadata to shared files and utilizing this for search operations. It offers a keyword-based search. The Content Management Platform also provides adapters for integrating non-textual content such as databases in to the DKOMP environment. Using adapters, DKOMP allows the users to capture, represent and consume knowledge in whatever form it occurs. Finally, the Content Management Platform has modules to support network security features such as access control and optionally Digital Rights Management (DRM).

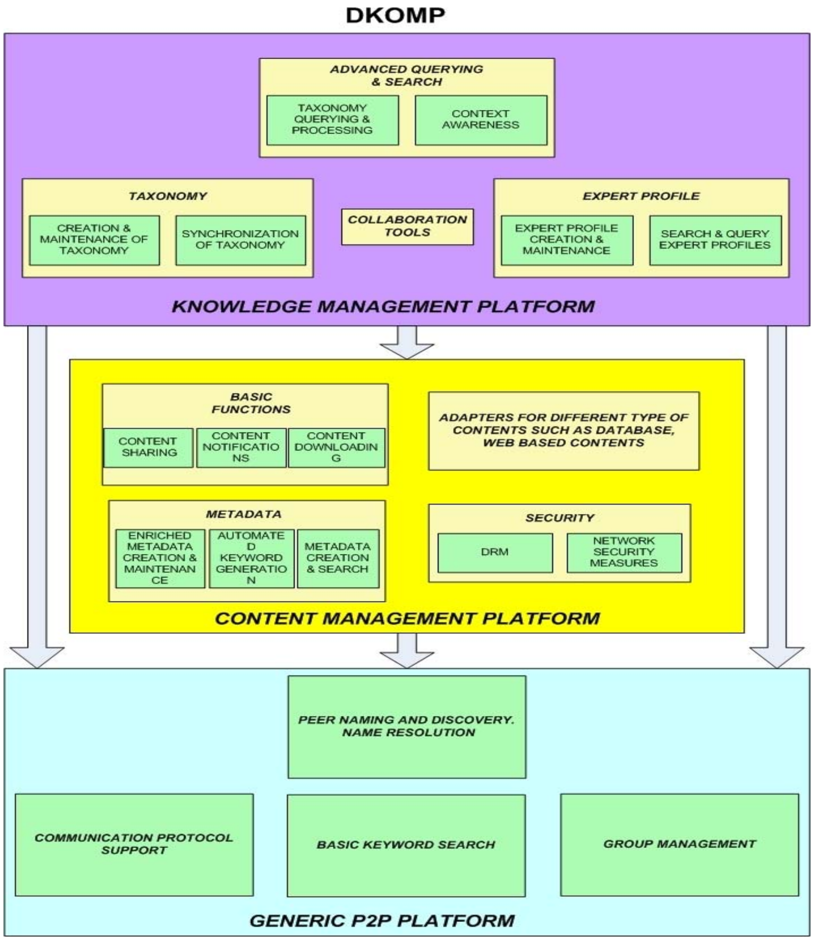

Fig. 1. The DKOMP architecture

The core knowledge-management functionality is embodied in the Knowledge Management Platform. The major functions of the KM platform are:

- Taxonomy: Taxonomy imposes a logical categorization on the shared knowledge. This classification of knowledge provides a strong criterion for searching the required knowledge in an efficient way in a diverse knowledgebase. This module manages the activities pertaining to distributed taxonomy management. The maintenance of the local copy of the taxonomy and synchronizing it as and when required is a crucial task in a distributed knowledge management scenario.
- Expert profiling: Identifying experts is a key component of any knowledge management initiative. Creation, sharing and searching of expert profiles in and across the enterprises is an important task in the collaborative work scenario. Using these profiles collaborative sessions can be created using various collaborative tools.

Table 1. Mapping of design principles to JXTA benefits

No.	Design Principle	Requirements from the P2P platform	JXTA benefits
1	Enterprise-grade qualities	Scalability	Uses a hybrid P2P network structure based a loosely consistent DHT
		Robustness	Intensive community development and testing of the platform
		Security	Supports various levels of security including digital certificates and distributed trust based on Poblano [5]
		Manageability	Integrated Metering and Monitoring capabilities
		Cost-effective	Based on a flexible Apache 2.0 open source license
2	Ease of deployment	Platform independence	JXTA is a platform specification and has bindings to Java, C, .Net, J2ME etc.
		Multiple application development models	Applications can be desktop, web-based or mobile-based
3	Ubiquitous usage	Platform independence	JXTA is a platform specification and has bindings to Java, C, .Net, J2ME etc.
		Firewall traversal	JXTA allows firewall traversal using specialized relay peers.
		Access to P2P resources from mobile handhelds	J2ME-based application development possible
4	Modularity and extensibility	Modular architecture	JXTA has a highly modular and layered architecture
		Amenable to augmentation of existing services and / or embedding new services	Open specification and open source allows easy augmentation of services and / or new services
5	Support for community building	Flexible group management	Simple and extensible secure group management API

- Advanced search and querying: The advanced search and querying module consists of:
  - Context awareness: Context is a significant value-addition by users to mere content/information. Context specifies how the knowledge artifacts were created, who used it, how was it useful, views of other users on the artifact etc. This module allows users to attach context information to the artifacts. It again emphasizes the social aspects of DKOMP. Context information can also be used for advanced knowledge search.
  - Taxonomy processing: While distributed taxonomy management provides a robust infrastructure, some ambiguity may still crop in. In addition, users may

wish to find “which categories are similar to category X?” The advanced taxonomy processing modules enable such queries.

2.2 Platform Selection and Enhancements

One question we answered during the initial phase of the development was on the choice of the P2P platform. Considering the design guidelines, we selected JXTA [12] as the basic platform. Table 1 presents the detailed rationale behind selection of the JXTA platform. Other platforms such as Groove [9] and Microsoft Windows P2P SDK [27] were not used as they are commercial and their source code is closed. We wanted to use an open-source platform to reduce licensing fees and have access to the source code for tight integration of the P2P platform with DKOMP. Another factor against both these platforms was that they are available for Windows OS only and this represents a limitation as far as ubiquitous deployment is concerned. Another decision we had to take was selection of the content management system (CMS). DKOMP is based on a distributed content repository, with peers maintaining local content repositories. It was essential to achieve a tight integration of the local content repository with the P2P and DKOMP platform running on the peers. Moreover, we wanted a lightweight CMS, so that it removed any requirement for CMS configuration on peers. This significantly increased the ease of deployment. Considering these factors, we selected the JXTA CMS content management system [12]. We had to enhance the JXTA and JXTA-CMS [12] platforms for supporting the facilities required for DKOMP. The key enhancements include:

- Rich metadata: JXTA-CMS supports basic meta-data regarding a file such as its name, type, size etc. However, these metadata attributes are insufficient for use in the knowledge management application. We enhanced the JXTA-CMS to accept additional metadata attributes from the Dublin Core (DC) [7] metadata. The enriched metadata includes the fields such as Title, Author, Publisher, Authors address, Taxonomy etc.
- Multi-Attribute query processing: JXTA-CMS uses the simple, single-attribute JXTA content search. Typically, the content search is based on combination of filename and wildcards. However, this search facility is restrictive and insufficient. E.g., we would like to search for knowledge based on a combination of the metadata attributes and logical operators. JXTA-CMS and JXTA platforms were augmented to support multi-attribute query formation and processing.
- Support for notifications: JXTA-CMS supports a traditional query-response operational mode. We augmented it to support notifications as well. Notifications are used for variety of purposes such as notifying availability of new content etc.
- Full text search: This is one of the most important enhancements to JXTA-CMS. Currently JXTA-CMS does not allow a full-text search. We embedded the Lucene [14] indexing service within CMS for permitting full text search. With this, a Lucene index is maintained for every shared file. With full-text search embedded, the content search works in three modes: only on metadata, only on the full-text indices or on combination of the metadata and the full-text indices.

Apart from these enhancements, we added several new features to the platform. These features are specific to knowledge management activities and build on JXTA and JXTA-CMS. The features are:

- Interest Group: Interest group is the group of peers having common interests in knowledge. The platform provides facility to generate such interest groups and publish them over the network. Other peers can subscribe to these groups to be able to take advantages of the activities done in the Interest Group. A peer can be member of multiple Interest Groups at a time. Interest groups impose some level of security over the knowledge shared in the network.
- Distributed taxonomy management: Given the emphasis on social aspects of distributed KMS, we have incorporated a distributed taxonomy management module in DKOMP. It allows the interest group to evolve a taxonomy specific to the group's requirements. Moreover, the collective intelligence behind the distributed taxonomy makes it richer and more meaningful. Any interest group starts with a blank taxonomy. As the users start sharing knowledge, they begin creating the taxonomy. Thus, taxonomy grows along with the knowledge exchange activities within an interest group. The key problem we tackled in distributed taxonomy management was of maintaining a coherent taxonomy. As there is no central control on taxonomy definition, it is possible that users create categories that clash with each other or have ambiguous hierarchy. Such chaotic scenarios defeat the purpose of a distributed taxonomy building. We have addressed the issues by:
  - Taxonomy synchronization: Before any operation on the distributed taxonomy is permitted, the platform synchronizes the locally available taxonomy with the global taxonomy. Secondly, whenever, a new category is added, a notification is sent to all peers informing availability of the new category.
  - Timestamps: To minimize the network overhead and computational overhead associated with taxonomy synchronization, we have used timestamps. Using timestamps only the latest taxonomy information is retrieved from the P2P network. Moreover, since multiple peers may respond to a taxonomy update request, reducing the responses using timestamps reduces the computational overhead associated with creation of the global view on the requesting peer.
- Expert profile: Expert profile is another major feature of DKOMP. An API enables users to create their own expert profile and publish the same over the network. DKOMP also provides facility of multi-attribute queries for searching experts from the available profiles. The expert profiles retrieved by a search can be used to establish collaborative sessions with experts as and when required. It is possible to embed various expert profiling algorithms and techniques in to this module for automating the profile generation. In a simpler application, profiles can also be generated manually.

2.3 Deployment Architecture

Figure 2 shows the deployment plan of DKOMP. The deployment uses a hybrid P2P approach [23], where centralized systems (super-peer) are used for key service-specific activities. The rationale behind using a hybrid P2P approach is to achieve scalability and allow firewall traversal. Moreover, certain functionalities such distributed taxonomy management and manageability can be deployed in an easier, robust and scalable manner using the hybrid approach. Using a centralized server does not affect the "P2P-nature" of the system. Applying Shirky's litmus tests [21], we observe:

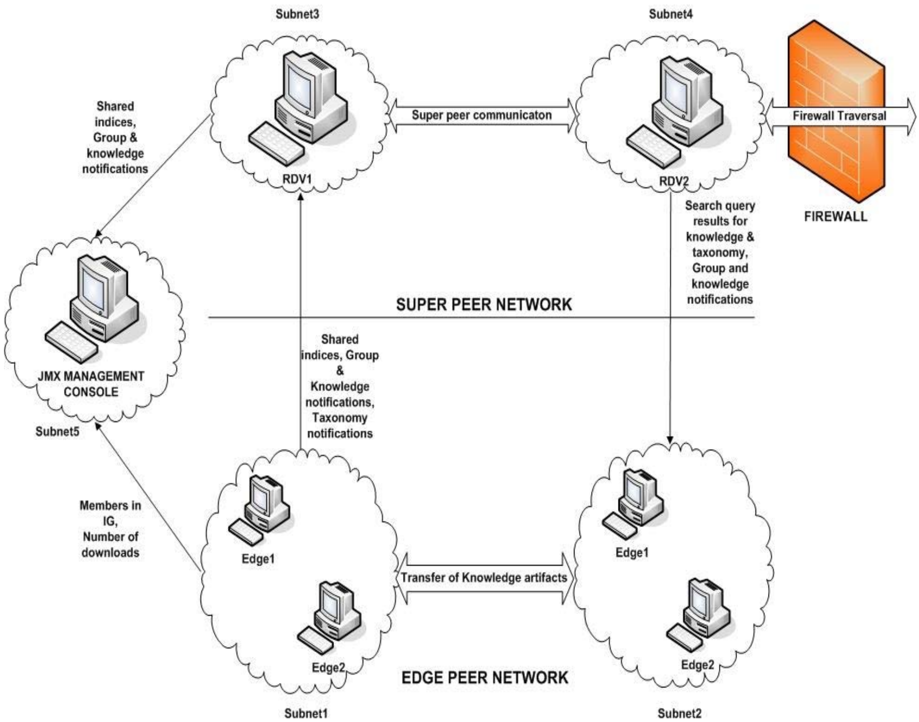

Fig. 2. An schematic depicting a deployment of an enterprise-wide distributed knowledge management environment based on DKOMP. The schematic also highlights the manageability features embedded in DKOMP.

- Litmus test 1: DKOMP does not rely on fixed network addresses for the peers and is able to handle variable connectivity.
- Litmus test 2: DKOMP gives maximum autonomy to peers, as the users have full control on creation of interest groups, sharing and unsharing of files, subscription to interest groups and most importantly taxonomy building.

2.3.1 Service-Specific Super-Peers

In case of DKOMP, the centralized systems are service-specific super-peers based on the JXTA Rendezvous service [24]. The DKOMP platform is adapted to provide the required super-peer functionality. The service-specific super-peers provide the following functionality:

- Centralized indices: Knowledge artifacts are shared locally and a local index is maintained for each shared artifact. However, since most firewalls prevent multicast routing and broadcast, search across firewalls becomes an issue. Most enterprises are spread across multiple sites/locations and site-networks are delineated by firewalls. Hence, the indices are also maintained centrally at the super-peers, such that search is possible across multiple sites and across firewalls.

- Notification management: Similarly, notifications, which are usually sent using multicast / broadcast, will be ineffective in a firewall-delineated enterprise network. Super-peers run a service that accepts notifications from one peer for an interest group and propagate them out to other connected peers of that interest group.
- Taxonomy synchronization: This is a key service running on the super-peers. Each update to the taxonomy of an interest group is sent to the taxonomy synchronization service on the super-peers. This service updates the taxonomy tree for an interest group on receiving the update. Prior to a taxonomy-specific operation, each peer requests the super-peer for the current taxonomy tree. This way, the taxonomy remains consistent and coherent. Putting the taxonomy synchronization on the super-peers ensures consistency even if the peers in an interest group are spread across firewall-delineated sites.

2.3.2 Manageability

One of the requirements of an enterprise-grade platform / application is manageability. We have implemented manageability using JXTA's Metering and Monitoring Project [11]. It allows metering of application-specific parameters on local peers or super-peers. However, most enterprises adopt standard application management protocols and tools such as JMX. [10] We have integrated the JXTA Metering and Monitoring Project with JMX protocol. Using this integration, we allow monitoring and management of the DKOMP-based distributed knowledge network. We have engineered DKOMP and made enhancements to the JXTA and JXTA-CMS platform to provide key information regarding health of the distributed knowledge network. In particular, we currently monitor the following parameters:

- Number of interest groups
- Number of artifacts shared per group and in the entire knowledge network
- Number of notifications per interest group
- Membership per interest group
- Number of downloads of an knowledge artifact

Of these parameters, the first three parameters are obtained from the services running on the super-peers. The last two parameters are obtained from the peers. To ensure that up to date information is available, changes to the peer-specific parameters (4 and 5) are pushed out proactively to the JMX console.

2.3.3 Security

We have engineered DKOMP to leverage existing enterprise security infrastructure for two operations: first for user authentication on application startup and second for interest group membership. In a proof-of-concept application, we integrated DKOMP with LDAP authentication service. Similarly, integration is possible with other authentication services such as Kerberos, RADIUS etc. When a user starts a DKOMP-based application, it is suggested that the user be presented an authentication window. The user is expected to use the corporate user identification (username) and password. These are used to authenticate the user against the enterprise authentication service. On successful authentication, the username is also used as the peer-name. When a user creates an interest group, she can restrict the users who can subscribe to the interest group. In DKOMP, these restrictions are based on the users and groups defined in the enterprise authentication service.

3 Related Work

Distributed artifact sharing and searching depends on the P2P network structure [15]: centralized, decentralized structured, decentralized and unstructured. In all network structures, the artifacts (e.g. documents, spreadsheets, presentations etc.) remain on the peers (the machines used by users for participating in the KMS). The location of indices determines the P2P network structure. In the centralized structure (e.g. Napster [17]); the indices are stored on a central server. In the decentralized, structured networks, uses a distributed data structure such as a Distributed Hash Table (DHT) [24] for maintaining resource indices. This structure is highly scalable and robust. Finally, we have the completely decentralized structure, wherein there is no central server used, e.g. Gnutella. Edutella [18] is a framework based on use of RDF [19] for query processing in decentralized, structured P2P systems. While the Edutella framework targets educational applications, it is general enough to be adapted for use in other usage and business domains. It builds on the JXTA [12] P2P framework, where JXTA offers the P2P communication mechanisms and Edutella offers resource description and query processing infrastructure. In Edutella, peers register the kind of queries they are interested in answering while querying peers register the queries. Queries are propagated through the Edutella network to the interested peers. Edutella also supports annotations for different domains. This can be used for discovering knowledge artifacts. Kex, a P2P distributed KMS based on JXTA [12] and reported in [4] uses a context-awareness module. Kex has a context-awareness module consisting of a context-browser and context-editor. Users are able to create contexts and use context information for query formation. Context-based operations such as querying based on context, building a knowledge map based on context are possible. Answer Garden [1], [2] is a key distributed expert identification work. In Answer Garden, users can search for experts on specific topics. The search is run over central servers that maintain expert profile. Once the experts are identified, a querying user is connected to an expert for collaboration. However, group collaboration is not available in Answer Garden. Lack of group collaboration limits the interactivity of Answer Garden. Collaboration, especially synchronous collaboration, has significant use in knowledge management. Synchronous collaboration facilitates community building and participation. Groove [9] is a key P2P collaboration platform. Several initiatives on JXTA such as myJXTA [16] also aim at providing a collaboration platform. However, these have to achieve maturity that Groove demonstrates. Apart from research-related works and commercial offerings, a comprehensive discussion of P2P knowledge management is available in Tsui [25]. With the exception of Kex, most of the works are focused on a specific component of a distributed KMS. Kex and our platform, DKOMP are both based on JXTA. There are several advantages of DKOMP:

- It supports distributed taxonomy evolution, giving maximum control to the community of practice / interest. We have built a Collaborative Taxonomy Building system [13], which makes the distributed taxonomy building more structured.

CDTB uses a P2Penabled adhoc workflow system to structure the taxonomy building. We have used roles of "proposer", "business expert" and "publisher" in the workflow. Users are normally proposers and they generate requests for addition / modification of the interest group's taxonomy. Business experts validate the request and make recommendations to the publisher. Publisher is ultimately responsible for actual modification of the taxonomy.

- DKOMP's interest-group based activity control facilitates focused activities within the knowledge network. Interest groups promote social community building within the enterprise and these can be leveraged for many applications.
- DKOMP also provides infrastructure for expert profile search and collaboration.
- DKOMP supports manageability features. This makes it suitable for deployment in an enterprise. Enterprises can exert control and monitoring over DKOMP-based applications without disrupting the power of P2Penabled distributed knowledge management.
- DKOMP is packaged as an infrastructure platform with a well-documented API. DKOMP's modularity makes it suitable for deployment in various scenarios.

4 Real-Life Application of DKOMP

We have implemented a TipBox Service based on the DKOMP platform. "Technical tips" form a very important source for knowledge in our company. The tips range from how-tos to best practices in diverse areas such as Programming, databases, domains etc. Currently, the Tips are sent through e-mail on a periodic basis. An analysis of the current implementation revealed the following lacunae:

- With tips landing in the email Inbox, it becomes difficult for users to manage the tips and search for tips. The email management and search capabilities of the email client is obviously the bottleneck.
- Method of user's subscription to various areas is not intuitive.
- Users do not get a feeling of "community" in the current implementation. They do not have an intuitive way of reaching out to experts in case they have more queries on the Tip.
- Only a limited set of users; identified as experts are permitted to submit tips. Vast majority of users who generate tremendous amount of tacit knowledge are left out, reducing the richness of the knowledge-base.

We found DKOMP to be suitable to address these lacunae. We have designed a "TipBox Service", that harnesses the collective intelligence of knowledge-workers by allowing them to send technical tips to communities of practice. TipBox Service is highly interactive and taps in to the massive knowledge generated at the organizational edges. DKOMP's interest group service is used for user subscriptions and pushing out Tips to interested users. More importantly, users are now empowered to submit tips. The collaborative taxonomy building feature of DKOMP is reused for validating and authorizing Tips. Hence the richness of the tip knowledge-base is higher. DKOMP's full-text search is used for searching Tips. Using the expert profile infrastructure of DKOMP, there is an "Ask the Expert" tool which users can use for contacting experts for query resolution or getting more information on Tips. The

TipBox Service is implemented as a desktop application and resides in the System-Tray of a Windows desktop. Users access the various functions of the Service through the tray-icon. Notifications of new tips and tip authorization requests pop-up through the tray-icon and users can take necessary actions on the pop-up. We have used the Service-Specific Super-peers for centralizing the Tip Content (for archival value) and maintaining the role database and business rules for workflow routing. Currently, we are in the process of deploying the TipBox Service with an initial user base of 1500 users. Gradually we intend to deploy this service across all development centers.

5 Conclusions

We have presented the design rationale, platform architecture and deployment architecture for DKOMP, a P2P infrastructure for distributed knowledge management. DKOMP is robust, secure and scalable. It addresses the key concerns of an enterprise regarding deployment of a distributed KMS. Moreover, since DKOMP is packaged as a platform API, various applications can be built over it. We have presented a case of the TipBox Service that is being deployed in an initial user-base of 1500 users. Going ahead, the key activity would be enhancing the DKOMP with automated expert profiling and better-automated metadata generation. We are also planning to extend the DKOMP-based distributed KMS to mobile users.

References

1. Ackerman, M.S.: Augmenting the organizational memory A field study of answer garden. ACM Conference on Computer Supported Cooperative Work (CSCW '94). (1994) 243-252.
2. Ackerman, M.S., McDonald, D.W.: Answer garden 2: Merging organizational memory with collaborative help. ACM Conference on Computer Supported Cooperative Work (CSCW '96). (1996) 97-105.
3. Adar, E., Huberman, B.A.: Free riding on Gnutella. First Monday. Available from http://www.firstmonday.dk/issues/issue5_10/adar/index.html; accessed May 30, 2006.
4. Bonifacio, M., Bouquet, P., Mameli, G., Nori, M. KEx: A Peer-to-Peer Solution for Distributed Knowledge Management. Lecture notes in computer science (2002) 490-500.
5. Chen, R., Yeager, W. Poblano: A Distributed Trust Model for Peer-to-Peer Networks. Available from http://www.jxta.org/docs/trust.pdf; accessed May 30, 2006.
6. Distributed Hash Table. Available from http://en.wikipedia.org/wiki/Distributed_hash_table; accessed May 31, 2006.
7. Dublin Core Metadata Initiative, http://dublincore.org.
8. Gnutella, http://www.gnutella.com; accessed May 31, 2006.
9. Groove Project, http://www.groove.net; accessed May 31, 2006.
10. Java Management Extensions (JMX). http://java.sun.com/products/JavaManagement/; accessed May 31, 2006.
11. JXTA Metering and Monitoring Project. http://meter.jxta.org/; accessed May 31, 2006.
12. JXTA Project. Sun Microsystems Inc. http://www.jxta.org; accessed May 31, 2006.
13. Kaulgud, V.: Collaborative Distributed Taxonomy Building. Infosys Technologies Ltd. (2006).
14. Lucene. http://lucene.apache.org/; accessed May 30, 2006.

15. Lv, Q., Cao, P., Cohen, E., Li, K., Shenker, S.: Search and Replication in unstructured peer-to-peer networks. 16th International Conference on Supercomputing. Linkoping, Sweden (2002).
16. myJXTA Project. http://myjxta.jxta.org; accessed May 31, 2006.
17. Napster. http://www.napster.com; accessed May 31, 2006.
18. Nejdl, W., Wolf , B. Qu , C., Decker , S., Sintek , M., Naeve, A., Nilsson , M., Palmér , M., Risch, T. EDUTELLA: a P2P networking infrastructure based on RDF. 11th international conference on World Wide Web., Hawaii, USA (2002).
19. Resource Description Framework (RDF), http://www.w3.org/RDF/; accessed May 31, 2006.
20. Schenk, S.: Introducing Social Aspects to Search in Peer-to-Peer Networks, PAIKM2005 (2005).
21. Shirky, C. (2000, November 24). What is P2P… and what isn't. Available from http://www.openp2p.com/pub/a/p2p/2000/11/24/shirky1whatisp2p.html; accessed May 30, 2006
22. Snowden, D.: Complex acts of knowing: Paradox and descriptive self-awareness. Journal of Knowledge Management. 6, 2(2002) 100-111.
23. Tiwana, A.: Affinity to infinity in peer-to-peer knowledge platforms. Communications of the ACM, 46, 5(2003) 77-80.
24. Traversat, B., Abdelaziz, M., Pouyoul, E.: Project JXTA: A Loosely-Consistent DHT Rendezvous Walker. Available from http://www.jxta.org/docs/jxta-dht.pdf; accessed May 31, 2006.
25. Tsui, E.: Technologies for Personal and Peer-to-Peer (P2P) Knowledge Management, CSC Leading Edge Forum (LEF), Computer Sciences Corporation, Available from http://www.csc.com/ aboutus/lef/mds67_off/uploads/P2P_KM.pdf; accessed May 30, 2006.
26. Wellman, B.: An electronic group is virtually a social network. In Culture of the Internet, S. Kiesler, Ed. Mahwah, New Jersey. (1997) 179-205.
27. Windows Peer-to-Peer Networking Overview. Available from http://www.microsoft.com/ windowsxp/p2p/overview.mspx; accessed May 31, 2006.

From Design Errors to Design Opportunities Using a Machine Learning Approach

Sanghee Kim

Engineering Design Centre, Department of Engineering, University of Cambridge
Trumpington Street, U.K.
shk32@eng.cam.ac.uk

Abstract. Human Errors, e.g. *a pilot mismanaged the fuel system causing engine failure and fuel starvation,* are known to contribute to over 66% of aviation accidents. However, in some cases, the real sources of the errors are the design of aircraft, e.g. *the pilot was confused with the different fuel systems across different models in the same manufacture.* The failed collaboration between human operators and the systems therefore has been a major concern for aviation industries. Aviation accident reports are critical information sources to understand how to prevent or reduce such problematic collaboration. In particular, the portions of the reports describing how the behaviour of human operators deviated from an established norm and how the design of aircraft systems contributed to this deviation are particularly important. However, it is a time-consuming and error-prone task to manually extract such information from the reports. One reason is that current accident reports do not aim specifically at capturing the information in format easily accessible for aircraft designers. Therefore, an automatic approach that identifies the sentences describing Human Errors and Design Errors is needed. A preliminary test using hand-crafted cue phrases, i.e. a special word or phrases that are used to indicate the types of sentences, showed a limited identification performance. Therefore, a machine learning technique that uses a greater variety of the linguistic features of the cue phrases than the pre-defined ones and makes the identification decisions based on the combinations these features, looks promising. The examples of the features are active or passive sentence styles and the position of keywords in the sentence. This paper presents the results of developing such automastic identification approach.

Keywords: Human and Design Errors, natural language processing, supervised learning approach.

1 Introduction

Failures of human operators to deal with the demands of controlling complex systems, such as automated aircraft, has become a major contributor to accidents. That is, the accidents are caused or contributed to by the limitations in the designs that normally have no effect on operator performance but which, under certain circumstances, can lead to acute or chronic deterioration of operator performance that can lead to active failure on the part of the operator. Research has found that over 66% of aviation

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 131 – 142, 2006.

accidents are attributed to Human Errors [4]. However, in some cases, the real causes to such accidents should be forwarded to the design of aircraft. To be able to design safer systems, the designers need to understand the role that automated functions and human-computer interfaces may have played in contributing to unsuitable operator actions leading to the accidents.

Accident reports are an essential resource to understand interaction failures between systems and operators. These are published as a result of an investigation carried out by a team of specialists when an accident happens. The facts, conditions, circumstances, and probable causes of the accidents are the main contents of the reports. Upon completing the investigations, the reports are put on public Web sites usually maintained by governments. Previous research has investigated the limitations of using the accident reports for identifying the findings about Human Errors induced by Design Errors [5,12]. The investigation has identified two main reasons. The first reason is that current accident reports do not aim specifically at capturing the information in such a way that to inform the designers effectively of the previous failures. For example, the systems do not include a search option that classifies the reports contents according to the identified causes making it easy to extract the accidents caused by Human Errors. The second reason is that the systems do not provide an efficient summary that highlights important information. Therefore, it is a time-consuming task to manually read the reports since each report is generally over 2-3 pages containing over 80 sentences. The identification of Human Errors and Design Errors using certain cue phrases, e.g. mismanaged, is easy to implement. However, since the descriptions of Human Errors and Design Errors are mostly implicit and ambiguous, it is difficult to automatically expand such hand-crafted cue phrases with their variants. A learning approach that learns the identification rules from examples, uses a greater variety of linguistic features and makes the identification decisions based on the combinations of these features, looks promising. This paper presents the result of developing the identification approach using the Support Vector Machines (SVM). As an example of statistical learning theory, SVM is known to outperform other techniques, e.g. neural networks, in recognizing patterns from examples. Tests show an improved recall with comparable precision through automatically identifying the Human and Design Errors in the accident reports.

The overall aim of our research is to understand how to make more information available to engineering designers in a readily useable form. The specific aim of the research described in this paper is to understand how the information extracted from the accident reports can be made to be more accessible and usable for the designers. In particular, the investigation and experimentation with various linguistic features in classifying the sentences according to their semantic types, e.g. Human Errors, is a primary feature that distinguishes the proposed approach from other research.

2 Related Work

The classification of sentences according to their semantic types draws from various researches including Information Extraction (IE), cue phrase identification and sentence-level classifications. IE, which is a sub-field of Natural Language Processing

(NLP) has been commonly used to extract domain entities of interest, e.g. person name or date, from unstructured texts [8]. IE uses shallow NLP techniques, e.g. Part-Of-Speech (POS) taggings, and stores the extracted information into database-like structures. IE performs well in extracting domain entities using lexical-syntactic patterns, i.e. person name is preceding a title (Mr.) and starts with capital letter. However, IE is not suitable for our research since the descriptions of Human Errors and Design Errors are implicitly expressed making it difficult to create the extraction rules. In addition, traditional IE systems aim to extract all the entities as completely as possible, whereas the proposed approach identifies a small number of sentences that are deemed as related to Human Errors or Design Errors.

Cue phrase identification has been used when it is difficult to find such syntactic-lexical patterns [1,13]. For example, by detecting the word 'but', a 'contrast' discourse relation between two adjacent texts can be identified. Whereas it is easy to implement the cue phrase, due to syntactic and semantic variations of it, e.g. the word 'mismanaged' can be rephrased as 'did not properly manage', relying entirely on the cue phrase can lead to low coverage and ambiguity. To address this problem, Abdalla and Teufel proposed a bootstrapping approach that incrementally enriched the cue phrase with variants [1]. The cue phrases tested were pairs of transitive verbs and objects, e.g. introduce and method. While the method demonstrated high accuracy, it is not suitable for our task: although the pairs of verb and object are useful, other types of cue phrases, i.e. nouns (mismanagement), or verbs without objects (did not properly manage), are included in our case.

Text classification systems take a text as an input and assigns pre-defined categories to the text. To do this the whole content of the text is compared to the summary of the each category. If the content is similar to the summary, and then the text is classified as belonging to the category. Recently, more studies have been conducted at the sentence-level classifications. Sentence classification is the automatic classification of sentences into pre-defined sentence types [9]. Example applications of the sentence classifications are automatic text summarization and semantic orientations. The objective of summarization systems is to create a shorten version of an original text in order to reduce the time spent reading and comprehending it. The extraction of important sentences is one of the common tasks for the summarization systems. There are two commonly used methods. First method is to identify the significance of words in the original text and to select set of sentences based on the occurrence of high-score words [24]. Second method is to use adverbs and adjectives, e.g. significant and hardly, and to exploit the positions of the sentences in the text [10]. The performance of the automatic systems is often measured by comparing the automatic summaries with the summaries generated by humans.

Semantic orientation looks for the evaluative character of a word, in order to extract opinions, feelings, and attitudes expressed in a text [22]. The orientation is classified as positive if it contains praise or recommendation. Negative orientation indicates criticism or non-recommendation. The semantic orientation does not apply to the sentences that contain only facts. Wiebe and Riloff [25] proposed a classification method that classifies a sentence as subjective if the sentence expresses a positive or negative opinion, otherwise as objective. A combination of cue phrases, e.g. excellent, low fees, and linguistic features is commonly used [23]. Those cue phrases can be created either manually or using a learning technique, e.g. PointWise Mutual Information (PMI)-IR or naive Bayes. On average, the accuracy is observed at 70%.

3 Developing the Proposed Approach to Human and Design Errors Identification

The proposed approach makes dual classification decisions: (1) the approach classifies a given sentence as relevant to Human Errors or Design Errors, and (2) the approach classifies the relevant sentence into either Human Errors or Design Errors.

3.1 Example

Figure 1 compares two approaches of searching for the information related to Human Errors and Design Errors on the accident reports. The diagram (a) in Figure 1 shows a current practice that the search is done by manually. It identifies Human Errors, i.e. the mismanagement of the fuel system by the pilot, and Design Errors, i.e. different design models between the fuel systems. The diagram (b) in Figure 1 shows how the contents in the diagram (a) could be analysed using four different levels of interpretation supported by the proposed approach. Using the approach, it is easy to establish the facts of the accidents and to gain insights as well. For example, two design opportunities are identified that could prevent the reoccurrence of the same Human Errors. These are the establishment of two new design standards for: (1) the

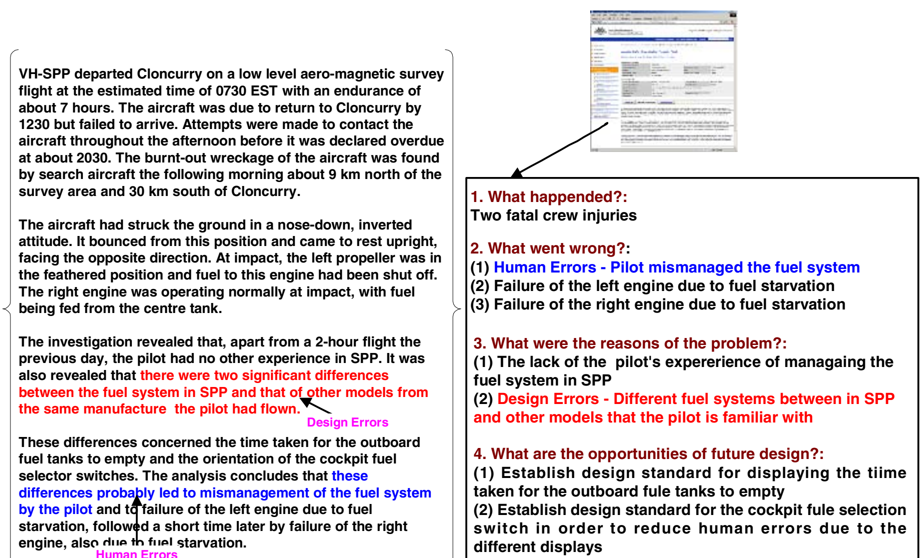

Diagram (a) Diagram (b)

Fig. 1. A comparison of two approaches accessing the accident reports

gauge displaying the time taken to empty a fuel tank; and (2) the switch for cockpit fuel selection. In doing so, it is easy not only to establish the facts toward conclusions but also to gain insights. Both of them are related to establish new design standards of the fuel systems in the same manufacture in order to prevent the reoccurrence of the same Human Errors. The example text used in both diagrams is the excerpt of the accident report occurred in 1994[1]. The report was extracted from the Australian Transport Safety Bureau (ATSB) database. The text is used as an example throughout this paper.

3.2 Dataset

It is important to base the proposed method on solid empirical evidence. To do this, we used the dataset created by Shin et al [22]. The dataset contains 50 aviation accident reports downloaded from the ATSB database. The accidents occurred between 1994 and 2000. A total of 3995 sentences were extracted from the reports and 208 sentences, i.e. 5% of the 3995 sentences, were tagged as either Human Errors or Design Errors. A single sentence can have more than one Human Errors or Design Errors. 113 sentences are classified as Human Errors and 114 sentences are classified as Design Errors. Sentences that do not have such taggings are deemed as irrelevant, i.e. Non Errors.

Precision, recall and F-measure are used for measuring the performance of the proposed approach. Precision is the percentage of the sentences that were automatically classified into Human Errors (or Design Errors) that are truly Human Errors (or Design Errors). Recall is the percentage of Human Errors (or Design Errors) that were automatically classified as Human Errors (or Design Errors). F-measure is used to compare the performance of different approaches since it is a single value of the harmonic mean between precision and recall [6].

3.3 Classification Using Cue Phrases

Cue phrases, either a single word, e.g. *mismanaged*, or multiple words, e.g. *significant differences*, carry semantics that are used to identify the types of sentences. For example, by recognizing the word *mismanaged* in a sentence, the sentence is classified into Human Errors. Systems that use cue phrases rely on hand-crafted lists, the acquisition of which is a time-consuming and error-prone task. Therefore, a semi-automatic extraction of the cue phrases is necessary. To do this we apply a syntactic parser, i.e. Apple Pie Parser [21] into the sentences classified either as Human or Design Errors.

The Apple Pie Parser generates POS taggings, e.g. NN for a singular noun, and phrase chunks, e.g. NP for a noun phrase, for an input sentence. It refers to the grammars defined in the Penn Treebank to determine the POSs [14]. We use simple regular expressions in order to extract the cue phrases based on the results of the syntactic parse. The most commonly occurred pattern is the negated sentence, e.g. 'did not properly manage' that is a syntactic pattern of <do or have> not <adverb> {verb}. The pattern has variations, e.g. <do or have> not {verb}, e.g. 'did not

[1] http://www.atsb.gov.au/publications/investigation_reports/1994/AAIR/pdf/aair199403314_001.pdf

manage', <do, be or have> <adverb> not {verb}, e.g., 'was apparently not managed', or {verb} <adverb>, e.g. 'managed badly'. Table 1 shows the cue phrases.

Table 1. Cue phrases for classifying Human Errors and Design Errors

Cue phrases for Human Errors	Cue phrases for Design Errors
<do or have> not {recognize, detect, notice, realize, scan, mismanage, misunderstand}	<do or have> not {alert, warn, provide, have}
{distract, distraction}	{workload, the modification to, amended>

As mentioned in the Section 3, the proposed approach makes dual classification decisions. For the first decision, it uses the combination of the cue phrases in Table 1. That is, if a given sentence is not matched with any of the cue phrases, and then it is classified as Non Errors. For the second decision, it tags a sentence as Human Errors if the sentence contains any of the cue phrases. Similarly, a sentence is classified as Design Error by looking for the existence of the cue phrases.

Using the cue phrases in Table 1, the precision of Non Errors achieved 82% with 41% recall. The precision of Human Errors achieved 91% with 39% recall. The precision of Design Errors was 85% with 35% recall. Overall, all three identifications show high precision but with low recall. The example of missed Human Errors is the following sentence 'The crew interpreted taxiway G3 to be taxiway Y on the basis of that information'. The precision of the Design Errors, i.e. 85%, is lower than the precision of Human Errors, i.e. 91%. It implies that some of the cue phrases for Design Errors are not 'accurate'. For example, although the word 'workload' is an good indicator to signal Design Errors, since it is also used to express potential causes that explaining why or how the Human Errors have been generated, classifying a sentence into Design Errors by looking up the existence of the word 'workload' can be misleading.

The problem of increasing recall while maintaining high precision is due to the difficulty of expanding the cue phrases with their variants using string-based patterns. For example, whereas the string 'did not conduct' is commonly identified in Human Errors, including such string into the cue phrases has decreased the precision while slightly improved the recall. Therefore, the problem of string-based patterns is that they are inefficient to encode syntactic generalizations, i.e. linguistic features, such as active or passive sentence constructions, the occurrence of domain entities or the position of the cue phrases in the sentence.

Since the classification using the pre-defined cue phrases shows a limited performance, it is necessary to extract the identification rules that are derived from a greater variety of the linguistic features of the cue phrases than the pre-defined ones. Supervised learning methods that automatically construct the identification rules from tagged examples are particularly suitable for this research since we have a handful of tagged examples.

3.4 Classification Using Support Vector Machines (SVM)

Machine learning algorithms require example documents to be in a form that can be processed by them. A commonly used conversion is to extract indexing keywords from the document contents and to associate indexing weighting to each of the indexing keywords. The weighting measures the quality of the indexing keyword in relation to an effective identifier of the document and specifies the numeric contribution. That is, each document is represented as pairs of indexing keywords and the associated numeric weighting. In doing so, the string-based indexing has been commonly used. Whereas such indexing is easy to implement, its inability to denote sufficient descriptions of the concept can lead to semantic ambiguity. Recent studies have emphasized the importance of encoding the example documents using various linguistic features [6,7].

In this paper, the linguistic features include unigram (one keyword), bigrams (two keywords), POS taggings, the information on sentence constituents including subjects, verbs, and objects, active or passive sentence constructions, Named-Entities (NE) or verb tense. For example, the following features are encoded for the sentence 'It was also revealed that there were two significant differences between the fuel system in SPP ands that of other Aero Commander models the pilot had flown':

Unigram: reveal, there, two, significant, difference, between, fuel, system, SPP other, Aero, Commander, model, pilot, flow

Bigrams: significant difference, fuel system, Aero Commander, Commander model, pilot flow

POS taggings: It/PRP, was/VBD, also/RB, revealed/VBN, that/IN, there/RB, were/VBD, two/CD, significant/JJ, differences/NNS, between/IN, the/DT, fuel/NN system/NN, in/IN, SPP/NNPX, and/CC, that/DT, of/IN, other/JJ, Aero/NNPX, Commander/NNP, models/NNS, the/DT, pilot/NN, had/VBD, flown/VBN

Subject: It
Verb: was revealed
Active/Passive: Passive
Verb tense: past

Subject: there
Verb: were
NEs: fuel system → Cockpit_Control_System
SPP, Aero Commander → Aircraft model
Pilot → Operator
Object: two significant differences…had flown.
Active/Passive: active
Verb tense: past

Both unigram and bigrams are extracted after removing common keywords, e.g. it, or also, based on the pre-defined stoplists. Each POS-tagged keyword, e.g. revealed is tagged as VBD (past tense verb), is compared with WordNet [15] definitions to achieve term normalization, e.g. revealed → reveal. Both keywords need special attention. A total of 31725 unigram including 23639 bigrams using the dataset in the Section 3.2 was extracted. Not only it is difficult to make use of such a large number of the keywords, but also due to the noisy in them, the identification performance can be decreased if all the keywords were used for indexing. It is well known that feature selection improves the accuracy of a classifier. The feature selection deletes noisy features and reduces the feature-space dimension. In general, the first *n* features based on one of the ranking criteria are selected and are assumed as more promising features for improving the identification performance. In this paper, the Information Gain (IG) measure is used for the feature selection [3]. The top 1000 unigram and bigrams based on the IG values are used for the testing. NEs are the domain entities and the identification of NEs is a part of the IE tasks. For example, SPP and Aero Commander are the types of Aircraft Models. The identification of NEs is included in the dataset.

Including the features above to the classification rules is important since it is difficult to expand cue phrases with their variants using string-based patterns as described in the Section 3.3. We use the SVM^{light} that is an implementation of the SVM in C programming language [11]. SVM^{light} requires an example to be represented as pairs of features and their numeric weightings. Term Frequency Inverse Document Frequency (TFIDF) method is used for the weightings [20]. 50% of the dataset was used for training SVM^{light} and the remaining 50% was used for testing the rules generated by the SVM^{light}. For the dual identification decisions of the proposed approach, the SVM is trained with two different examples. The first example dataset is to classify sentences either into Non Errors or Errors (Human Errors and Design Errors). The second example dataset is to classify sentences either into Human or Design Errors. Table 2 shows the identification results. P indicates precision, R denotes recall and F means F-measure. SVOs denote the features extracted from the subjects, verbs, and objects including the active or passive sentence constructions and verb tenses.

Both identification tasks using the unigram show lowest performance. Therefore, we have decided not to test the identification performance that involves the unigram. The best precision of the first identification, i.e. Non Errors vs Errors, is obtained using the combination of Bigrams, NEs, and SVOs, i.e. 94% precision with 61% recall. The F-measure for this precision and recall is 74%. The best precision of the second identification, i.e. Human Errors vs Design Errors, is achieved using the combination of Bigrams and NEs, i.e. 89% precision and 71% recall. The F-measure for this precision and recall is 79%. Recall for both identification decisions is significantly improved in compared to the cue phrase-based identifications mentioned in the Section 3.3 with comparable precision, i.e. up to 20% increase for the first identification and up to 34% increase for the second identification.

Table 2. Identification results using the SVM

Linguistic Feature	Non Errors vs Errors identification			Human Errors vs Design Errors identification		
	P	R	F	P	R	F
Unigram	75%	42%	64%	63%	49%	55%
Bigrams	93%	56%	70%	79%	67%	73%
Bigrams + NEs	85%	53%	65%	**89%**	**71%**	**79%**
Bigrams + SVOs	87%	55%	67%	52%	67%	59%
NEs + SVOs	97%	44%	61%	80%	56%	66%
Bigrams + NEs + SVOs	**94%**	**61%**	**74%**	73%	58%	65%

3.5 Evaluating Alternative Techniques

We compared the SVM with two other machine techniques, i.e. Inductive Programming (ILP) and probability-based classification. As an example of ILP, Progol [17] is used and naïve Bayesian is used as an example for the probability-based classification [16]. Progol has been used for learning extraction rules for NEs [2] and classification rules for Web pages [19]. Naïve Bayesian probability has been commonly used for text classifications [25].

Progol represents examples using the first-order logic forms meaning that it can learn more complex, structured, or recursive descriptions and generates the outputs as sets of IF-THEN rules. Naïve Bayesian probability is based on a simplified theorem that assumes variables to be independent in each class. It compares the linguistic features of a new sentence s_i with those of Human Errors c_1 and Design Errors c_2 and computes the probability $P(c_j \mid s_i)$ for them.

Both techniques were tested with the second identification decision, i.e. classifying the sentences into either Human Errors or Design Errors. Using the IF-THEN rules by Progol, the precision is shown at 82% with 45% recall. Using the probability-based rules by the naïve Bayesian, the precision is shown at 76% with 55% recall. The F-measure for the Progol classification is 58%. The F-measure for the naïve Bayesian is 64%. Therefore, the F-measure of the SVM, i.e. 79%, outperforms those of the both methods.

4 Testing with a New Dataset

Since the proposed approach learns the identification rules from tagged examples, there might be biases introduced by the rules. That is, if the examples are similar to one another and less heterogeneous than the sentences that the rules will ultimately applied to, the same level of performance is not expected to be repeated. Therefore, we decided to test the proposed approach with a new dataset. We exploit the large amount of information available on the Web for collecting the dataset in order to reduce the time and effort spent on the manual dataset collection. To do this, we use the Google search engine to automatically search for the accident reports containing Human Errors and Design Errors using the following steps:

Step 1: Query the Google search engine

We send queries with keywords that are extracted from the dataset in the Section 3.2. The keywords were sorted as the order of frequencies and the top seven keywords were selected: *pilot controller aircraft plane occurrence date time*. The Google search returns 263000 hits when it searches on those seven keywords. For this experiment, the top 40 ranked return were used.

Step 2: Analyse the top 40 documents retrieved by the Google

Each retrieved document is decomposed into paragraphs, each of which is segmented into sentences. Each sentence is analysed and encoded with the linguistic features in the Section 3.4. A total of 2355 sentences were extracted from the 40 documents after removing the embedded HTML tags. We manually examined those sentences and classified them into Human Errors, Design Errors or Non Errors. A total of 57 sentences were identified as Human Errors, 60 sentences were tagged as Design Errors and the remaining sentences were tagged as Non Errors.

Step 3: Classify the sentences using the proposed approach

The proposed approach uses the SVM classification rules that are learned from the dataset in the Section 3.2. The rules are based on the combinations of the features that showed the best performance in Table 3. The precision for the first classification task (Non Errors vs Errors) is observed at 82% with 52% recall. The F-measure for this precision and recall is 64%. The precision for the second classification task (Human Errors vs Design Errors) is observed at 85% with 61% recall. The F-measure for this precision and recall is 71%. Overall, the performance of the proposed approach with the new dataset is slightly lower than the precision and recall of it with the existing dataset, i.e. 10% decrease in F-measure for the first classification task and 8% decrease in F-measure for the second classification task. However, it is noticeable for this relatively small dataset size of 208 sentences (in the existing dataset) that the performance is relatively good at around 64%-71% for the new dataset.

5 Conclusions

It is important to recognise human factors that are reported to be related to the occurrence of aviation accidents when designing human system interactions. In doing so, it aims to reduce the reoccurrence of the accidents caused by the same human factors. Aviation accident reports play a critical role of obtaining such information. However, since the reports do not specifically target at providing information to the interface designers, the usability of the reports to learn how to prevent such accidents is limited. The main objective of this paper is to investigate and experiment with a machine learning technique in order to make more information on the human factors available to the designers in a readily useable form.

To do this our initial intention was to extract cue phrases that are easy to identify and implement. However, a preliminary test with the cue phrases showed high precision but with low recall. In order to increase the recall, the learning technique based on the SVM was used. The identification rules generated by the SVM exploited a greater variety of linguistic features of the cue phrases and determined the identification decisions based on the combinations of these features. Tests showed improved recall with comparable precision, i.e. 61% recall with 94% precision for the first classification task, and 71% recall with 89% precision for the second classification task. The performance of the SVM also outperforms those of other two commonly used learning techniques. The test with a new dataset demonstrated that the SVM classification was reliable and good at maintaining high precision and recall showing slight performance drops compared to the performance with the existing dataset.

Acknowledgments. This work was funded by the University Technology Partnership for Design, which is a collaboration between Rolls-Royce, BAE SYSTEMS and the Universities of Cambridge, Sheffield and Southampton. We specially thank InJae Shin of the University of Bath for the dataset and comments.

References

1. Abdalla, R. M., Teufel, S.: A Bootstrapping Approach to Unsupervised Detection of Cue Phrase Variants. In Proc. on the Association for Computational Linguistics (ACL), Australia, (2006).
2. Aitken, J. S.: Learning information extraction rules: An inductive logic programming approach. In Proc. European Conference on Artificial Intelligence ECAI, France, (2002), 335-359
3. Ayan, N. F.: Using Information Gain as Feature Weight. In Proc. 8th Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN'99), Turkey, (1999).
4. Berninger, D. J.: Understanding the Role of Human Error in Aircraft Accidents. In Transportation Research Record No. 1298 (1990).
5. Bruseberg, A., Johnson, P.: Understanding Human Error in Context: Approaches to Support Interaction Design Using Air Accident Reports, In Proc. 12th Int. Symposium on Aviation Psychology, U.S.A. (2003), 166-171.
6. Forman, G.: An Extensive Empirical Study of Feature Selection Metrics for Text Classification. Technical Report, HPL-2002-147R1 (2002).
7. Grimaldii, M., Cunningham, P., Koraram, A.: An Evaluation of Alternative Feature Selection Strategies and Ensemble Techniques for Classifying Music. Technical Report, TCD-CS-2003-21, University if Dublin (2003).
8. Grishman, R.: Information Extraction: Techniques and Challenges (1997). In Lecture Notes in Artificial Intellgence, Vol. 1299.
9. Hackey, B., Grover, C.: Sequence modeling for sentence classification in a legal Summarisation System. In ACM symposium on Applied computing. (2005). U.S.A. 292-296
10. Hovy, E., Lin, C. Y.: Automated Text Summarisation in Summarist. In Advances in Automated Text Summarisation, Mani, I., Maybury, M. (editors). (1999).
11. Joachims, T.: Making large-Scale SVM Learning Practical. In Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.) (1999).
12. Johnson, C.: A First Step Towards the Integration of Accident Reports and Constructive Design Documents. In Proc. on SAFECOMP (1999)
13. Knott, A., Dale, R.: Using linguistic phenomena to motivate a set of coherence relations. In Discourse Processes, 18(1), 35-62 (1995).
14. Macus, M.P., B. Santorini, M., Marcinkiewicz, A. M.: Building a Large Annotated Corpus of English: The Penn Treebank. In Computational Linguistics. 19(2). (1994). 313-330.
15. Miller, G. A., Beckwith, R.W., Fellbaum, C., Gross, D., & Miller, K.: Introduction to wordnet: An on-line lexical database, In Int. Journal of Lexicography, 3(4), (1993). 235-312.
16. Mitchell, T.: Machine Learning, McGraw Hill. (1997).
17. Muggleton, S.: Inverse entailment and progol. *New Generation Computing*. (1995) 13:245-286
18. Nasukawa, T., Yi, J.: Sentiment Analysis: Capturing Favorability Using Natural Language Processing. In Int. Conf. on Knowledge Capture. (2003). U.S.A. 70-77
19. Parson, R. and Muggleton, S.: An experiment with browsers that learn, In K. Furukawa, D. Michie and S. Muggleton (Eds.), Machine Intelligence, 15, Oxford University Press, (1998).
20. Salton, G.: Advanced Information-Retrieval Models, In Automatic Text Processing, (Salton, G. Ed.), Addison-Wesley Publishing Company, (1989).

21. Sekine S., Grishman R.: A Corpus-Based Probabilistic Grammar with only two Non-Terminals In Fourth Int. Workshop on Parsing Technologies. (1995). 216-223.
22. Shin, I., Kim S., Busby, J.S., Hibberd, R.E. and Mcmahon, C.A. (2006).: An application of semantic annotations to design error. In IEEE International Conference on Hybrid Information Technology, Korea, (in press).
23. Turney, P. D.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In Proc. 40th Association of the Computational Linguistics (ACL), U.S.A. (2002). 417-424.
24. Strzalkowski, T., Wang, J., Wise, B.: A Robust Practical Text Summarisation System. In AAAI Intelligent Text Summarisation Workshop. (1998). U.S.A. 26-30.
25. Wiebe, J., Riloff, E.: Creating Subjective and Objective Sentence Classifiers from Unannotated Texts. In Int. Conf. on Computational Linguistics and Intelligent Text Processing, (2005), Invited Paper, Springer LNCS Vol. 3406 © Springer-Verlag.

Text Mining Through Semi Automatic Semantic Annotation

Nadzeya Kiyavitskaya[1], Nicola Zeni[1], Luisa Mich[2],
James R. Cordy[3], and John Mylopoulos[4]

[1] Dept. of Information and Communication Technology, University of Trento, Italy
{nadzeya, nzeni}@dit.unitn.it
[2] Dept. of Computer and Management Sciences, University of Trento, Italy
luisa.mich@unitn.it
[3] School of Computing, Queens University, Kingston, Canada
cordy@cs.queensu.ca
[4] Dept. of Computer Science, University of Toronto, Ontario, Canada
jm@cs.toronto.edu

Abstract. The Web is the greatest information source in human history. Unfortunately, mining knowledge out of this source is a laborious and error-prone task. Many researchers believe that a solution to the problem can be founded on semantic annotations that need to be inserted in web-based documents and guide information extraction and knowledge mining. In this paper, we further elaborate a tool-supported process for semantic annotation of documents based on techniques and technologies traditionally used in software analysis and reverse engineering for large-scale legacy code bases. The outcomes of the paper include an experimental evaluation framework and empirical results based on two case studies adopted from the Tourism sector. The conclusions suggest that our approach can facilitate the semi-automatic annotation of large document bases.

Keywords: semantic annotation, large-scale document analysis, conceptual schemas, software analysis.

1 Introduction

The Web is the greatest information source in human history. Unfortunately, mining knowledge out of this source is a laborious and error-prone task, much like looking for the proverbial needle in a haystack. Many researchers believe that a solution to the problem can be founded on semantic annotations that need to be inserted in web-based documents and guide information extraction and knowledge mining. Such annotations use terms defined in an ontology. We are interested in knowledge mining the Web, and use semantic annotations as the key idea in terms of which the mining is to be done.

However, adding semantic annotations to documents is also a laborious and error-prone task. To help the annotator, we are developing tools that facilitate the annotation process by making a first pass at the documents, inserting annotations on the basis of textual patterns. The annotator can then make a second pass improving

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 143 – 154, 2006.

manually the annotations. The main objective of this paper is to present a tool-supported methodology that semi-automates the semantic annotation process for a set of documents with respect to a semantic model (ontology or conceptual schema). In this work we propose to approach the problem using highly efficient methods and tools proven effective in the software analysis domain for processing billions of lines of legacy software source code [2]. In fact, document analysis for the Semantic Web and software code analysis have striking similarities in their needs:

- robust parsing techniques, given that real documents rarely match given grammars;
- a semantic understanding of source text, on the basis of a semantic model;
- semantic clues drawn from a vocabulary associated with the semantic model;
- contextual clues drawn from the syntactic structure of the source text;
- inferred semantics from exploring relationships between identified semantic entities and their properties, contexts and related other entities.

On the basis of these considerations, we have adapted software analysis techniques to the more general problem of semantic annotation of text documents. Our initial hypothesis is that these methods can attain the same scalability for analysis of textual documents as for software code analysis. In this work we extend and generalize the process and architecture of the prototype semantic annotation tool presented earlier in [3]. The contribution of this work includes also an evaluation framework for semantic annotation tools, as well as two real-world case studies: accommodation advertisements and Tourist Board web sites. For the first experiment, we use a small conceptual schema derived from a set of user queries. For the second experiment, we adopt more elaborated conceptual schemas reflecting a richer semantic domain.

Our evaluation of both applications uses a three-stage evaluation framework which takes into account:

- standard accuracy measures, such as Recall, Precision, and F-measure;
- productivity, i.e. the fraction of time spent for annotation when the human is assisted by our tool vs. time spent for manual annotation "from scratch"; and
- a calibration technique which recognizes that there is no such thing as "correct" and "wrong" annotations, as human annotators also differ among themselves on how to annotate a given document.

The rest of the paper is organized as follows. Our proposed annotation process and the architecture of our semantic annotation system are introduced in section 2. The two case studies are presented in section 3, and section 4 describes the evaluation setup and experimental results. Section 5 provides a short comparative overview of semantic annotation tools and conclusions are drawn in section 6.

2 Methodology

Our method for semantic annotation of documents uses the generalized parsing and structural transformation system TXL [4], the basis of the automated Year 2000 system LS/2000 [5]. TXL is a programming language specially designed to allow by-example rapid prototyping of language descriptions, tools and applications. The system accepts as input a grammar and a document, generates a parse tree for the input document, and applies transformation rules to generate output in a target format. The architecture of our solution (Fig. 1) is based on the LS/2000 software analysis architecture, generalized to allow for easy parameterization by a range of semantic domains.

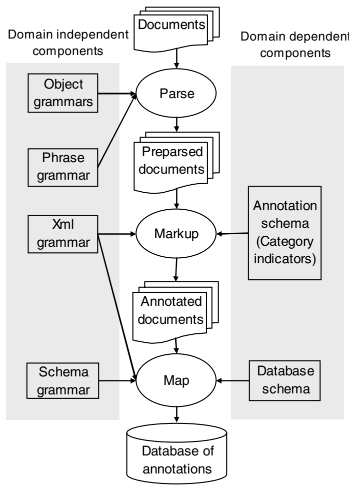

Fig. 1. Architecture of our semantic annotation process

The architecture explicitly factors out reusable domain independent knowledge such as the structure of basic entities (email and web addresses, dates, and other word-equivalent objects) and language structures (document, paragraph, sentence and phrase structure), shown on the left hand side, while allowing for easy change of semantic domain, characterized by vocabulary (category word and phrase lists) and semantic model (entity-relationship schema and interpretation), shown on the right.

The process consists of three phases. In the first stage, an approximate ambiguous context-free grammar is used to efficiently obtain an approximate phrase structure parse of the source text using the TXL parsing engine. Using robust parsing techniques borrowed from compiler technology [6] this stage results in a deterministic maximal parse. As part of this first stage, basic entities are recognized. The parse is linear in the length of the input and runs at compiler speeds.

In the second stage, initial semantic annotation of the document is derived using a wordlist file specifying both positive and negative indicators for semantic categories. Indicators can be both literal words and phrases and names of parsed entities.

Phrases are marked up once for each category they match – thus at this stage a sentence or phrase may end up with many different semantic markups. Vocabulary lists are derived from the semantic model for the target domain. This stage uses the structural pattern matching and source transformation capabilities of the TXL engine similarly as for software markup to yield a preliminary marked-up text in XML form.

The third stage uses the XML marked-up text to populate an XML database schema, derived from the semantic model for the target domain. Sentences and phrases with multiple markups are "cloned" using TXL source transformation to appear as multiple copies, one for each different markup, before populating the database. In this way we do not prejudice one interpretation as being preferred.

The outputs of our process are both the XML marked-up text and the populated database. The populated database can be queried by a standard SQL database engine.

3 Experimental Case Studies

Our case studies involve two applications in the Tourism area. Tourism is a very broad sector of economy which comprises many heterogeneous domains: accommodation and eating structures, sports, means of transport, historical sites, tourist attractions, medical services and other areas of human activity. Information available from heterogeneous data sources must be integrated in order to allow effective interoperability of tourism information systems and to enable knowledge mining for the variety of roles and services that characterize such a compound sector (e.g. composition of services for tourist packages). This is where semantic annotations come in handy.

3.1 Accommodation Ads

As a first full experiment in the application of our new method, we have been working in the domain of travel documents, and in particular with published advertisements for accommodation. This domain is typical of the travel domain in general and poses many problems commonly found in other text markup problems, such as: partial and malformed sentences; abbreviations and short-forms; location-dependent vocabulary; monetary units; date and time conventions, and so on.

In the first case study we used a set of several hundred advertisements for accommodation in Rome drawn from an online newspaper. The task was to identify and mark up the categories of semantic information in the advertisements according to a given accommodation conceptual schema (Fig. 2), which was reduced by hand to an XML schema for input to our system. The desired result was a database with one instance of the schema for each advertisement in the input, and the marked-up original advertisements. To adapt our semantic annotation methodology to this experiment the domain-related wordlists were constructed by hand from a set of examples.

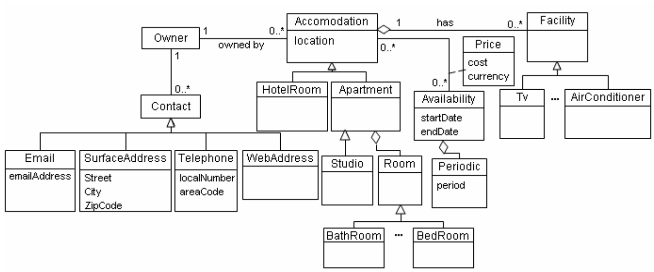

Fig. 2. Conceptual schema for accommodation ads

3.2 Tourist Board Web Pages

In the second case study we pursued two main goals: to demonstrate the generality of our method over different domains, and to verify the scalability of our approach on a richer semantic model and larger natural language documents. For this purpose, we considered the web sites of Tourist Boards in the province of Trentino (Italy)[1] as input documents. In contrast to the classified ads, this domain presents a number of specific problematic issues: free unrestricted vocabulary; differently structured text; a rich semantic model covering the content of web sites.

This experiment was run in the collaboration with the marketing experts of the eTourism[2] group of University of Trento. From the point of view of tourist marketing experts in tourism, the high-level business goal of this case study was to assess the communicative efficacy of the web sites based on content quality or *informativity*, that is, how comprehensively the web site covers relevant topics according to the strategic goals of the Tourist Board.

In order to assess the communicative efficacy we performed semantic annotation of the web pages revealing the presence of information important for a Tourist Board web site to be effective. The list of semantic categories and their descriptions was provided by the tourism experts (Fi. 3).

Geography
- Climate
- Weather predictions
- Land Formation
- Lakes and Rivers
- Landscape

Local products
- Local handcrafting
- Agricultural products
- Gastronomy

Culture
- Traditions and customs
- Local history
- Festivals
- Population
- Cultural institutions and associations
- Libraries
- Cinemas
- Local literature
- Local prominent people

Artistic Heritage
- Places to visit: museums, castles
- Tickets, entrance fees, guides

Sport
- Sporting events
- Sport infrastructure
- Sport disciplines

Accommodation
- Places to stay
- How to book
- How to arrive
- Prices
- Availability

Food and refreshment
- Places to eat
- Dishes
- Degustation
- Time tables
- How to book

Wellness
- Wellness centers
- Wellness services

Services
- Transport, schedules
- Information offices
- Terminals, stations, airports
- Travel agencies

Fig. 3. Relevant categories for communicative efficacy of a Tourist Board web site

In this second experiment, we adapted our annotation framework to the new domain by replacing the domain-dependent components with respect to this specific task. For this purpose, the initial rough schema provided by the domain experts was transformed into a richer conceptual schema consisting of about 130 concepts systematized into a hierarchy and connected by semantic relations (see the partial view in Fig. 4[3]).

Domain dependent vocabulary was derived semi-automatically, expanding concept definitions with the synonyms provided by the *WordNet*[4] database and on-line

[1] http://www.trentino.to/home/about.html?_area=about&_lang=it&_m=apt
[2] http://www.economia.unitn.it/etourism
[3] The visualization tool RDFGravity: http://semweb.salzburgresearch.at/apps/rdf-gravity/
[4] http://wordnet.princeton.edu

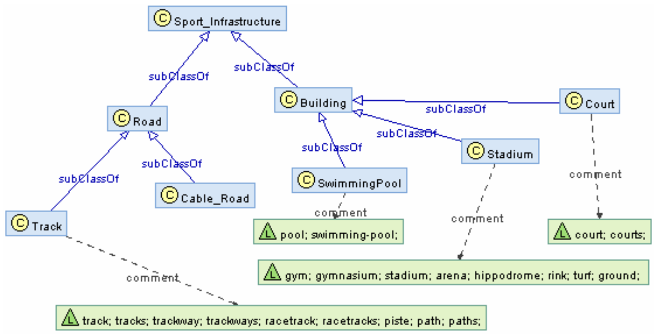

Fig. 4. A slice of the conceptual schema showing semantic (placement in the hierarchy, relationships, attributes) and syntactic (keywords or patterns) information associated with concepts. This view shows only *is-a* relations, because this type of relation is essential in guiding the annotation process. The complete model includes many more relations apart from taxonomical ones.

Thesaurus[5] and mined from a set of sample documents. The total number of keywords collected was 507 and an additional four object patterns were re-used from previous application to detect such entities as monetary amounts, e-mails, web addresses and phone numbers.

To begin this experiment we downloaded the English version of 13 Tourist Board web sites using an offline browser software[6]. For some of them (which are generated dynamically) we had to apply a manual screen-scraping technique. Then two human annotators and the tool were given 11742 text fragments for annotation. The required result was a database with one instance of the schema for each Tourist Board web site, and the marked-up original text (Fig. 5).

```
   <FoodAndRefreshment>Bread and wine snack in the shade of an elegant
park.</FoodAndRefreshment>
   <FoodAndRefreshment>Dinner at the “La Luna Piena” restaurant,
consisting of the “Il Piatto del Vellutaio”</FoodAndRefreshment>
   <ArtisticHeritage>Museo del Pianoforte Antico: guided visit and
concert proposed within the “Museum Nights” programme on the 3, 10, 17
and 24 of August.</ArtisticHeritage>
```

Fig. 5. Example of XML-marked up content of a tourism web site

4 Experimental Evaluation

4.1 Evaluation Framework

The performance of semantic annotation tools is usually evaluated similarly to information extraction systems, i.e. by comparing with a reference correct markup and calculating recall and precision metrics.

In order to evaluate our initial experimental results, we designed a three stage validation process. At each stage, we calculated a number of metrics [7] for the tool's

---

5 http://thesaurus.reference.com

6 SurfOffline 1.4: http://www.bimesoft.com

automated markup compared to manually-generated annotations: *Recall* evaluates how well the tool performs in finding relevant items; *Precision* shows how well the tool performs in not returning irrelevant items; *Fallout* measures how quickly precision drops as recall is increased; *Accuracy* measures how well the tool identifies relevant items and rejects irrelevant ones; *Error rate* demonstrates how much the tool is prone to accept irrelevant items and reject relevant ones; *F-measure* is an harmonic mean of recall and precision.

In the first step of our evaluation framework, we compare the system output directly with manual annotations. We expect that quality of manual annotations constitutes an upper bound for automatic document analysis. Of course, this type of evaluation can't be applied on a large scale for cost reasons.

In the second step, we check if the use of automatic tool increases the productivity of human annotators. We note the time used for manual annotation of the original textual documents and compared it to the time used for manual correction of the automatically annotated documents. The percentage difference of these two measures shows how much time can be saved when the tool assists the human annotator.

Finally, in our third step we take into account disagreement between annotators to interpreted the automatically obtained annotation. Then, we compare system results against the final human markup made by correcting the automatically generated markup.

4.2 Experimental Results

Experiment 1: Accommodation Ads. The details of our evaluation for the accommodation ads application can be found in [2]. We only say that as a result of this first experiment, even without local knowledge and using a very small vocabulary and only few TXL rules, we obtained results comparable to some of the best heavyweight methods, albeit on a very limited domain. Performance of our untuned experimental tool was also already very fast, handling for example 100 advertisements in about 1 second on a 1 GHz PC.

Experiment 2: Tourist Board Web Pages. As the semantic model in this experiment was fairly extensive, we could not afford humans to handle properly all of the entities of the rich domain schema. Accordingly, in our evaluation we considered only general categories in the annotation schema (*Geography, Sport, Culture, Artistic Heritage, Local Products, Wellness, Accommodation, Food and Refreshment, Services*). For these we performed simple metrics-based validation (Tables 1a, b, c) and calibration of the results taking into account inter-annotator disagreement (Table 2) for the entire set of 11742 paragraphs.

Table 1a. Evaluating system annotation vs. human Annotator 1

Topic / Measure	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	68.23	68.18	72.49	82.28	82.57	83.19	68.29	16.67	76.42
Precision	85.62	82.19	93.16	97.38	78.35	96.12	94.92	50.00	91.01
Fallout	0.59	0.34	0.50	0.19	1.50	0.11	0.08	0.03	0.43
Accuracy	97.88	98.95	97.16	98.39	97.52	99.39	99.26	99.85	98.31
Error	2.12	1.05	2.84	1.61	2.48	0.61	0.74	0.15	1.69
F-measure	75.94	74.53	81.53	89.19	80.40	89.19	79.43	25.00	83.08

Table 1b. Evaluating system annotation vs. human Annotator 2

Topic / Measure	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	42.19	59.09	74.85	70.57	73.86	68.91	40.24	16.67	59.43
Precision	69.83	82.54	59.81	59.31	62.24	50.62	55.93	33.33	33.96
Fallout	0.94	0.29	4.75	4.25	2.94	2.11	0.68	0.05	6.62
Accuracy	96.27	98.80	93.48	93.71	95.63	97.01	98.08	99.82	91.54
Error	3.73	1.20	6.52	6.29	4.37	2.99	1.92	0.18	8.46
F-measure	52.60	68.87	66.49	64.45	67.55	58.36	46.81	22.22	43.22

Table 1c. Evaluating system annotation vs. humans – average category scores

Measure	Tool vs. A1	Tool vs. A2
Recall	68.70	56.20
Precision	85.42	56.40
Fallout	0.42	2.51
Accuracy	98.52	96.04
Error	1.48	3.96
F-measure	75.37	54.51

Table 2. Comparing system results vs. human annotators

Measure	A2 vs. A1	Tool vs. A1	A1 vs. A2	Tool vs. A2
Recall	61.75	68.70	76.47	56.20
Precision	76.47	85.42	61.75	56.40
Fallout	1.00	0.42	2.50	2.51
Accuracy	96.70	98.52	96.70	96.04
Error	3.30	1.48	3.30	3.96
F-measure	66.79	75.37	66.79	54.51

As shown in Table 2, for the given annotation schema the task turned out to be difficult both for the system and for the humans due to the vague definitions of the semantic categories. For example, text about local food may be associated with either or both of the *Local Products* category and the *Food and Refreshment* category, depending on the context. Explicit resolution of such ambiguities in the expert definition would improve the results. Interpreting the results of this case study, we must take into account also that the diversity in accuracy metrics is partially caused by the different experience of the annotators in the tourism area. If we compare the difference in scores of F-measure, as the most aggregate characteristic, the overall difference in performances of the system and the humans is approximately 10%.

In the second stage of evaluation, the human annotators were observed to use 72% less time to correct automatically annotated text than they spent on their original unassisted annotations.

In the third stage, when the human annotators corrected automatically marked up documents, the results of comparison to the final human markup are given in Tables 3a, b, c and calibration to human performance in Table 4.

Table 3a. Evaluating system annotation vs. human Annotator 1 as assisted by the tool

Topic / Measure	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	96.88	94.32	97.34	96.91	96.68	94.96	90.24	83.33	93.36
Precision	100.00	93.26	98.50	100.00	83.21	99.12	100.00	100.00	96.10
Fallout	0.00	0.16	0.14	0.00	1.28	0.03	0.00	0.00	0.22
Accuracy	99.85	99.72	99.64	99.74	98.59	99.82	99.80	99.97	99.44
Error	0.15	0.28	0.36	0.26	1.41	0.18	0.20	0.03	0.56
F-measure	98.41	93.79	97.92	98.43	89.44	97.00	94.87	90.91	94.71

Table 3b. Evaluating system annotation vs. human Annotators2 as assisted by the tool

Topic / Measure	Geo-graphy	Local Prod-ucts	Cult-ure	Artistic Heritage	Sport	Accom-moda-tion	Food & Refresh-ment	Well-ness	Ser-vice
Recall	100.00	97.73	99.11	100.00	99.17	99.16	100.00	66.67	98.10
Precision	94.58	97.73	90.79	73.14	84.45	72.39	89.13	80.00	92.41
Fallout	0.30	0.05	0.95	3.31	1.20	1.19	0.26	0.03	0.46
Accuracy	99.72	99.90	99.05	96.96	98.82	98.82	99.74	99.92	99.46
Error	0.28	0.10	0.95	3.04	1.18	1.18	0.26	0.08	0.54
F-measure	97.22	97.73	94.77	84.49	91.22	83.69	94.25	72.73	95.17

Table 3c. Evaluating system annotation vs. humans – average scores

Measure	Tool vs. A1	Tool vs. A2
Recall	93.78	95.55
Precision	96.69	86.07
Fallout	0.20	0.86
Accuracy	99.62	99.16
Error	0.38	0.84
F-measure	95.05	90.14

Table 4. Comparing system results vs. humans assisted by the tool

Measure	A2 vs. A1	Tool vs. A1	A1 vs. A2	Tool vs. A2
Recall	80.99	93.78	92.54	95.55
Precision	92.54	96.69	80.99	86.07
Fallout	0.19	0.20	1.00	0.86
Accuracy	98.88	99.62	98.88	99.16
Error	1.12	0.38	1.12	0.84
F-measure	86.02	95.05	86.02	90.14

In contrast to the first experiment, this second case study was much more difficult to set up and evaluate than the first for the following reasons:

– Ambiguity in annotations: the large conceptual model of the domain is more difficult for usage as it allows ambiguities in interpretation.
– Difficulty in identifying fragments to be annotated: web documents contain various text structures such as tables, menu labels, free text and others.

– Size of the documents: in contrast to ads, which contained only a few sentences, the Web sites were of about 300 kbyte of text in HTML markup for each site.

However, in conclusion of this experiment we can say that our semantic annotation framework was able to demonstrate reasonable quality of results on the more general documents and the richer domain while maintaining fast performance.

5 Related Work

A number of tools have been shown to do well for various kinds of assisted or semi-automated semantic annotation of web content.

Text mining approaches usually use text itself as the basis for an analysis. For example, in [8] linguistic patterns and statistical methods are applied to discover a set of relevant terms for a document. Some tools combine data mining techniques with information extraction techniques and wrappers, as DiscoTEX [9].

SemTag [10] is an application that performs automated semantic tagging of large corpora. It tags large numbers of pages with terms from an ontology, using corpus statistics to improve the quality of tags. SemTag detects the occurrence of the entities in web pages and disambiguates them.

The KIM platform [11] is an application for automatic ontology-based named entities annotation, indexing and retrieval. In KIM, as well as in SemTag, semantic annotation is considered as the process of assigning to the entities in the test links to their semantic descriptions, provided by ontology. KIM performs recognition of named entities with respect to the ontology and is based on GATE[7].

Another tool that has been used on a large-scale is SCORE [12], which integrates several information extraction methods, including probabilistic, learning, and knowledge-based techniques, then combines the results from the different classifiers.

Our approach fundamentally differs from all these tools: it uses a lightweight robust context-free parse in place of linguistic analysis; our method does not have the learning phase, instead it has to be tuned manually when being ported to a particular application, substituting or extending domain dependent components; and it does not necessarily require a knowledge base of known proper entities, rather it infers their existence from their structural and vocabulary context in the style of software analyzers. This advantage helps make our tool faster and less dependent on the additional knowledge sources.

Much of the work in the information extraction community is aimed at “rule learning”, automating the creation of extraction patterns from previously tagged or semi-structured documents [13] and unsupervised extraction [14]. While learning issues are not addressed by our work, the application of patterns to documents is in many ways similar to our method, in particular the ontology-based method of Embley et al. [15]. The major differences lie in the implementation – whereas Embley’s method relies primarily on regular expressions, our approach combines high-speed context-free robust parsing with simple word search.

Wrapper induction methods such as Stalker [16] and BWI [17] which try to infer patterns for marking the start and end points of fields to extract, also relate well to our

[7] General Architecture for Text Engineering: http://gate.ac.uk/

work. When the learning stage is over and these methods are applied, their effect is quite similar to our results, identifying complete phrases related to the target concepts. However, our results are achieved in a fundamentally different way – by predicting start and end points using phrase parsing in advance rather than phrase induction afterwards. The biggest advantage of wrappers is that they need small amount of training data, but on the other hand they strongly rely on contextual clues and document structure. In contrast, our method uses context-independent parsing and does not require any strict input format.

6 Conclusions and Future Work

We have presented and evaluated a tool-supported process for the semantic annotation of web documents. The evaluation of our proposal included two case studies and the experimental results suggest good performance on the part of the semantic annotation tool. More importantly perhaps, the results suggest that productivity of a human annotator can increase substantially if the annotator works with the output of our tool, rather than conduct the annotation task manually. Our experiments also suggest that the tool is scalable when used with larger document sets. Apart from the experimental evaluation, we also consider the evaluation scheme itself as a novel contribution in that it measures not only the quality of the annotation, but also productivity improvements for human annotators. Our evaluation framework also takes into account inter-annotator disagreements to appropriately interpret the scores of the tool (since the human's performance is the upper bound for the automatic tool).

Our future research plans include tackling the problem of automatically generating inputs to the annotation process, such as object grammars and category keywords. We also propose to conduct experiments adapting other techniques used in software analysis to improve the quality of annotations and to accommodate different annotation granularities.

References

1. Isakowitz, T., Bieber, M., Vitali, F.: Web Information Systems. *Communications of the ACM*, Vol. 41(1) 78–80, 1998
2. Cordy, J., Dean, T., Malton, A., Schneider, K.: Source transformation in software engineering using the TXL transformation system. *Information and Software Technology Journal*, Vol. 44 (2002) 827–837
3. Kiyavitskaya, N., Zeni, N., Cordy, J.R., Mich, L., Mylopoulos, J.: Applying Software Analysis Technology to Lightweight Semantic Markup of Document Text. In *Proc. of Int. Conf. on Advances in Pattern Recognition (ICAPR 2005)*, Bath, UK, 2005, 590–600
4. Cordy, J.: TXL – a language for programming language tools and applications. In *Proc. of the 4th Int. Workshop on Language Descriptions, Tools and Applications*, Electronic Notes in Theoretical Computer Science, Vol. 110 (2004) 3–31
5. Dean, T., Cordy, J., Schneider, K., Malton, A.: Experience using design recovery techniques to transform legacy systems. In *Proc. 17 Int. Conf. on Software Maintenance,*2001, 622–631

6. Cordy, J., Schneider, K., Dean, T., Malton, A.: HSML: Design-directed source code hotspots. In *Proc. of the 9th Int. Workshop on Program Comprehension,* 2001, 145–154
7. Yang, Y.: An evaluation of statistical approaches to text categorization. *Journal of Information Retrieval,* 1999, Vol. 1 (1/2) 67–88
8. R. Feldman, M. Fresko, H. Hirsh, Y. Aumann, O. Liphstat, Y. Schler, M. Rajman. Knowledge Management: A Text Mining Approach. In *Proc. of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM98)*, 29–30, 1998.
9. Nahm, U. Y.; Mooney, R. J.: Text Mining with Information Extraction. In *Proc. of the Spring Symposium on Mining Answers from Texts and Knowledge Bases*, Stanford/CA, 2002, 60–67.
10. Dill, S., Eiron, N., Gibson, D., Gruhl, D., Guha, R., Jhingran, A., Kanungo, T., McCurley, K.S., Rajagopalan, S., Tomkins, A., Tomlin, J.A., Zien, J.Y.: A Case for Automated Large-Scale Semantic Annotation. *Journal of Web Semantics*, 2003, Vol. 1(1) 115–132
11. Kiryakov, A., Popov, B., Terziev, I., Manov, D., Ognyanoff, D.: Semantic Annotation, Indexing, and Retrieval. *Journal of Web Semantics*, 2005, Vol. 2(1), 49–79
12. Sheth, A., Bertram, C., Avant, D., Hammond, B., Kochut, K., Warke, Y.: Managing Semantic Content for the Web. *IEEE Internet Computing*, 2002, Vol. 6(4) 80–87
13. Nobata, C., Sekine, S.: Towards automatic acquisition of patterns for information extraction. In *Proc. of Int. Conf. on Computer Processing of Oriental Languages*, 1999
14. Etzioni, O., Cafarella, M.J., Downey, D., Popescu, A.M., Shaked, T., Soderland, S., Weld, D.S., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. *Artificial Intelligence 165* (2005) 91–134
15. Wessman, A., Liddle, S.W., Embley, D.W.: A generalized framework for an ontology-based data-extraction system. In *Proc. of the 4th Int. Conf. on Information Systems Technology and its Applications* (2005) 239–253
16. Muslea, I., Minton, S., Knoblock, C.A.: Active learning with strong and weak views: A case study on wrapper induction. In *Proc. of the 18th Int. Joint Conf. on Artificial Intelligence (*2003) 415–420
17. Freitag, D., Kushmerick, N.: Boosted wrapper induction. In *Proc. of the 17th National Conf. on Artificial Intelligence* (2000) 577–583

Extended Ontological Model for Distance Learning Purpose

Emma Kushtina, Przemysław Różewski, and Oleg Zaikin

Faculty of Computer Science and Information Technology, Szczecin University of Technology, Zolnierska 49, 71-210, Szczecin, Poland
{ekushtina, prozewski, ozaikine}@wi.ps.pl

Abstract. In the article authors propose an extended ontological model for distance learning, concerning pedagogical and cognitive requirements of the teaching/learning process as well as environmental requirements represented by SCORM standard. The main characteristic of the dedicated ontological model is reusability, which manifests itself in the possibility of adapting the knowledge model to different contexts and for different users by simply enabling knowledge sharing and knowledge management. Additionally, the article contains case studies (SCORM's course) presenting the proposed model in action.

Keywords: ontological model, knowledge management, distance learning, SCORM, learning object.

1 Introduction

Distance learning (DL) is an important and vital research problem. The most important reason for this statement is the fact that knowledge possessed by a person is quickly becoming outdated due to rapid growing of the amount of information and knowledge that appears in the world that surrounds us. This calls for methods of fast knowledge acquisition, like DL systems, regardless of the time and space constraints [3]. Fast knowledge outdating leads to long-life learning, which usually is realized by the DL systems. Because of the technological progress some jobs change their definitions every few years and without continuous vocational development it would be impossible to work in these fields.

From the organization's point of view DL is a framework that allows introducing methods of knowledge management. DL systems like Learning Management System (LMS) and Learning Content Management System (LCMS) [12] enable knowledge management in corporations on the basis of methods of sharing and storing knowledge through learning systems, e.g. in the form of courses or tutorials. A special place in corporational systems is occupied by DL thematical knowledge repositories, e.g. best-practice bases.

From the knowledge management perspective, both at a university and in a corporation, one of the greatest challenges standing in front of DL is ensuring reusability of knowledge gathered in DL systems. The problem of knowledge reusability is difficult to solve because: (a) there exist different types of knowledge,

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 155 – 165, 2006.

(b) there exist different ultimate environments and modes of working with knowledge, (c) knowledge users have different cognitive characteristics.

In the article authors propose an ontological knowledge model. The model fulfils all three (a-c) requirements and allows creating a complex system of knowledge management in the framework of a DL system.

2 Problem Statement

The issue of ensuring knowledge reusability in DL comes down to the problem of building a proper knowledge model and developing interfaces (procedures) enabling management of the gathered knowledge. It is also necessary to meet the following criterions resulting from the specificity of DL:

- Possibility to manipulate knowledge on account of the goal of learning.
- Possibility to manipulate knowledge on account of the planned level of competency to be obtained by a student.
- Working in LMS/LCMS environment (conformity with existing standards).
- Possibility to create a description of a given domain at different levels of depth (different levels of knowledge repository accuracy).
- Open and flexible structure enabling modification of knowledge contained in the model.

The requirements introduced above are met by the concept of an ontological model. The ontological approach is a widely discussed research problem from the point of view of e-commerce [8], artificial intelligence [4] or ontological engineering [11]. In our discussion we will, however, focus on the DL problems, that is: *(1) how to create an environment for managing ontological model in DL which ensures reusability? (2) what structure should ontological model in DL have?*

3 Related Work

The issue of knowledge reusability is a research problem discussed in literature (general approach to reusability [25], knowledge reusability in medicine [26], chemistry [9], engineering [32]) and as a part of many projects (e.g. PROLEARN http://www.prolearn-project.org/).

Generally speaking, the problem of knowledge reusability in DL should be connected to the issue of knowledge management. This issue is also related to the problem of creating knowledge bases and repositories adjusted to DL where the main problem is dividing knowledge into proper knowledge objects (called Learning Objects [7]) and choosing a model of knowledge representation to be used for storing the given didactic content. There are also standards covering the issue, the most important of which is SCORM (Sharable Content Object Reference Model), which standardizes creation of LMS/LCMS systems meeting the following requirements: accessibility, interoperability, durability, reusability, adaptability, affordability [23].

Managing ontologies in DL faces many problems [11]. The fundamental ones are the difficulties connected to creating and manipulating an ontology. Researchers are continuously searching for an effective way of automatic or semi-automatic ontology merging [14]. The basic problem of ontology mismatches defined in [28] is a result of the difficulty of concept mapping [6,24]. A separate aspect that has to be considered is the issue of ontology and the Semantic Web for E-learning [22], what translates into creating a personalized working space on the basis of Semantic Web technology.

The concept of creating and managing an ontology in DL proposed in literature is limited to the level of a knowledge object [18,30] avoiding the problem of the ontology level – that is the precise description of a given domain.

4 Extended Ontological Model

When considering the possibility of creating an ontological model for DL that ensures reusability, one has to pay attention to the necessity of formal definition of appropriate operations and model (structure). In DL we deal with the situation when with the help of a structure we try to reflect domain knowledge together with the corresponding pedagogical and cognitive requirements.

When domain knowledge is the object of structuralization, concepts that can be sorted in different ways become the main components. The aim of the domain knowledge structure is ensuring conformity of the knowledge representation model with the student's cognitive profile and learning goals. This is exactly the kind of mission a teacher carries on during direct contact with a student. Another – just as important – task results from this: ensuring possibility for adjusting the established knowledge structure to the personal educational situation in the situation when teacher is not present.

The above discussion leads us to considering the following conditions influencing the extended ontological model in DL:

a. When creating ontological model it is vital to differentiate knowledge: the fundamental (theoretical) from the procedural (operational) one, which together make up domain knowledge. Fundamental knowledge represents conceptual thinking, in result of which new paradigms, problems, task assumptions and behavior rules can be formulated. Procedural knowledge is essential for developing and realizing algorithms, performing operations, manipulating equipment etc. Analysis of different situation a person encounters bases on simultaneous usage of both kinds of knowledge, in different proportions – depending on the level of problem's complexity and its origin.
b. Following David Ausubel, the author of the concept of meaningful learning [1,20], we can state that learning consists in assimilating new concepts to the cognitive structures (conceptual maps) already existing in the student's mind. Contrary to rote learning, learning through understanding (meaningful learning) uses discovery learning process [5], where all concepts attributes are individually identified by the student. In order for this to happen the following requirements must be met: (i) learning material has to be conceptually clear, language and examples should be adapted to knowledge already possessed by the student, (ii) student has to be able to recognize and understand the linking between new material and his/her knowledge, (iii) student has to make effort and choose meaningful learning by himself/herself.

c. *(C)* While designing a DL system based on knowledge it is especially necessary to pay attention to the relationship: information – knowledge – competency. Competences and qualifications are one of the basic mechanisms of assessing knowledge assimilated by the student [17]. Competencies are always connected to a certain person and represent his/her knowledge and ability to use this knowledge in certain domain. On the other hand, qualifications reflect the competences basing on some assessment system. Nowadays several systems are being created (e.g. Framework for Qualifications of the European Higher Education Area [2]) to enable assessing qualifications/competences at different educational levels. The qualification system defines a spectrum of qualifications, where the bottom qualification includes noticeably smaller amount competencies then the top one. Every qualification is related to a corresponding set of competences.
d. Developing a system which would ensure human-computer interaction requires a detailed analysis of the problem's cognitive aspects. When analyzing an ontological knowledge system being developed with special attention paid to DL it is important to consider among others the following: (i) the way different types of knowledge are stored in our mind, (ii) organization of the information – knowledge processing chain in the mind, (iii) human perceptual constraints, (iv) cognitive learning styles.

In the domain of knowledge management, ontology is defined as an intentional, formal specification of concepts in a given domain and relations between them [13]. In [19] an ontology definition most accurate for learning purposes occurs, which differentiates between ontology defining structure and thesaurus.

The suggested by the authors extending of the ontological model consists in adjusting the classically understood ontology to the presented above requirements (a-d) and also considering the demand for reusability. The aim of the authors is extending the ontological model for the benefit of learning goals. Basing on the present state of knowledge it is not yet possible to create a universal ontology, which could be used in different contexts. Therefore, it is necessary to create a methodology for dedicated model management. We assume that a typical scenario of working with the knowledge model in its first step has the teacher choose the scope of knowledge. The choice of knowledge scope can be narrowed to defining the set of concepts which will make up the basis for creating didactic material. In the second step, semantic depth – the scope of competences to be passed on to the student – is defined for each concept .

The ontology structure of the extended ontological model consists of the way of describing concepts structure and the accepted relations between concepts. The relations have been divided into taxonomic and non-taxonomic ones and also a set of axioms making up the platform of domain knowledge of the modeled domain. Thesaurus defines the vocabulary that can be used while defining concepts and relations and also references for concepts and relations. The ontological model (*OM*) can be presented as follows:

$$OM = \{S, T\},$$

where: S – ontology structure, T – domain description thesaurus,

$$S = \{S_c, R, T, A\},$$

where: S_c – concept structure, R – relation between concepts, T – taxonomy, A – set of domain axioms.

$$T = \{T_p, T_r, F_p, F_r\},$$

where: T_p – thesaurus for the set of concepts, T_r – thesaurus for the set of relations, F_p – references for concepts, F_r – references for relations.

When developing knowledge models presented to the student in DL systems, it is necessary to consider the goal of learning, aiming at required competences and student's basic knowledge. In such situation it is not enough anymore to use terms, precise representation of knowledge capacity and depth contained in each concept becomes necessary. The level of knowledge capacity corresponds to the subset of objects counted to the class which was nominated by the name of the concept. The level of knowledge depth corresponds to the set of characteristics' values of each object recognized as a member of a certain class, with consideration of constraints forced by a specified goal of learning and level of required competences.

In literature we come across many definitions of concept. The main characteristic of concept is usually mental representation structure created through abstraction and generalization. [27,29] proposed an interpretation of concept as a nomination of a certain class of objects which have similar features. In our research the following definition of a concept will be used [15]: concept is a nomination of classes of objects, phenomena, abstract category, for each of them the common features are specified in such way that there is no difficulty with distinguishing every class.

The approach bases on using semantic operations PART_OF, IS_A, KIND -OF (see fig. 1). Aggregation (PART_OF), generalization (IS_A) and specialization (KIND_OF) are semantic operations, which can be considered as results of abstraction creation method. Concept can be seen as an abstraction [10], what helps to understand a complex object through decomposing it into less complicated components. Through PART_OF relation it is possible to describe the set of characteristics sufficient to recognize a certain abstract object as a member of the considered class. IS_A states that a specific object with the given values has been counted as a member of the same class. KIND_OF means that specific objects listed by name have been counted into the considered class.

Developing a concept matrix structure (fig. 1) bases on choosing and using existing definitions of concepts. For this, intentional and extensional approaches can be used. Each of the definitions is the basic point for specifying the characteristics of the object being defined, finding sets of defining characteristics and creating classifying schemas.

Formally the content, capacity and depth of knowledge can be described in the following way as a matrix concept structure:

$$G = \left\| g_{ij} \right\|, i = \overline{0, i^*}, \ j = \overline{0, j^*}, \text{ where:}$$

$$g_{ij} = \begin{cases} \hat{G}, i = 0, j = 0, \textit{concept's name} \\ O_i, i = \overline{1, i^*}, j = 0 - \text{names of objects belong to class } \hat{G}, \\ W_i, i = 0, j = \overline{1, j^*} - \text{names of common attributes belong to class } \hat{G}, \\ g_{ij}, i = \overline{1, i^*}, j = \overline{1, j^*} - \text{value of feature } j \text{ for object } i, \end{cases}$$

then: tuple $< \hat{G}, W_1, W_2, \ldots, W_{j^*} >$ describes the content of knowledge corresponding to the term of concept $\hat{G}$. Tuple $< \hat{G}, O_1, O_2, \ldots, O_{i^*} >$ describes the capacity of

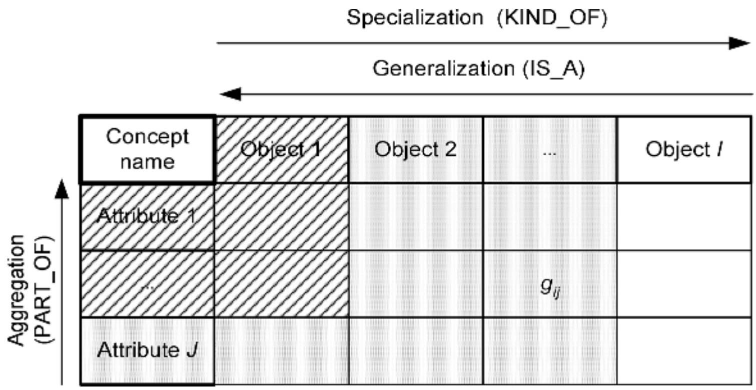

Fig. 1. Concept's matrix description structure

knowledge corresponding to the term of concept $\hat{G}$. The set of values of all features of all objects $\{g_{ij}\}$ is a description of the depth of knowledge corresponding to the term of concept $\hat{G}$. Adding new elements to set $I = 1, i*$ while maintaining the content of the concept means broadening the examined class of objects. Adding or removing elements from set $J = 1, j*$ means changing the content of the concept. Intersection $\delta = I \bigcap J \neq 0$ is the measure of acceptable tolerance for different forms of the concept, which correspond to the border of the domain being examined. If $\delta = I \bigcap J = 0$ we deal with a situation when the same word in different domains of knowledge refers to a different thing or phenomenon. The adequate extended ontological model creation algorithm has been discussed in detail in [21,31].

Developing ontological domain model in an educational situation requires analyzing a specific curriculum and learning goal, which play the role of constraints for capacity and depth of concepts used in didactic materials. In the context of the above presented definition of ontology, using concept's matrix structure for describing S_c leads to a two-level layout of the ontological domain model, where the first level is a network of concepts (similar to semantic network) and the second level defines the depth and capacity of knowledge included in each concept. Rules of creating the two-level arrangement can be used many times in reference to the same originally defined ontological model. This gives the possibility of developing a multi-level ontological model. Using the proposed approach enables adjusting the ontological domain model to specific educational goals. The ontological domain model extended in such way will allow for a significant level of automation of processing the model into a modular structure of didactic materials dedicated to being used in the learning process.

5 Case Study: SCORM's Course

Distance learning technologies allow for, among others, performing classes in an asynchronous and synchronous way, organizing repositories, creating knowledge sharing environment (e.g. virtual laboratory) and creating personalized didactic material.

In the following chapter authors present how the discussed ontological model works in DL learning environment conform to SCORM standard. In SCORM's course case the concept of adapting reusable knowledge to a specified student will be presented.

Authors chose the SCORM standard to test the proposed approach because it enables analysis of the proposed knowledge model in the context of real market solutions. SCORM is one of the most important DL standards, supported by most of the DL systems. Obtaining a guarantee that the given knowledge model works with SCORM ensures the possibility of implementing the proposed solution in LMS/LCMS systems. From the functionality point of view and alternative for SCORM could be IMS Learning Design Specification (IMS LD), describing the conceptual model and data structures, as well as the behavioral model and runtime behavior. However, authors decided to use SCORM due to its greater popularity and maturity. The presented discussions are easily transferable to IMS LD.

The new paradigm applied in e-learning systems described by the SCORM 2004 standard [23], assumes creating a knowledge repository containing a certain domain knowledge divided into knowledge objects (called Learning Object – LO). Didactic material meant for a certain student is built through creating a sequence of LO. SCORM standard covers following fields: directions for localization and creation of a structured education content (terminology and metadata model), methods of activating didactic materials in the e-learning environment, methodology of creating a sequence of LO in an interactive environment. Unfortunately, SCORM standard does not have any information about the knowledge model (its structure and creating methods). Let us discuss how the proposed extended ontological model works with SCORM standard. The discussion bases on [16], where the readers can find a more detailed explanation of the proposed algorithm named the didactic materials compilation algorithm.

The algorithm presents on the figure 2 adapts the extended ontological model to a certain student, with the consideration of current e-learning standards, as well as educational conditions of the process. The result of the algorithm is an educational sequence (LO sequence consistent with the SCORM standard) which, through telecommunication links, will be made available to the student.

The first step of the algorithm assumes ontological model dimension reduction on the basis of the learning objectives, which usually reduces the target level of knowledge from the given area. Basing on the reduced ontology and information about the level of the student's so-far-gained knowledge included in the student's profile basic concepts are selected and adapted to required competency profile.

Next, a clustered graph of concepts is created basing on the method presented in [21,31]. After that, clustered graph is being transformed into a tree, which corresponds to SCORM Activity Tree structure. The root of the tree is the final education goal – the basic concepts are placed at one of the end-levels of the hierarchical network.

Each of the trees is analyzed from the coverage algorithm point of view in order to form a "knowledge portion". The individual knowledge portion is supposed to be a teaching/learning process element and in the form of LO will be presented to the student as an element of the learning process. Transforming the "knowledge portion"

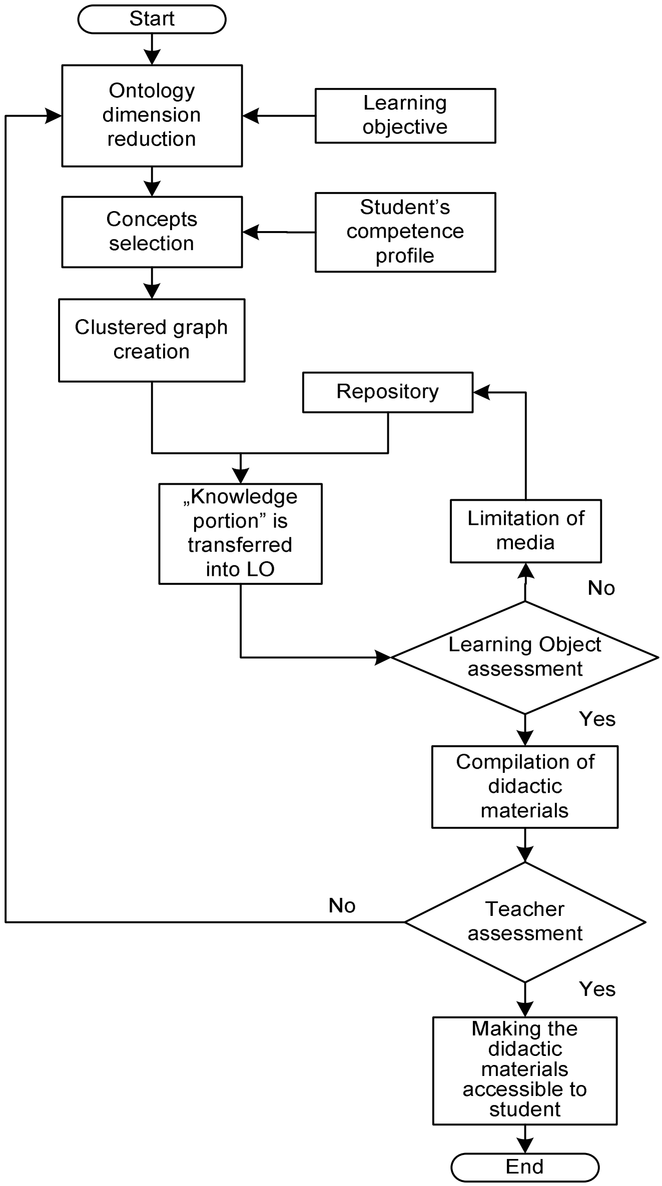

Fig. 2. Didactic materials compilation algorithm

into the form of a Learning Object requires applying the ontological model onto the repository in order to set up adequate computer metaphors. This action is connected to knowledge portion transformation into an information structure – Learning Object, which is assessed in the aspect of technological constraints of the process caused by the telecommunication network and the digital character of the content being integrated.

As we already mentioned above, LO accumulate similar concepts into one object. The question is – how many concepts should be incorporated into one Learning Object? The answer comes from the cognitive science. Research on the mind provides a useful memory model. The idea is to relate the “size” of the learning object to the short-term memory capacity. This issue is discussed in detail in [21,31].

The last group of activities, just before the final teacher’s control, is determining the sequence of passing through the scaled up graph. Cognitive science assumes necessity of connecting the knowledge being assimilated with the already owned one. SCORM gives possibility to define the LO sequence by dedicated script language. Next, all the elements are compiled into a form consistent with SCORM. The created product is finally analyzed by the teacher, who checks the efficacy of the didactic material by referring to his educational experience in teaching the given subject. After being accepted, the course is made available to the student through LMS/LCMS mechanisms.

The aim of the discussed case study is to present how a domain representation prepared on the basis of the proposed ontological model can be reused in relation to different students. Parameters we can manipulate here are: ontology dimension – influenced by the learning goal, concept selection – decided by the expected student’s competence profile and the size of LO, determined on the basis of student’s cognitive constraints analysis. The presented concept shows how in an uncomplicated way one can incorporate the learning goal, cognitive characteristics and the learning model into one, uniform system working in accordance with SCORM standard.

6 Summary

The level of ontological model's reusability allow for multiple usage in different didactic contexts. This means that it is possible to adapt the model to pedagogical requirements e.g. different didactic goals, adapt to different users through reproducing their cognitive characteristics in the model and the possibility to adapt to the amount of knowledge they possess. It is also possible to adapt to technical constraints, e.g. internet connection or safety policy, through a proper choice of computer methaphors. At the same time, the proposed formalization methods allow for a representation in the form of a knowledge model of different domains, basing on the generic idea of a semantic network.

The presented research shows that the concept of developing a general knowledge model in the form of an ontology which could be used in different contexts, for different kinds of knowledge and for different users is hard to do. One possibility of handling this dilemma is developing effective mechanisms of adapting the knowledge model to the requirements. The proposed approach, conform to the middleware concept, aims at concerning factors characteristic for a given domain. The presented ontological model and its management mechanism are placed between the knowledge repository and DL system. Their main task is to transform digital computer metaphors into education sequences sent to the student.

References

1. Ausubel, D.P., Novak, J.D., Hanesian, H.: Educational Psychology: A Cognitive View, 2nd edition, Rinehart and Winston, New York (1978).
2. Bologna Working Group on Qualifications Frameworks, A Framework for Qualifications of the European Higher Education Area, report, (Downloadable from website http://www.bologna-bergen2005.no/) (2005).
3. Burgess, J.R.D., Russell, J.E.A.: The effectiveness of distance learning initiatives in organizations. Journal of Vocational Behavior, 63(2)(2003) 289-303.
4. Chandrasekaran, B., Josephson, J., Benjamins, V.: What Are Ontologies, and Why Do We Need Them?. IEEE Intelligent Systems, 14(1)(1999) 20-26.
5. de Jong, T.: Scientific Discovery Learning with Computer Simulations of Conceptual Domains, Review of Educational Research, 1998 (68) 179-201.
6. Doerr, M.: Semantic Problems of Thesaurus Mapping, Journal of Digital Information, 1(8)(2001) Article No. 52.
7. Downes, S.: Learning objects: Resource for Distance Education Worldwide. International Review of Research in Open and Distance Learning 2(1)(2001).
8. Fensel, D.: Ontologies: Silver Bullet for knowledge Management and Electronic Commerce. Springer-Varleg, Berlin (2001).
9. Fernandez, M., Gómez-Perez, A., Pazos, J., Pazos, A.: Building a chemical ontology using methontology and the ontology design environment. IEEE Intelligent Systems and their Applications 14(1)(1999) 37-45.
10. Goldstein, R.C., Storey, V.C.: Data abstractions: Why and how?. Data & Knowledge Engineering, 29(3)(1999) 293-311.
11. Gomez-Perez, A., Corcho, O., Fernandez-Lopez, M.: Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web. Springer-Verlag, London (2004).
12. Greenberg, L.: LMS and LSMS: What's the Difference?. Learning Circuits - ASTD's Online Magazine All About E-Learning, 2002, (Downloadable from website http://www.learningcircuits.com/ 2002/dec2002/Greenberg.htm) (2002).
13. Gruber, T.R.: A translation approach to portable ontologies. Knowledge Acquisition, 5(2)(1993) 199-220.
14. Hameed, A., Sleeman, D., Preece, A.: Detecting Mismatches Among Experts' Ontologies Acquired through Knowledge Elicitation. In: Research and Development in Intelligent Systems XVIII, Springer-Verlag (2001) 9-22.
15. Kushtina, E., Zaikine, O., Różewski, P., Tadeusiewicz, T., Conceptual model of theoretical knowledge representation for distance learning, In: proceedings of the 9th Conference of European University Information Systems (EUNIS'03), Amsterdam, (2003) 239-243.
16. Kushtina E., Zaikin O., Różewski P.: On the knowledge repository design and management in E-Learning. In: Lu, Jie; Ruan, Da; Zhang, Guangquan (Eds.), E-Service Intelligence: Methodologies, Technologies and applications, Studies in Computational Intelligence , Vol. 37, Springer-Verlag (2006) 497-517.
17. Lahti, R.K.: Identifying and integrating individual level and organizational level core competencies. Journal of Business and Psychology, 14(1)(1999) 59–75.
18. Lin, Y.T., Tseng, S.S., Tsai, Chi-Feng: Design and implementation of new object-oriented rule base management system. Expert Systems with Applications, 25(3)(2003) 279-481.
19. Maedche, A., Staab, S.: Discovering Conceptual Relations from Text, In: Technical Report 399, Institute AIFB, Institute AIFB, Karlsruhe University, (2000) 321-325.

20. Novak, J.D.: Learning, Creating, and Using Knowledge: Concept Maps As Facilitative Tools in Schools and Corporations. Lawrence Erlbaum Associates (1998).
21. Różewski, P.: Method of information system design for knowledge representation and transmission in distance learning, unpublished doctoral thesis. Szczecin University of Technology, Faculty of Computer Science and Infor-mation Systems, (in Polish), (2004).
22. Sampson, D.G., Lytras, M.D., Wagner, G., Diaz, P.: (ed.), Special Issue on "Ontologies and the Semantic Web for E-learning", Journal of Educational Technology & Society, 7(4)(2004).
23. SCORM - Sharable Content Object Reference Model. Advanced Distributed Learning Initiative, (Downloadable from website http://www.adlnet.org).
24. Sun, Y: Methods for automated concept mapping between medical databases, Journal of Biomedical Informatics, 37(2004) 162-178.
25. Taboada, M., Des, J., Mira, J., Marín, R.: Development of diagnosis systems in medicine with reusable knowledge components. IEEE Intelligent Systems 16(6)(2001) 68-73.
26. Taboada, M., Martínez, D. and Mira, J.: Experiences in reusing knowledge sources using Protégé and Prompt. An application in a medical domain. International Journal of Human-Computer Studies, 62 (2005) 597-618.
27. Tsichritzis, D.C., Lochovsky, F.H.: Data models, Prentice Hall (1982).
28. Visser, P.R.S., Jones, D.M., Bench-Capon, T.J.M., Shave, M.J.R.: An Analysis of Ontology Mismatches; Heterogeneity versus Interoperability. In; Working notes of the Spring Symposium on Ontological Engineering (AAAI'97), Stanford University, (1997) 164-172.
29. Wong, H.K.T., Mylopoulos, J.: Two Views of Data Semantics A Survey of Data Models in Artificial Intelligence and Database Management. INFOR, 15(3)(1977) 344-382.
30. Wu, C.-H.: Building knowledge structures for online instructional/learning systems via knowledge elements interrelations. Expert Systems with Applications, 26(4)(2004) 311-319.
31. Zaikin, O., Kushtina, E., Różewski, P.: Model and algorithm of the conceptual scheme formation for knowledge domain in distance learning. European Journal of Operational Research 175(3)(2006) 1379-1399.
32. Ziga, T: Construction informatics: Definition and ontology. Advanced Engineering Informatics 20(2006) 187-199.

A Peer-to-Peer Virtual Office for Organizational Knowledge Management

Enrico Le Coche [1], Carlo Mastroianni[3], Giuseppe Pirrò[1]
Massimo Ruffolo[2,3], and Domenico Talia[1,2]

[1] DEIS, Università della Calabria, Rende, Italy
{gpirro, talia}@deis.unical.it, lecoche@hotmail.com
[2] EXEURA S.r.l., Rende, Italy
[3] Istituto di Calcolo e Reti ad Prestazioni (ICAR-CNR)
{mastroianni, ruffolo}@icar.cnr.it

Abstract. Peer-to-peer (P2P) systems enable users to build individual and cooperating autonomous communities. Recently the peer-to-peer paradigm has been proposed as a technological support for Distributed Knowledge Management (DKM). DKM solutions allow an easy sharing of knowledge created by Individual Knowledge Workers (IKWs) inside communities. In such dynamic environments peers can frequently join or leave the network and update their personal knowledge bases. In today's ubiquitous information society where users need to continue to work at any time from everywhere it is becoming very important to support IKWs with semantic "always on" virtual workspaces providing multi-task and knowledge handling capabilities. This paper presents a semantic virtual office model and its JXTA implementation in the K-link+ system. K-link+ is an ontology based P2P system for cooperative work and knowledge management that provide users with virtual office capabilities.

Keywords: Organizational knowledge management, distributed knowledge management, peer- to-peer networks, ontology, virtual office.

1 Introduction

Collaborative P2P applications for knowledge management are increasingly becoming popular. P2P systems permit an easy and quick creation of dynamic and collaborative groups even composed of people from different organizations. Those groups access and share knowledge interacting in both synchronous and asynchronous way. Communities of practice (CoPs) can be formed according to this approach. A CoP can be viewed as a group of people who share similar goals and interests [8]. They employ common practices, work with the same tools and express themselves in a common language. Therefore, a CoP can be viewed as a virtual place where individuals can produce and learn new concepts and processes from the community, allowing the same community to innovate and create new knowledge. "Community is the social dimension of the practice, the ideal context for creating knowledge. It is a virtual space where the individuals learn, by sharing personal experiences and social competences" [3].

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 166 – 177, 2006.

In this way, organization can become a community of communities, offering space for creating autonomous sub-communities. To support these new requirements, distributed knowledge management (DKM) has been proposed as a new vision of knowledge management. This paradigm is based on the principle that different perspectives within complex organizations should not be viewed as an obstacle to knowledge exploitation, but rather as an opportunity that can foster innovation and creativity. The two core DKM principles are:

- *autonomy* - communities of knowledge should be granted the highest possible degree of semantic autonomy to manage their local knowledge, and
- *coordination* - the collaboration between autonomous entities must be achieved through a process of semantic coordination, rather than through a process of semantic homogenization [1].

P2P solutions seem to naturally fit the DKM requirements. A technological DKM solution allows assembling and management of technological components can be assembled and managed together. DKM solutions support easy creation and sharing of knowledge available within virtual groups through collaboration. By collaboration we mean the way through which a group of people work together beyond the walls of a company or a department.

In today‘s ubiquitous information society more and more people work outside of the traditional office for many hours of the day. Current technologies do not properly support this new style of working and every day it is becoming harder and harder to exchange information in a labyrinth of network connections, firewalls, file systems, tools, applications, databases, voicemails, and emails. Individual Knowledge Workers (IKWs) spend most of their time in finding and exchanging information or reaching people, and very little is left to actually do productive work. In order to go beyond these issues, many companies use strategies that include portals, extranets, VPNs, and browser-based application strategies that at best have been only partially successful.

We argue that IKWs need a virtual workplace where the physical office can be recreated and where everybody and everything is easily available from anywhere at anytime. The virtual office approach can solve most of the aforementioned issues. Virtual offices can be viewed as work environments defined regardless of the geographic locality of employees. A virtual office fulfils the roles of the traditional, centralized office although the employees collaborate for the most part electronically with sporadic physical contacts. This model is becoming more and more central since, even in conventional offices, today many business relationships are necessarily maintained across distributed environments, for instance, customers and suppliers are located at different sites, project co-workers are often located in different department, and a CEO's speech may be listened remotely [7].

We developed a P2P system, named *K-link+*, that implements the virtual office model and allows users to create flexible and collaborative P2P applications for knowledge management. K-link+ aims at allowing users to integrate different applications (knowledge sharing, messaging, shared boards, agenda, etc.) within the same environment and to enrich the system with new tools to be added as new components.

The basic concept under the K-link+ virtual office approach is the *workspace*. A K-link+ workspace can be viewed as a work area integrating people, tools and resources each of them enriched with a semantic meaning given by ontologies. Ontologies have been shown to be the best way to model the semantic of information and to give a shared interpretation of a particular knowledge domain. Semantics in K-link+ are handled by an ontology framework [6] developed through the OntoDLP language [5].

The paper is organized as follows. Section 2 describes the ontology framework supporting the system and the K-link+ approach to virtual office. Section 3 discusses the K-link+ JXTA implementation. Section 4 describes the implemented system. Section 5 discusses related work and section 6 draw some conclusions.

2 The K-Link+ Semantic Support

Remote collaboration and knowledge handling are the main features of the virtual office paradigm and DKM solutions seem to naturally fit its requirements. Most of the current approaches to collaborative work - while being worthwhile - do not take into account semantic aspects of knowledge management. A complete and useful collaborative system needs to be able to deal with both information and knowledge on a semantic basis. Semantic has been recognized as the main challenge in the next generation of both standalone and web applications. Giving a semantic meaning to information and activities enables users to efficiently find useful knowledge (people, activities, knowledge object, etc.). In the latest years the knowledge management community is considering ontologies as an adequate semantic support. Ontologies are abstract models of knowledge domains that support their modeling in terms of concepts, relationships between concepts, class hierarchies and properties. Ontologies also offer a way for defining a set of possible instances, thus providing links between the model and the modelled reality and permit reasoning about the knowledge contained in their domain models.

For coping with the semantic aspects of information in the K-link+ system we developed an ontology framework. The framework is organized in two layers.

The first layer contains the *Upper Ontology (UO)* and a set of *Core Organizational Knowledge Entities (COKE)* ontologies. The UO contains the concepts of a particular knowledge domain relevant for an organization. The UO is used as a shared basic set of concepts and relationships that enable semantic collaboration among the IKWs belonging to an organization. It is worth noting that the UO can be dynamically specialized and enriched with new concepts. The enrichment and specialization process starts from the initial UO that constitutes a common abstract view of modeled reality. The UO enrichment process allows developing the organizational knowledge background in an emerging way. The COKE ontologies aim at giving a semantic definition of some organizational sources of knowledge. We identified five main types of COKE that describe:

- *Human Resources:* in terms of single persons and organizational groups (Groups, Community of Practices, Project Teams). For each person, personal data, skills, organizational areas and groups memberships, participation to business processes and concepts of interest are represented.

- *Knowledge Objects:* (Textual Documents, Database elements like tables and/or table fields) in term of their metadata (data of creation, document type, etc.) and main concepts,
- *Technological resources:* in terms of tools by which knowledge objects are created, acquired, stored and retrieved.
- *Business Processes:* in terms of sub-processes, activities, transitions, transition states and conditions, transition patterns and concepts characterizing the specific activities and process instances.
- *Services:* in terms of provided knowledge objects, processes in which they are involved and people allowed to access them. This ontology for example can constitute a semantic description of the available web services.

The relationships existing between the COKE ontologies and the UO are *annotation* relationships. With annotation we mean that each COKE ontology instance is semantically annotated at least at one concept of the UO. For instance, let us consider a human resource expert in object-oriented programming. The annotation relationship states that the human resource should have a semantic annotation to the Java programming language or similar concepts contained in the UO. In analogous way the other organizational source of knowledge represented in our ontology framework can be semantically annotated. This way the K-link+ system can exploit the semantic meaning given to each organizational asset for searching and reusing knowledge on a semantic basis. The execution of a query to the UO can be performed using a specific tool able to retrieve all the elements related with a specific concept. Elements can be filtered to obtain specific COKE elements related to the query. For example a query result can contain people knowing a given concept or systems containing knowledge objects related to a concept. This allows the management of implicit and explicit knowledge stored in structured, semi-structured or unstructured machine-readable forms. Figure 1 shows in detail the ontology framework structure and the relationships existing between the different types of ontologies.

The second layer of the ontology framework contains a set of UO extensions called Workspace Ontologies and a Personal Ontology. A Workspace Ontology (WO) built from one or more UO concepts allows workspace members to specialize UO concepts for coping with the reality of interest of a workspace. Thanks to this task-based ontology, users do not have to deal with complex ontology representations that can push him/her to desert the ontology use. Initially the workspace members share the same set of concepts. However the WO can be extended in a distributed way (e.g., through a specific software component that handles a distributed voting mechanism involving all the participants to a workspace) for facing emerging aspects. Moreover, during the WO lifetime a concept can also be promoted to become part of the UO and then to belong to the entire organizational background. A Personal Ontology (PO) represents a IKWs personal network of concepts, obtained as a specialization of one or more UO concepts and used in all Personal Knowledge Management operations. The relationships existing between the UO and the Workspace and Personal Ontologies are *specialization* relationships since such ontologies are specializations of one or more UO concepts.

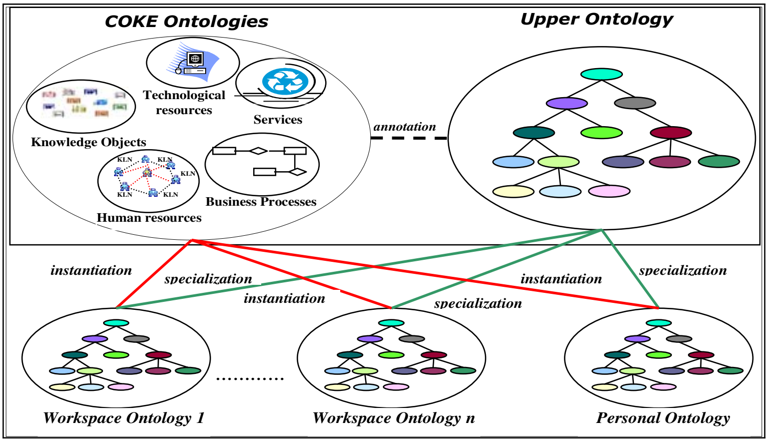

Fig. 1. K-link+ ontology support

2.1 The K-Link+ Approach to Virtual Office

The K-link+ VO environment is based on the workspace concept. A workspace (WS) can be viewed as a common semantic work area accessible at any time from everywhere and composed by COKE instances (e.g human resources, knowledge objects, processes and activities) and tools. The K-link+ system is designed for allowing users to work collaboratively in this common environment. Each *K-link node* (KLN) can be member of different workspaces. For each workspace two different roles are defined:

- *Manager:* A workspace administrator with full capabilities.
- *Participant:* A workspace participant with reduced but extensible (under Manager control) capabilities.

A workspace setting-up is necessary when a new organizational task must be carried out. For example, let us consider a process-based organization that must define a project proposal to apply for an UE FP7 call in the intelligent content creation and management objective. For dealing with this task the organizational project leader can set up a proper workspace delegated to fulfil the commitment requirements. Then by using the K-link+ functionalities, the project leader:

- Chooses the proper execution process related to a FP7 proposal definition from the process ontology that is handled by the K-link+ ontology editor/browser.
- Chooses the existing literature and document templates concerning the project topic. In this case it is valuable to populate the workspace document base with knowledge objects related to the project topic. Interesting knowledge objects can be searched by the K-link+ file sharing service able to handle keyword and semantic based searches.
- Defines an appropriate team of IKWs whose skills can be exploited to solve the problem as fast as possible. In the above mentioned example the K-link+ system should be able to find through the ontology support at least the following IKW

profiles: a number of experts in financial planning and a number of experts in knowledge management. People having the selected profiles become members of the workspace after receiving inviting messages sent by the workspace manager through the K-link+ inviting service.
- Assigns project activities to the IKWs by sending proper messages containing information about activities to perform.
- Chooses a set of services enabling an automatic proposal submission to the UE FP7 programme. The services ontology should contain a reference to the web services available on the UE FP7 web site. Services can be embedded directly in the K-link+ workspace perspective providing IKW with a common work environment gathering different kinds of applications.

Once the project leader (that becomes the workspace manager) has done the above mentioned activities and sent inviting messages to the chosen IKWs, the invited IKWs become part of the workspace and start cooperating to carry out the task.

The second important aspect for completing a workspace setting up is the choice of a set of tools through which the workspace members can perform the effective work. The workspace manager or its delegates can choose the required tools among a basic set of tools with which the K-link+ system is endowed. Moreover, it is also possible to develop specific tools that can be plugged into the system as libraries. The tool adding process is entirely handled at run-time. When a new tool is added to a workspace all the workspace members will automatically be informed and a local tool instance will be created. Hence, each tool updating, in terms of data, will be forwarded to all workspace members. This way the distributed IKWs can work as they were in the same physical place.

A K-link+ workspace is semantically supported by the UO. A workspace creation is followed by a parallel semantic definition obtained through a group ontology instance creation in the human resources ontology. This instance is semantically annotated at least to a concept of the UO. In the example above the created workspace must be semantically annotated to the concepts of “project proposal definition”, “knowledge management”, “financial planning” and so on. Thus, the workspace profile will include member profiles and the core ontology concepts used to semantically annotate the COKE instances (e.g the concepts to which knowledge objects, processes and other entities are annotated in the workspace). This way the K-link+ system performs reasoning operations. For example, the next time that the same organization must deal with a similar commitment, as the specification of a new *FP7* project, a search can be issued for a workspace that in its profile contains concepts like “project proposal definition”, “financial planning” and so on. This search will likely discover the FP7 project workspace. Thereafter the project leader can select the documents, templates and human resource profiles that can profitably be reused for the new FP7 project.

3 K-Link+ JXTA Implementation

In designing the K-link+ virtual office architecture we considered typical needs of an IT company. Our analysis identified “ubiquity” as the main requirement that a virtual office should fulfil. In fact, as discussed in the previous sections, it is becoming a common habit for IKWs to work outside of the "company walls".

The K-link+ system has been implemented using J2SE 5.0 and the JXTA library version 2.3.6 6. (ww.jxta.org).

The K-link+ architecture (figure 2) includes services built through the JXTA protocols. In the following we analyze the implemented architecture giving a concise description of each layer.

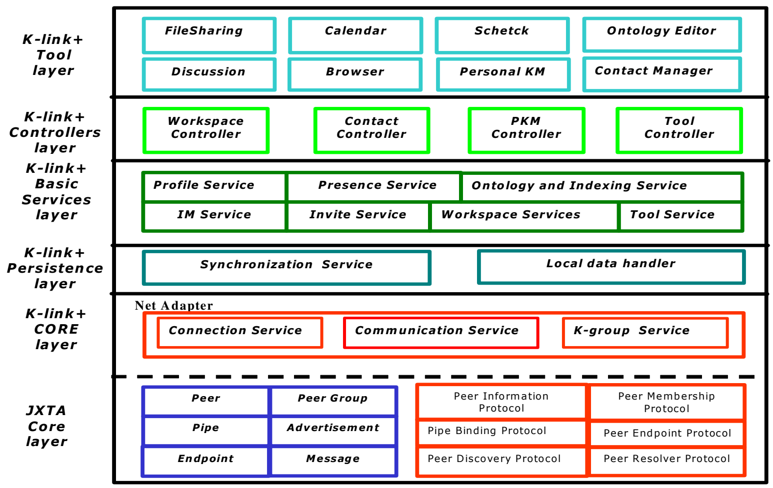

Fig. 2. The K-link+ implementation architecture

K-link+ Core layer

This layer implements the K-link+ basic services upon the JXTA ones adding some improvements for system purpose. The K-link+ system maps a K-link+ peer to a JXTA peer and a K-link+ group (K-group) to a JXTA peer group. The system is endowed with a basic super-group (*K-link+ super group*) which is included inside the worldwide JXTA Network. Each KLN is provided with information about this super group in order to speed up its convergence inside a K-link+ network. Thus, a KLN can be defined as a *peer inside the K-link+ super group* and a K-group can be defined as a *peer group inside the K-link+ super group*. Through the *communication service* all the communication operations within the K-link+ network are managed.

K-link+ Persistence layer

This service enables K-link+ to operate in a pure P2P and online/offline mode. Both in the online and offline modality the workspace data changes are caught by the *local data handler*. The local data handler is responsible for maintaining a set of repositories each of them delegated to maintain specific set of data. In the online modality the workspace data changes are both locally stored and sent to a *K-link+ superpeer* (synchronization service) which aim is to maintain a common data view for the workspace members. In the current implementation the K-link+ superpeer stores information in a MySQL relational database.

Conversely, offline changes will be temporary stored in a local buffer. When the user reconnects, through the *synchronization service* he/she empties its buffer updating the global workspace view by sending changes to the synchronization server. All the communications between a KLN and the synchronization server are checked by an acknowledgement message. Figure 3 shows more in detail the data interactions between the layer of the K-link+ architecture. In the figure, different kinds of user actions (contact management, profile management, etc.) are represented with different colors. Such colors help to identify the architectural components that are involved at the different layers for each kind of user action. It can be noted that the *service layer* components exchange data both with the local *repository layer*, through the *personal data handler*, and with the *synchronization server* through the *synchronization service*.

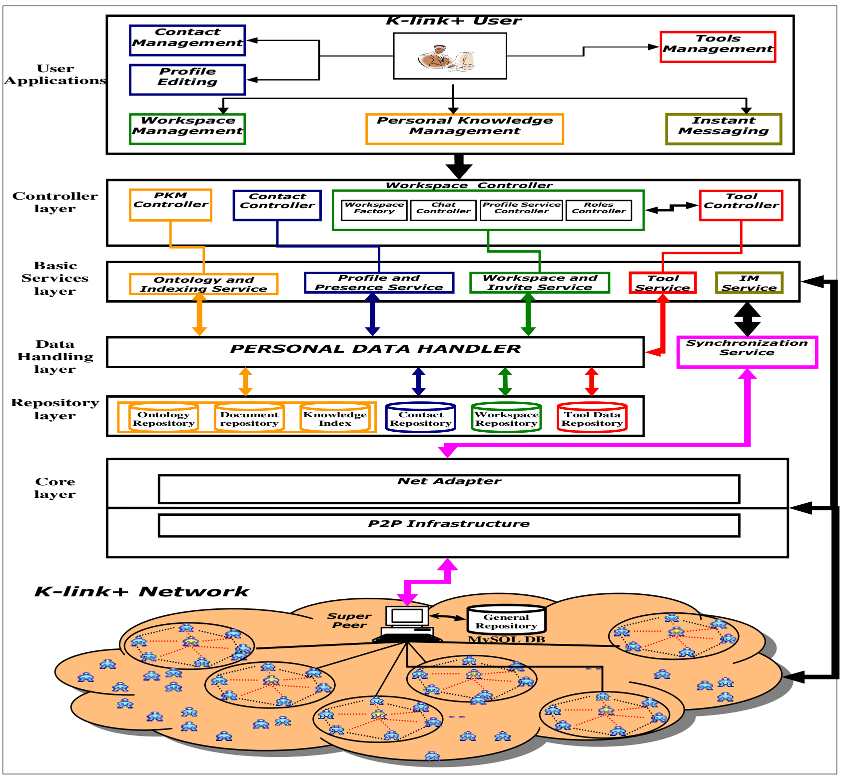

Fig. 3. The K-link+ abstract architecture

K-link+ Basic Services layer

K-link+ services contain a set of services responsible to interact wit the K-link+ main components. The *invite service* enable workspace managers to send invite message to the chosen workspace member in which a text message can be also specified. When an invitation event is recognized, the service shows the message in the GUI and the user can accept or decline the invitation. The *instant messaging service* allows KLNs to communicate each other via a chat-like system. The *profile service* allows users to publish their profile within the K-link+ network. The *ontology and indexing service* component enables users to build and manage personal and workspace ontologies through a graphical ontology editor. The *workspace service* manages workspace settings such as general info, sets of tools, member roles and role capabilities and responsibilities. Each workspace update is notified to the superpeer. If a peer has been offline for a stint, when it get reconnected this service queries the superpeer for synchronizing local workspaces settings also in term of data.

K-link+ Controllers layer
This layer contains a set of controllers through which the system interacts with the persistence layer. The *workspace controller* handles the workspace management operations such as workspace setting up and removing. The *contact controller* enables a KLN to discover other KLNs from the network and add some of them to its personal contact list. The *personal knowledge management controller* (PKMC) is delegated to manage operations on the KLN personal knowledge. A KLN can associate its documents to the personal ontology concepts giving them a semantic meaning. This component enables users to search inside their knowledge on a semantic basis. In fact, starting from a concept contained in the personal ontology, the system can return all the documents associated to that concept. Additionally the PKMC manages a Lucene (lucene.apache.org) handler delegated to index personal knowledge for keyword search. Finally the *tool controller* allow workspace managers to add and remove tools within a workspace.

K-link+ Tools layer
This layer enable workspace manager to chose the set of tools to include in a workspace. Basically K-link+ enables workspace members to choose among a set of tools (file sharing, shared calendar, contact manager, etc.). Moreover, other tools can be developed and included in the system as components. In fact, the development of a tool in the K-link+ system can be carried out by third parties with the only requirement that the K-link+ tool interface must be implemented. Here we present the K-link+ tool interface:

```
public interface KlinkTool {
 public void init(Workspace ws);
 public Container getToolGui();
 public ToolAdvertisement getToolAdv();
 public void close();
 public ImageIcon getIcon() }
```

The *init* method takes as parameter a workspace that represents the scope of the tool providing it with the necessary functionalities. Through this method the tool loader can create a new tool instance within the environment. The *getToolGui* method returns the GUI of the tool that will be added to the workspace GUI. The *getToolAdv* method returns a *Tool Advertisement*, that is a JXTA advertisement describing the tool (tool version, tool creator, etc.). The *close* method permits to delete the instance of a tool within a workspace and the *getToolIcon()* method returns an icon associated to the tool. Moreover, in order to add a new tool within the system, a user must provide each tool with an XML document (called plugin.xml) containing information about the tool main class. Here we can see a document describing the file sharing tool:

```
<?xml version="1.0" encoding="UTF-8"?>
<?k-link version="2.0"?>
<plugin>
<toolname>File sharing</toolname>
<mainclass>tool.filesharing.FileSharingEngine</mainclass>
<description>File sharing tool</description>
<creator>Giuseppe</creator>
</plugin>
```

4 Knocking on the K-Link+ Virtual Office Door

K-link+ has been implemented and it is under evaluation in a small software company for supporting project teams. In this section we present some interfaces of the implemented system.

Virtual Desk

The Virtual Desk (figure 4) represents the starting point for monitoring the K-link+ elements including workspaces, contacts, presence, and for executing basic functions such as creating new workspaces and communicating with or inviting users in workspaces.

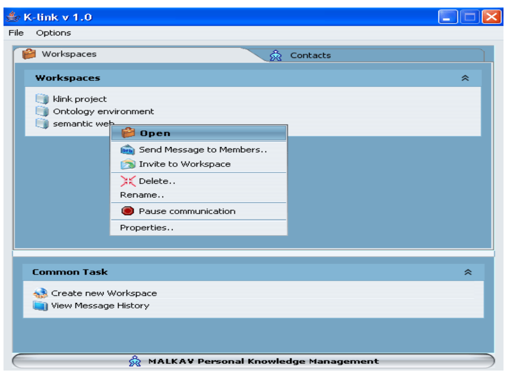

Fig. 4. K-link+ Virtual Desk

Workspace Perspective

Figure 5 shows the workspace perspective view that allow a user to choose the tool to deal with by clicking on the corresponding tab in the bottom of the workspace GUI. Moreover the GUI contains on its right side the contact view providing information about online and offline contacts. The workspace creation is very simple and can be done in few steps. First of all, a user insert the workspace name and description. The second step is the team building performed by inviting other users to take part of the new workspace. A KLN can send an invitation to all the users of its contact list. Finally the workspace manager chooses the set of tools to be included in the workspace.

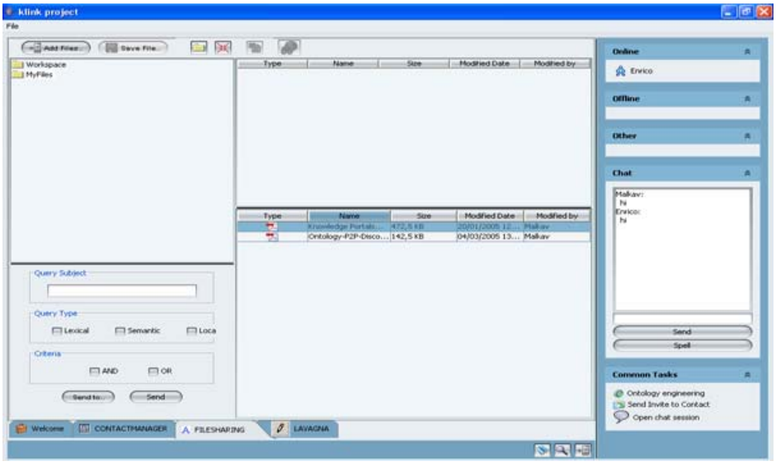

Fig. 5. K-link+ Workspace perspective

5 Related Work

Recently some systems and prototypes that share some features with K-link+ have been proposed. Ifreestyle (www.digitaldream.jp) is a group-based collaboration software, which manages individual information. As a Personal Information Management (PIM) software, ifreestyle provides features for schedules management, message exchange, address book and search operations by using simple and user-friendly user interfaces. As a team-collaboration software, ifreestyle provides features for creating a shared information space (group creation), group invitation, group membership, and iDESK. iDESK is a powerful group communication tool, consisting of answer board, chat tools, bulletin board, and management decision support tools.

Groove Virtual Office (http://www.groove.net) is an integrated environment for creating distributed virtual offices. Collaboration activities with Groove take place in a shared application space, which is accessed from a rich application client, called transceiver. A shared space, including tools and persistent data, is duplicated on every space member's computer. Data within a shared space is encrypted, both on disk and over the network, to assure confidentiality and integrity. Both data and commands are transformed, stored and transmitted as XML documents. Every modification made in a shared space is propagated to the other peers. Though its approach is very promising, differently from K-link+, Groove does not include ontologies nor provides components with any semantic meaning.

SWAP (Semantic Web and Peer to Peer) [4] is a research project started in 2002, aiming at combining ontologies and P2P for KM purposes. SWAP allows local KM by a component called LR (Local node repository), which gathers knowledge from several sources and represents it in RDF-Schema. Like K-link+, SWAP allows to search for knowledge; it uses a language called SeRQL an evolution of RQL.

KEEx [1] is a P2P architecture that aims to combine both semantic and P2P technologies. This system implemented in JXTA allows a set of K-nodes to exchange information on a semantic basis. Semantic in KEEx is achieved by the notion of context. A context in KEEx, represents a personal point of view about reality and it is represented using a proprietary language called CTXML [2]. KEEx lets the user completely free about context creation without providing them with any organizational background. This approach is innovative, due to the presence of an automatic mapping algorithm which aims to find correspondences between concepts present in different contexts.

6 Conclusions

The paper described K-link+, a semantic virtual office environment based on a peer-to-peer architecture. The K-link+ architecture and implementation have been outlined and its user interface and features have been presented. In concluding the paper we summarize the main properties of K-link+ that make it an interesting system for supporting distributed collaborative work according to the virtual office approach: it is endowed with ontology support for managing knowledge; it is platform independent because it has been implemented in Java; it is an environment extensible by developing custom ad-hoc tools; it supports both online and offline work via superpeer synchronization.

References

1. Bonifacio, M., Bouquet, P., Mameli, G., Nori, M.: KEEx: A Peer-to-Peer Solution for Distributed Knowledge Management. Proc. of the 4th International Conference on Practical Aspects of Knowledge Management PAKM, Vienna, Austria (2002)
2. Bouquet, P., Serafini, L., Zanobini, S.: Semantic coordination: A new approach and an application. In: Fensel, D., Sycara, K., Mylopoulos, J. (eds.): Proc of the 2nd ISWC Conference, Sanibel Island, Florida, USA, Lecture Notes in Computer Science, Springer-Verlag, Vol. 2870 (2003) 130–145
3. Cuel, R.: Trade off between organization and technology in the knowledge management. A case of study. Phd Thesis, University of Udine, Italy (2003)
4. Ehrig, M., Tempich, C., Broekstra, J., Van Harmelen, F., Sabou, M., Siebes, R., Staab, S., Stuckenschmidt, H.: SWAP: Ontology-based Knowledge Management with Peer-to-Peer Technology. Proc. of the 1st Workshop Ontologie-basiertes Wissensmanagement, Lucerne, Switzerland (2003)
5. Eiter, T., Leone, N., Mateis, C., Pfeifer, G., Scarcello, F.: A Deductive System for Non-Monotonic Reasoning. In: Dix, J. Furbach, U. Nerode, A. (eds.): Proc. of the 4th International Conference on Logic Programming and Non Monotonic Reasoning LPNMR, LNAI, Springer-Verlag, Vol. 1265 (1997) 364-375
6. Gualtieri, A., Ruffolo, M.: An Ontology-Based Framework for Representing Organizational Knowledge. Proc. of the I-Know - International Conference on Knowledge Management, Graz, Austria (2005)
7. Lave, S., Wenger, E.: Situated learning: Legitimate peripheral learning. Oxford University Press (1991)
8. Marshak, D. S.: Groove Virtual Office Enabling Our New Modes of Work. Report by Patricia Seybold Group, http://www.groove.net/pdf/backgrounder/GV0-Marshak.pdf (2004)

Mining and Supporting Task-Stage Knowledge: A Hierarchical Clustering Technique

Duen-Ren Liu[1], I-Chin Wu[2], and Wei-Hsiao Chen[1]

[1] Institute of Information Management, National Chiao Tung University, Taiwan
{dliu, noin}@iim.nctu.edu.tw
[2] Department of Information Management, Fu Jen Catholic University, Taiwan
icwu@im.fju.edu.tw

Abstract. In task-based business environments, organizations usually conduct knowledge-intensive tasks to achieve organizational goals; thus, knowledge management systems (KMSs) need to provide relevant information to fulfill the information needs of knowledge workers. Since knowledge workers usually accomplish a task in stages, their task-needs may be different at various stages of the task's execution. Thus, an important issue is how to extract knowledge from historical tasks and further support task-relevant knowledge according to the workers' task-needs at different task-stages. This work proposes a task-stage mining technique for discovering task-stage needs from historical (previously executed) tasks. The proposed method uses information retrieval techniques and a modified hierarchical agglomerative clustering algorithm to identify task-stage needs by analyzing codified knowledge (documents) accessed or generated during the task's performance. Task-stage profiles are generated to model workers' task-stage needs and used to deliver task-relevant knowledge at various task-stages. Finally, we conduct empirical evaluations to demonstrate that the proposed method provides a basis for effective knowledge support.

Keywords: knowledge-intensive task, task-relevant knowledge, task-stage mining, hierarchical agglomerative clustering.

1 Introduction

Reusing knowledge assets extracted from historical tasks is the key to providing effective knowledge support for workers during the performance of tasks. Among the different types of knowledge management, the repository of structured and explicit knowledge, especially in document form, is a codified strategy for managing knowledge [5],[16]. Thus, intellectual content is generally codified in an explicit form to facilitate knowledge reuse and sharing [3],[11].

As the operations and management activities of enterprises are mainly task-based, organizing and delivering relevant knowledge from the perspective of business tasks is regarded as a desirable and effective way to fully reuse organizational knowledge assets [1],[6]. The KMS developed by the *KnowMore* project is an example of proactive delivery of task-specific knowledge based on the contexts of tasks within a process [1]. Moreover, for knowledge-intensive tasks, such as research projects in

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 178 – 188, 2006.

academic organizations, project management in firms, and product development in R&D departments, it is more difficult to provide task-relevant knowledge during a task's execution. Generally, knowledge workers require a long time to accomplish knowledge-intensive tasks. As observations from pilot studies in information search process areas [9], [14] show, the information needs of knowledge workers vary according to the different stages of a task's performance. Vakkari [14] concentrated on the user's information seeking activities during task performance. The empirical studies demonstrate that users' information needs depend on the particular task stage, and the features of each stage provide clues about the worker's task needs.

This work focuses on providing knowledge support based on task-stages for knowledge-intensive tasks within organizations. In our previous study, we built a task-based knowledge support system without considering the information needs of knowledge workers at different stages of a task's performance [10]. The objective of this work is to support the provision of task-relevant knowledge according to workers' task-needs at different task-stages. To this end, we propose a task-stage mining technique for discovering task-stage needs from sets of historical tasks. The proposed method adopts information retrieval techniques and a modified *Hierarchical Agglomerative Clustering* (**HAC**) algorithm to identify task-stage needs by analyzing codified knowledge (documents) accessed or generated during the task's execution. *Hierarchical Agglomerative Clustering* belongs to a class of clustering approaches [7],[8] that produce clusters by merging based on the similarity of two existing clusters. In this work, we modify the traditional HAC algorithm to generate task-stages (clusters) by setting a time window to select candidates for merging. Once the stages of a task have been identified by the modified HAC algorithm, task-stage profiles are generated to model workers' task-stage needs and used to deliver task-relevant knowledge at various task stages. Finally, we conduct empirical evaluations to confirm the effectiveness of the proposed technique.

The remainder of the paper is organized as follows. Section 2 formulates the problem addressed in this work. Section 3 describes the proposed task-stage mining technique. The experimental evaluations are reported in Sections 4. Finally, we present our conclusions and discuss the direction of future work in Section 5.

2 Problem Formulation of Mining Task-Stage Needs

2.1 Overview of This Work

This work broadly defines a task in an organization as a unit of work, such as a project, research work, or activity. We use a profiling approach to model workers' task-stage needs, i.e., information needs (profiles). Task-stage profiles are generated to model worker's information needs at each task stage and to provide task-relevant knowledge accordingly. A task-stage profile, which specifies the key subjects of a task-stage and is represented by a feature vector of weighted terms, is used to retrieve relevant codified knowledge from a repository of previous tasks. The key contents of codified knowledge, namely document profiles, are also represented as a feature vector of weighted terms. Relevant documents can be retrieved to provide knowledge support for a task's execution according to the similarity measures (e.g., cosine measures) of task-stage profiles and document profiles.

A task class contains tasks with similar properties (similar tasks). The identification of task classes is based on historical tasks, and a task belonging to a particular class is called an instance of that class. A task class/instance is analogous to the concept of an object class/instance. Tasks in different task classes generally have different task-stage needs, while tasks belonging to the same task class may have similar task-stage needs. Accordingly, task-stage profiles generated by analyzing historical tasks in the same task class are used to provide relevant knowledge in support of the execution of the task at hand.

The term *virtual task* is used to represent a class of similar tasks. Each historical task instance is associated with documents accessed/generated during task performance. The problem of identifying the task-stage needs of a virtual task is described as follows. *"Given a virtual task and a set of task instances with associated documents, find the number of task stages and the corresponding profile of each task-stage."*

2.2 Term Definitions

Definition I - Task: A task is a fundamental unit in business.

- **T:** *Task set*; T = $\{t_1, t_2, \ldots, t_r, \ldots, t_n\}$, where a task is either an executing-task (on-going task) or an existing task in the task-based working environment.
- t_r**:** *Existing-task*; An existing-task is a historical task accomplished within the organization.
- t_v**:** *Virtual task;* A virtual task that represents a class of similar tasks.
- *Task instance*: A task instance of t_v is an existing-task belonging to t_v.
- T_v**:** *Set of task instances;* A set of task instances of the virtual task t_v.

Definition II - Task Document Sequence: A task document sequence is a sequence of documents retrieved during a task's performance. Documents are sorted according to their retrieval time. Each task instance has its own task document sequence.

- **TDS(**t_r**)** : *Task Document Sequence of* t_r; A sequence of documents accessed/generated while performing a task t_r. TDS(t_r)=<$d_1, d_2, \ldots, d_m$>.

Definition III - Task Stage: Task document sequences of task instances are clustered into stages based on the similarity measures and retrieval time of documents. The clustering result provides the stages of the task at hand.

- **TS(**t_r**):** *Task-stages of a task instance*; A task t_r comprises several task-stages. **TS**(t_r)=<$ts_r[1], ts_r[2], \ldots, ts_r[k]$>, where $ts_r[i]$ denotes the task-stage i of t_r.
- **TS(**t_v**):** *Task-stages of a virtual task*; The task-stages of a virtual task t_v, which are derived from the task stages of task instances in t_v. **TS**(t_v)=<$ts_v[1], ts_v[2], \ldots, ts_v[k]$ >

Definition IV - Task Stage Documents and Profiles: Each cluster of documents represents a task-stage with associated documents, called task-stage documents. Documents in task-stage k of task t_r are denoted by $ts_r[k]$.docs, while documents in task-stage k of virtual task t_v are denoted by $ts_v[k]$.docs. Each task-stage profile is derived from the feature vectors of documents in each task-stage. Let $ts_r[k]$.profile and $ts_v[k]$.profile denote the profile of task-stage k of task t_r and virtual task t_v, respectively.

3 Mining and Supporting Task-Stage Knowledge

3.1 The Process of Mining Task-Stage Knowledge

The key contents of a document can be represented as a feature vector of weighted terms in *n*-dimensional space. Term transforming and term weighting steps are employed to find the most discriminating terms [2]. We use the normalized *tf-idf* approach to derive a term's weight. The document database and the system log contain historical tasks with associated documents and usage data. The documents of a task are pre-processed and organized as a *Task Document Sequence* (**TDS**) according to the time they were accessed/generated during the task's performance. Clustering techniques are then employed to cluster the task document sequences of similar tasks belonging to the same task class. We propose a *Task-stage Hierarchical Clustering* (**TSHC**) algorithm that clusters task document sequences based on the similarity measures and retrieval time of documents. Each cluster represents a task-stage with associated documents of the task at hand. Finally, the feature vector of each task-stage is extracted from each cluster of documents to construct the task-stage profile. The relevant codified knowledge (documents) can then be retrieved to provide knowledge support for the task's execution according to the similarity measures (e.g., the cosine measures) of the task-stage profiles and document profiles.

3.2 Mining Task-Stage Knowledge: A Modified Hierarchical Clustering Technique

Given a virtual task t_v and a set of task instances with associated documents, the mining process identifies the number of task-stages and the corresponding profile of each task-stage for t_v. Figure 1 shows the mining procedure step-by- step.

Step1. Identifying the task-stages of each task instance. This step identifies the task-stages of each task instance. The proposed **TSHC** algorithm modifies the *Hierarchical Agglomerative Clustering* (**HAC**) algorithm by considering the *time variant* used to determine clusters, as shown in Table 1. Let TDS(t_r)=$<d_1, d_2, \ldots, d_m>$ be the task document sequence of a task instance t_r. The sequence is clustered into several clusters that represent the stage information of the target task t_r. A task t_r comprises several task stages: **TS**(t_r)=$<ts_r[1], ts_r[2], \ldots, ts_r[k]>$, where $ts_r[i]$ denotes the task-stage i of t_r. Each cluster represents a task-stage with associated documents, where $ts_r[i]$.docs denotes the documents in task stage i of task t_r.
We now explain the **TSHC** algorithm in detail.

***TSHC* algorithm:** Traditional hierarchical agglomerative clustering (**HAC**) algorithms [8] start by putting each item in its own cluster and iteratively merge clusters until all items are in one cluster. At each level, the **HAC** algorithms generate new clusters by merging clusters from the previous level. Several techniques can be used to determine if two clusters should be merged according to the distance threshold and the distance between any two points in two target clusters. Two clusters are merged if the minimum (maximum or average) distance between any two points in the clusters is less than or equal to the distance threshold. The HAC algorithms examine all clusters in the previous level to determine new clusters without considering the time dimension.

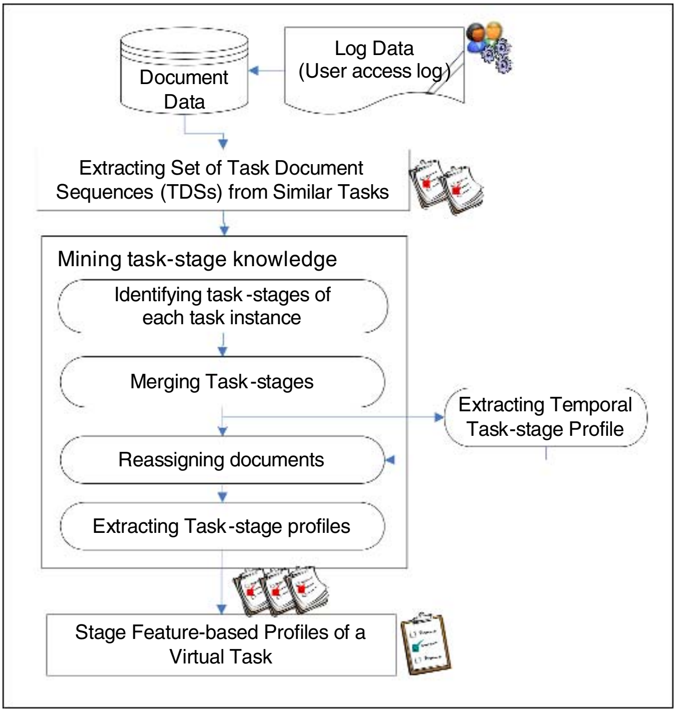

Fig. 1. Process of mining task-stage knowledge

The proposed **TSHC** algorithm adopts the **HAC** algorithm, but also considers the time dimension, since the **TDS** is generated by ordering documents based on their retrieval time during the task's performance. A time window size, *w*, is set to define the scope of the candidate clusters to be merged. Note that the index position in the sequence, rather than the actual time, is used to decide if the candidate cluster is within the time window.

Initially, each document in the **TDS** forms its own cluster. The **TSHC** algorithm then iteratively merges clusters until the number of clusters is less than *MinNumStages* (the minimum number of stages). The algorithm finds the clustering result in which the number of stages is not less than *MinNumStages* and not greater than *MaxNumStages* (the maximum number of stages). Due to the nature of our research problem, we prefer to cluster each **TDS** into three to six clusters. To derive the best clustering result, a clustering examination procedure is used to select the optimal level of clustering based on intra-similarities and inter-similarities. In the following, we explain the algorithm in detail. The algorithm is presented in Table 1.

During the clustering procedure, each document is initially regarded as a single cluster. Accordingly, **TDS** is a sequence of clusters. To merge two target clusters, c_i and c_j, into a new cluster, the time window is moved from the first to the last cluster in TDS to determine the number of merging candidates for each cluster. Then, all merging candidates are combined as follows. In the merging process (lines 9 to 12), the pair of candidates with the maximum similarity is selected from the merge list. If neither candidate in the selected pair has been merged with another candidate in any other pair on the list, then the two corresponding candidates in **TDS** are merged. If no clusters can be merged during the iteration, the *Threshold* is reduced by a constant (in our case, the constant is set to 0.01); otherwise, the threshold remains unchanged.

Table 1. Algorithm of *Task Stage Hierarchy-based Clustering* (**TSHC**) with Time-variant

```
   Input: TDS(t_r): Task Document Sequence of t_r
   Output: Task stages of target task
    function TSHC(TDS) {
1      Set a constant offset to be 0.01;
2       do {
3            foreach cluster c_i in TDS {
4              c_temp={c_{i-w},c_{i-w+1},...,c_{i-1},c_i ,c_{i+1},  ...,c_{i+w-1},c_{i+w}};
               //According to time window size to determine clusters
5              foreach cluster c_j in c_temp where c_i <> c_j {
6                Calculate similarity of c_i and c_j as similarity;
7                  if (similarity is higher than Threshold) then
8                     add {c_i, c_j, similarity} to the merge_list;
               } }
9            do while (merge_list is not empty) {
10                 Select and remove the pair (c_i, c_j) with maximum
               similarity from merge_list;
11             if (c_i and c_j had not been merged with another cluster)
12             then merge c_i and c_j in TDS
             }
13           if (there's no cluster merged) then
14               Decrease the Threshold by offset;
15           if (number of clusters in TDS ≤ MaxNumStages) then {
16                 Let Q value be the quality of clustering value of
                 TDS;
17               Add {TDS, Q} to the list result;
             } }
18       while (MinNumStages ≤ Num. of clusters in TDS ≤ MaxNumStages)
19       return result; }
```

The algorithm records the clustering results that contain clusters within the range *MinNumStages* to *MaxNumStages*. The quality measure, *Q* value, is derived to indicate the quality of each clustering result (lines 15 to 17). We select the clustering result with the best quality value. The clusters selected by the **TSHC** algorithm are the stages of a task instance. The profile of each task-stage can be derived by averaging the feature vectors of documents in the corresponding cluster. We now explain the *Q* value, which is used to verify the quality of clusters.

Clustering quality: Let the inter-similarity between two clusters c_i and c_j be defined, respectively, as the average of all pairwise similarities among the documents in c_i and c_j, denoted as $sim(c_i, c_j)$. In addition, let the intra-similarity within the cluster c_i be defined as the average of all pairwise similarities within c_i, denoted as $sim(c_i, c_i)$. The best clustering result is the one that maximizes the intra-similarities and minimizes the inter-similarities of the clusters produced at each level [4]. The *Q* value is defined as the summation of the inter-cluster similarities divided by the number of clusters and the summation of the intra-cluster similarities of clusters.

Step2. Merging Task-stages. Step 1 uses the **TSHC** algorithm to generate the task-stages of each task instance. This step generates the task stages of the virtual task t_v by merging the stages of task instances in t_v. $\mathbf{T}_v$ is the set of task instances of t_v. A kernel task instance, t_r, is selected to represent the virtual task t_v. Then, the stages of other task instances are merged with the stages of t_r. The kernel task is the task instance with the best average Q value of clusters among task instances. During the merging process, documents from the stages of other task instance t_f are merged with the stages of the kernel task instance. Several candidate stages in t_r can be chosen to merge with a document in t_f. For each stage i of t_f, each document d in $ts_f[i]$.docs is merged with the candidate stage k of t_r that has the maximum similarity measure cos(d.profile, $ts_r[k]$.profile). There are three cases in which the candidate stages of t_r are selected for merging with stage i of t_f. Let n_r and n_f be the number of stages (clusters) of t_r and t_f, respectively.

Case 1: $n_r > n_f$: The number of stages of t_f is less than the number of stages of t_r. The candidate stages of t_r are selected from stages i to $i+(n_r-n_f)$, where $n_r - n_f + 1$ stages of t_r are chosen as the candidate stages.
Case 2: $n_r = n_f$: The number of stages of t_f is equal to the number of stages of t_r. The candidate stages of t_r are selected from stages i-1 to i+1, where three stages of t_r are chosen as the candidate stages.
Case 3: $n_r < n_f$: The number of stages of t_f is greater than the number of stages of t_r. The candidate stages of t_r are selected from stages $max(1, i-(n_f-n_r))$ to $min(i, n_r)$.

Step3. Reassigning documents. This step reorganizes the documents in the task stages of t_v to enhance the homogeneity of each task-stage. Documents may be re-assigned to other task stages according to the similarity measures (cosine measures) of document profiles and task-stage profiles. A document d is assigned to the stage k of t_v that has the maximum similarity measure, i.e., cos(d.profile, $ts_v[k]$.profile), among all task stages of t_v. Consequently, we derive the final document set at each task-stage of the virtual task t_v.

Step4. Extracting the task-stage profiles of each virtual task. The profile of each task stage can be derived by averaging the feature vectors of documents in that task stage. For a stage k of a virtual task t_v, a task-stage profile, $ts_v[k]$.profile, is the vector obtained by averaging the feature vectors of documents in $ts_v[k]$.docs.

4 Experiments

Experiments were conducted using a real application domain from a laboratory in a research institute. The tasks concerned writing research papers or conducting research projects. Over 500 task-related documents were collected, each of which contained an average of ninety distinct terms after information extraction, and document pre-processing. Three virtual tasks were selected from the research domain to evaluate the effectiveness of the proposed method. For each virtual task, we selected a current task relevant to the virtual task to evaluate the effectiveness of the proposed technique. The effectiveness of knowledge support is measured in terms of precision, recall, and the F1-measure, as in IR research [12],[13]. *Precision* is the fraction of retrieved items (tasks or documents) that are relevant, while *recall* is the fraction of the total known relevant items retrieved.

$$precision = \frac{|retrieved\ items\ that\ are\ relevant|}{|total\ retrieved\ items|} \tag{1}$$

$$recall = \frac{|relevant\ items\ that\ are\ retrieved|}{|total\ known\ relevant\ items|} \tag{2}$$

The F1-metric, which is used to balance the trade-off between precision and recall, assigns equal weight to precision and recall and is given by,

$$F1\text{-}metric = \frac{2 \times precision \times recall}{precision + recall} \tag{3}$$

4.1 Experiment I: Effect on Time Window Size

A time window size *w* is set to define the scope of candidate clusters to be merged. When the window size is equal to *n*, the proposed **TSHC** algorithm is the same as the standard *Hierarchical Clustering Algorithm* (**HAC**). Thus, we conducted an experiment to evaluate the impact of the time window size parameter *w*. The best time window size is selected for use in the task-stage mining technique. Table 2 shows the experiment results with time windows of different size in terms of precision, recall, and the F1-measure.

Table 2. Results of knowledge support with time windows of different size (Top-30)

Parameters	**Virtual Task 1**			**Virtual Task 2**			**Virtual Task 3**		
Window Size	**Pre.**	**Re.**	**F1**	**Pre.**	**Re.**	**F1**	**Pre.**	**Re.**	**F1**
***w*=1**	**0.267**	**0.203**	**0.231**	**0.231**	**0.084**	**0.123**	**0.344**	**0.175**	**0.232**
***w*=2**	0.255	0.161	0.197	0.166	0.065	0.093	**0.344**	**0.175**	**0.232**
***w*=3**	0.255	0.199	0.224	0.156	0.063	0.090	0.244	0.124	0.165
***w*=n**	0.167	0.116	0.137	0.133	0.019	0.033	0.267	0.135	0.180

✳ **Pre.** denotes precision, **Re.** denotes recall, and **F1** denotes F1-metric.

Discussion: The results show that, generally, the smaller the time window size, the more effective the knowledge support will be in terms of higher precision, recall, and the F-measure. Accordingly, we set the time window size to one (*w*=1) and conduct further experiments. It is obvious that the larger the time window, the less effective the proposed method will be. In other words, when the time window size is equal to *n* number of documents in **TDS** of a task instance, the proposed **TSHC** algorithm is the same as standard **HAC** algorithm.

4.2 Experiment II: Effect on Task-Stage Knowledge Support

The objective of this experiment was to evaluate the effectiveness of the proposed **TSHC** algorithm. The *task-stage knowledge support method* is compared with a baseline method, *non-stage knowledge support method. Task-stage knowledge support* provides needed documents based on the task-stage profiles, whereas *non-stage knowledge support* provides knowledge support based on the task profile. The task-stage profiles or the task profile are used to deliver task-relevant knowledge for an executing task that is similar to the virtual task. Table 3 shows the results of using the *task-stage knowledge support method* and the *non-stage knowledge support method* in terms of precision, recall, and the F1-measure. Three virtual tasks were evaluated under various levels of top-*N* (*N*=10, 20, and 30) document support.

Table 3. Comparison of the task-stage method with the non-stage method

Virtual tasks		Task Stage Knowledge Support			Non-Stage Knowledge Support		
	Top-N	Precision	Recall	F-measure	Precision	Recall	F-measure
Virtual Task 1	Top-10	**0.333**	**0.080**	**0.128**	0.300	0.074	0.118
	Top-20	0.267	0.126	0.171	**0.317**	**0.162**	0.213
	Top-30	0.267	0.203	0.229	**0.289**	**0.229**	0.252
	Average	0.289	0.136	0.176	**0.302**	**0.155**	**0.194**
Virtual Task 2	Top-10	**0.300**	**0.047**	**0.080**	0.233	0.033	0.058
	Top-20	**0.217**	**0.069**	**0.104**	0.167	0.051	0.078
	Top-30	**0.231**	**0.084**	**0.123**	0.178	0.079	0.108
	Average	**0.249**	**0.067**	**0.102**	0.193	0.055	0.081
Virtual Task 3	Top-10	**0.433**	**0.074**	**0.126**	0.233	0.039	0.068
	Top-20	**0.350**	**0.118**	**0.176**	0.300	0.102	0.152
	Top-30	0.344	0.175	0.232	**0.367**	**0.186**	0.247
	Average	**0.376**	**0.122**	**0.178**	0.300	0.109	0.155

Discussion 1: In general, the average precision, recall and F1-measure of *task-stage knowledge support method* are better than those of the *non-stage method.* Furthermore, in the *task-stage knowledge support method*, the lower the number of supporting documents, the higher the precision value will be. The results show that the *task-stage knowledge support method* provides more effective knowledge support when there are fewer task-stage relevant documents.

Discussion 2: Figure 2 shows the average result of the two methods, respectively. The average precision of the *task-stage knowledge support method* is better than that of the *non-stage method* under top-10, top-20, and top-30 knowledge support.

Discussion 3: Figure 3 shows the average results of the *two methods for the three task stages*. The average precision of the *task-stage knowledge support method* is higher than that of the *non-stage method* in respective stages.

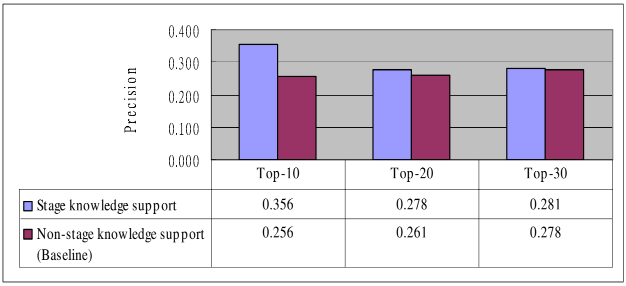

Fig. 2. Average precision of retrieved documents under various Top-N

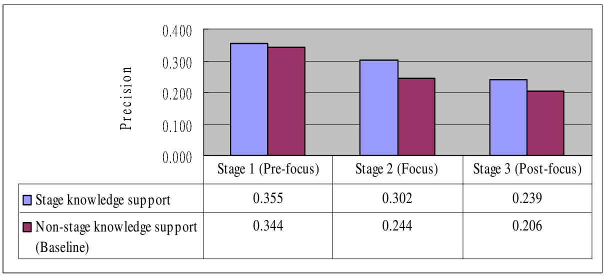

Fig. 3. Average precision of retrieved documents under different task stages

5 Conclusion and Future Works

Codifying knowledge in an explicit form to facilitate knowledge reuse is a widely adopted strategy in contemporary knowledge management systems (KMS). However, workers still encounter difficulties in accessing information from the vast amount of codified knowledge stored by organizations. In this paper, we have proposed a task-stage mining technique to address the problem. Valuable knowledge about each task-stage is extracted to build a task-stage profile to meet workers' initial needs in each task-stage. Preliminary experiments were conducted to evaluate the effectiveness of the proposed technique. In our recently published work [15], we developed a task-stage identification model to determine the changes of a worker's task stages. In the future, we will integrate the proposed task-stage mining technique with the task-stage identification model to empirically investigate the contribution of the task-stage knowledge support model.

Acknowledgements

This research was supported by the National Science Council of the Taiwan (Republic of China) under the grant NSC 94-2416-H-009-015.

References

1. Abecker, A., Bernardi, A., Maus, H., Sintek, M., Wenzel, C.: Information Supply for Business Processes: Coupling Workflow with Document Analysis and Information Retrieval. Knowledge Based Systems. 13(1) (2000) 271-284.
2. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. The ACM Press. New York (1999)
3. Bolloju, N., Khalifa, M., Turban, E.: Integrating Knowledge Management into Enterprise Environments for the Next Generation Decision Support. Decision Support Systems 33(22) (2002) 163-176
4. Chuang, S.-L., Chien, L.-F.: A Practical Web-based Approach to Generating Topic Hierarchy for Text Segments. *CIKM* (2004) 127-136
5. Davenport, T. H., Prusak, L.: Working knowledge: How Organizations Manages What They Know. Harvard Business School Press, Boston MA (1998)
6. Fenstermacher, Kurt D.: Process-Aware Knowledge Retrieval. Proc. of the 35th Hawaii Intl. Conf. on System Sciences, Hawaii, USA. (2002) 209-217
7. Jain, A.K., Murty, M.N., Flynn, P.J.: Data Clustering: A Review. *A*CM Computing Surveys 31(3) (1999) 264-323
8. Johnson , S. C.: Hierarchical Clustering Schemes. Psychometrika 2 (1967) 241-254
9. Kuhlthau, C.: Seeking Meaning: A Process Approach to Library and Information Services. Ablex Publishing Corp., Norwood, NJ (1993)
10. Liu, D.-R., Wu, I.-C., Yang, K.-S.: Task-based $\mathcal{K}$-Support System: Disseminating and Sharing Task-relevant Knowledge. Expert Systems with Applications 29(2) (2005) 408-423
11. Markus, M.L.: Toward a Theory of Knowledge Reuse: Types of Knowledge Reuse Situation and Factors in Reuse Success. Journal of Management Information Systems 18(1) (2001) 57-94
12. van Rijsbergen, C.J.: Information Retrieval, Second ed. Butterworths, London (1979)
13. Riloff, E., Lehnert. W.: Information Extraction as a Basis for High Precision Text Classification. ACM Transaction on Information System 12(3) (1994) 296-333
14. Vakkari, P.: Cognition and Changes of Search Terms and Tactics during Task Performance: A Longitudinal Case Study. Proceedings of the RIAO'2000 Conference. Paris: C.I.D (2000) 894-907
15. Wu, I.-C., Liu, D.-R. Chen, W.-H.: Task-stage Knowledge Support Model: Coupling User Information Needs with Stage Identification. Proc. of the IEEE 2005 Intl. Conf. on Information Reuse and Integration (IRI), Las Vegas, USA. (2005)
16. Zack, M.H.: Managing Codified Knowledge. Sloan Management Review 40(4) (1999) 45-58

Towards a Process Model for Identifying Knowledge-Related Structures in Product Data

Christian Lütke Entrup, Thomas Barth, and Walter Schäfer

Information Systems Institute, University of Siegen, Siegen, Germany
{luetke-entrup, barth, jonas}@fb5.uni-siegen.de

Abstract. Systematic and efficient use of expert's knowledge in industrial product and process design is gaining in importance in many knowledge-intensive business processes along the product life cycle (PLC). In the majority of cases, this knowledge is hidden in product and process data and not made explicit by the experts. In this paper an approach is presented to support knowledge intensive business processes by analyzing product and process related data on the basis of a general process model for identifying knowledge in data, aiming at search patterns to find similar historical cases for reusing their solutions. The process model is a top-down-approach from analyzing business processes to applying algorithms to specific data. Considering "offer engineering" in a scenario from automotive supplier industry as a knowledge intentsive task, and since in the product's development phase 70-80% of its cost is determined, this phase in the PLC is used as a guideline to demonstrate the usefulness of the process model. A tool is presented which allows an adaptive, fuzzy search process in numerical, alphanumerical, and geometrical data based on the evaluated support strategies. First results validate the potential benefit of this approach for conceptual planners in automotive supplier industry.

1 Introduction

Recently automotive suppliers have been confronted with increasing cost pressure by their customers, whilst the rising complexity of their products implies increasing requirements on the product and process knowledge. Hence, utilizing the aggregated knowledge is a key issue for enterprises to persist in global competition. To gain an order from a customer, a supplier has to present the best offer in terms of cost/performance ratio. Thus, "offer engineering" is one of the company's core business processes. Preparing a promising offer is only possible by (re-)using all available knowledge during all relevant processes. A highly significant part is the tacit knowledge and experience of employees about their specific domain. This knowledge comprises both economical and technical aspects from costing, engineering, and enterprise-wide resources. Results are made explicit in terms of persistent data (e.g. bill of materials (BOM), working plan), but underlying knowledge remains implicit. The approach presented in this paper aims at the support of knowledge-intensive business processes by analyzing the explicit data available. The underlying knowledge from domain experts necessary to derive e.g. a product's geometry, its material and the required machinery from a customer's request is – in contrast to this – not

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 189 – 200, 2006.

made explicit. Because the influencing factors vary between every two customer requests, the indeterminacy of the problem domain prevents from generating a thorough problem solving model e.g. in terms of a set of rules of the (very simplified) form "if material==x and resource==z then site:=y". This classifies the support of the "offer engineering"-process as an "open-world-problem" [1], which leads to the approach of supporting this process by case based reasoning (CBR) with fuzzy search techniques (see also section 3).

The processes involved in the design of new products can be speeded up by using solutions of previous "similar" products as the case base. Implicit knowledge is hidden in the data of earlier offers. In order to enable utilization of knowledge within data, relevant data has to be identified, gathered, analyzed, and prepared for further application. In [2], the relevance of supporting the "offer engineering"-process is explained and the solution approach of reusing knowledge through searching for similar products in product and process data is motivated. This article focuses on another part of a framework under development: a process model for identifying knowledge related structures in data and prototypical implementation of search algorithms depending on the data structures for the retrieval of similar cases.

This paper is structured as follows: next, the proposed process model is described. Chapter 3 gives a brief overview on some relevant topics in related research areas. Afterwards, the knowledge-intensive process under analysis ("offer engineering") is discussed in detail and the derived data model is developed. In chapters 6 and 7 the solution approach as well as its prototypical implementation is presented. The paper is concluded with a summary and several topics for future work.

2 A Process Model for Identifying Knowledge-Related Structures

To support the knowledge-intensive core processes with search algorithms on historical data, a process model has to be generated that defines the overall strategy for developing the optimal support system. A process model has several advantages, not only for analysis and implementation of software systems, but also for management purposes. In software engineering, process models like the waterfall model [3] or extreme programming [4] are common instruments for systemizing and formalizing the process of software development, thereby improving productivity and quality while simultaneously reducing the complexity. This is achived by a step-by-step procedure defining milestones and break-points and leads to the management purposes of process models. Within a structured sequence of processes, planning and controlling as well as validation is possible, whilst the milestones and break-points definded in the software development process can be leveraged to the management level and used as operating figures.

In the context of supporting knowledge intensive processes by applying search algorithms to business data, a process model has to be defined that clearly delimits the single phases of the general task. A model is presented showing the approach of identifying knowledge-related structures within data for reuse in search algorithms.

In a first step the process itself is subject of the analysis, decomposing it into (fine-) grained units for further knowledge-related examination. Methods of regular business process analysis and modelling techniques like the event driven process chain (EPC, [5]) can be used to derive a model that serves as the basis for the next step, the determination of the knowledge itensive tasks (KIT, following [6]). Identifying these tasks regarding the intensive use of knowlegde within a process chain mainly depends on the context respectively the aspects the project is concerned with. A classification of aspects of KIT can be found in [7].

After identification of the KIT in question a data model has to be prepared that contains all relevant data every single process unit respectively the KIT is concerned with. The data model should contain the cumulative data of the process step, including that of (formalizable) routine work in contrast to KIT to reduce complexity of later enhancements. The attributes storing the in- and output of KIT can now be analyzed to determine suitable search algorithms. The overall intention of this process model, supporting KIT through presenting previous solutions of similar problems, can be reached by applying matching search algorithms on specific data. In addition, iterating through different algorithms results in a reduced list of similar problems, that way shortening the list of choices the user can pick from.

Fig. 1 presents the concept of the process model. The business process (BP)-steps are identified, with *f(input)=output* describing rather simple "routine work" as the explicitly given relation *f*. BP-step 2 is identified as a KIT, where the user has to infer the output from input by applying task-specific knowledge. All BP-steps are looking up from and storing data into a data store, using it to accomplish their assignments, but KIT also use the data store as an information base for non-routine tasks. This information in terms of database attributes or documents (in Fig.1: Attributes L1/L2) is also used to determine the appropriate search algorithms to retrieve the matching historical cases, which are then presented to the user as templates for solving specific problems.

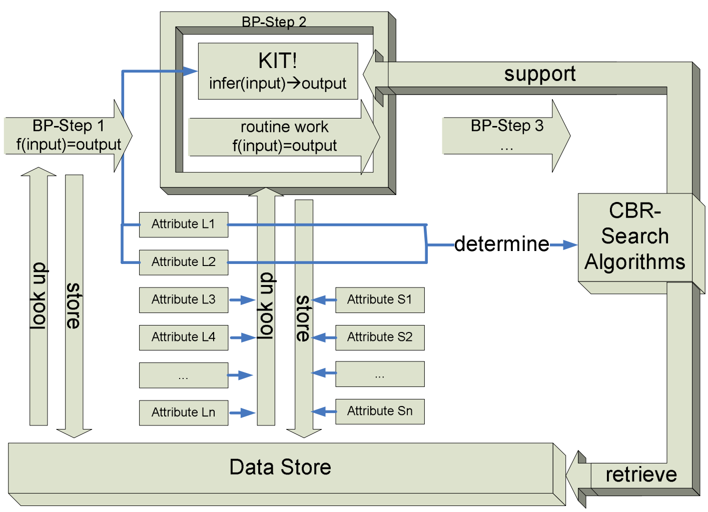

Fig. 1. Abstract concept of the process model, including business process (BP) steps, knowledge intensive task (KIT), the data store, and CBR-algorithms

3 Related Work

A variety of approaches to support knowledge intensive business processes are currently under research. Some focuses relevant to the work presented here can be distinguished, namely process models in knowledge engineering (section 3.1), fuzzy logic (section 3.2), and methods and tools for data mining and knowledge discovery in databases (KDD, section 3.3). This approach aims at applying search techniques for the retrieval phase of case based reasoning (CBR, section 3.4).

3.1 Knowledge-Related Process Models

As already mentioned, process models are developed in order to support (or enforce) standardized and systematic procedures when designing and building complex systems, e.g. in software engineering or mechanical engineering. A wide variety of process models exist in the domain of knowledge management and knowledge engineering: The Knowledge Modelling and Description Language (KMDL®, [8]) includes a process model focusing the in-depth analysis (and hence modelling) of the (business) processes within an enterprise and the needed knowledge. PROMOTE® (Process-oriented methods and tools for knowledge management) is a comprehensive framework including modelling, management aspects, and methods for supporting knowledge as well as knowledge-intensive processes (s. e.g. [9]).

In the domain of engineering, MOKA aims at a process model especially for capturing knowledge in engineering (s. e.g. [10]).

In addition to the aforementioned approaches, the work presented here aims at an integrated approach covering (business) processes along the product lifecycle in engineering domains, adequate models of these processes, relevant knowledge and data. Overall goal is a service-oriented toolbox for the analysis of product and process data with composable services as parts of workflows. Hence, the focus is on methods for knowledge reuse, the integration into workflows based on a distributed, service-oriented software platform, and a model covering the overall process. Existing methods for modelling of processes, knowledge etc. will be reused and extended only if necessary to be compliant to standards if possible.

3.2 Fuzzy Logic

Applying techniques from Fuzzy Logic is an approach to handle uncertainty and vagueness in data [11]. Along a product's life cycle, product- and process-related data is typically subject to many changes. Hence, an adequate support to reuse knowledge should handle this kind of vagueness within data. So far, the approach presented here utilizes as a first step a fuzzy string search [12] for the analysis of alphanumerical product- and process-related data. Since the importance of alphanumerical information (e.g. product description, product name, description of a working plan) is considered by domain experts to be higher compared to numerical information (e.g. dimension and weight of a product), the analysis of this part of the data was given priority. Similarity search based on fuzzified numerical data will be integrated and evaluated regarding its benefit for the efficiency of the search process.

3.3 Data Mining and Knowledge Discovery in Databases (KDD)

KDD refers to the process of finding knowledge within data by applying machine learning algorithms to data sets. These algorithms can be classified as supervised and unupervised learning, both dealing with the classification of data sets. With supervised learning, training data and its classification is given to generate a concept (=set of rules) for the classification of new data, whereas unsupervised learning algorithms classify data by applying heuristics and rules and iteratively adjust the results to the data set to generate classification rules [13].

As “offer engineering” being an “open-world-problem”, where a set of rules can hardly be derived from data, the approach presented here is focused on applying solutions of similar previous solved problems to the new situation rather than generating a set of rules for universal application in this domain.

3.4 Case Based Reasoning (CBR)

CBR describes the way of solving new problems based on the solution of similar historical problems. The basic principal of this approach is the fact that similar problemes have similar solutions. In knowledge-itensive processes, the knowledge and experience influencing the result or solution becomes manifest in the documents generated in the course of the process. These documents can be taken as the basis for the solution of the new problem. The CBR-cycle consists of the four phases RETRIEVE a past solution, REUSE it, REVISE the new solution and RETAIN it in the case base [14].

In this approach, the phases reuse, revise, and retain are covered by the companies regular business information systems, whereas there is demand for search techniques that deal with the retrieving of the most similar historical cases.

4 Analysis of the Knowledge-Intensive Process of Offer Engineering in Automotive Supplier Industry

In automotive industry – similar to other industry sectors – the Product Lifecycle (PLC) can be roughly divided into the phases of planning and design, manufacturing, distribution, and maintenance up to recycling (s. e.g. [15;16]). In this context, the PLC will be used as a guideline to identify knowledge-intensive business processes.

In the early phases of the PLC (in conceptual planning and design) about 70% of the product cost is determined [16]. Hence, supporting early phases is of particular importance for any enterprise planning, engineering, and manufacturing complex products. Due to the complexity of the products, engineering an offer is already a complex, demanding, and knowledge-intensive task itself. Along the PLC, many more knowledge-intensive business processes can be identified: E.g. optimal utilization of resources (e.g. machinery, material, personnel), logistics between customers and suppliers, or strategic planning of an enterprise’s future product portfolio (s. e.g. [7]).

In this context, the focus is put on supporting the process of offer engineering. In Fig. 2, an overview model of this process, its position in the overall lifecycle, the necessary knowledge, the resulting documents, and the product and process data stored in PDM/PLM and Enterprise Resource Planning (ERP) systems is given.

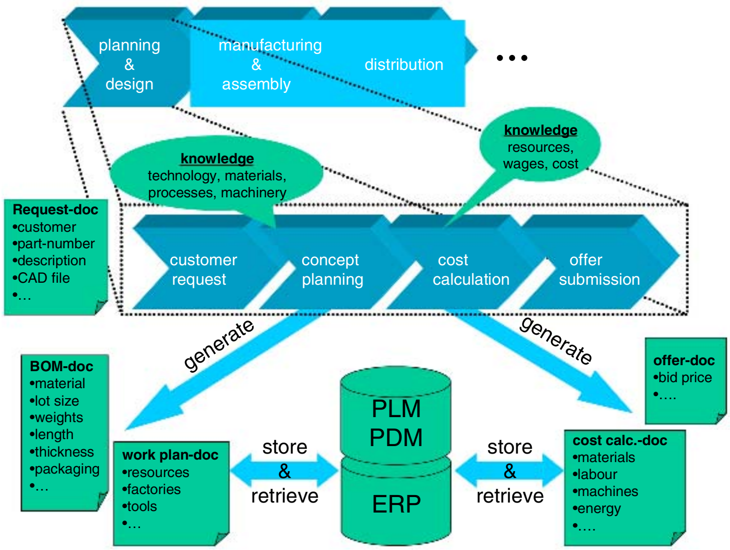

Fig. 2. Overview model of the offer engineering process with relevant knowledge and generated documents and data

Support to this process can be provided by analyzing the data of previous products considering specified features of a newly requested product. In the process of offer engineering, the customer's request must be "translated" into a (preliminary) bill of materials and a working plan by conceptual planners. This translation mainly relies on their experience and ability to remember (or find) similar products providing a template as a basis for the new offer. Similarity in this context can be based on a variety of different attributes of a product, divided into the following two types:

- Class list of characteristics, e.g. customer, material, weight, lot size, production site, product description.
- Geometrical attributes, e.g. size, shape, position of a component in a product.

The cost calculation then determines the bid price being mostly the decisive criteria to win a bid. Without the ability to search and explore existing data, a systematic and efficient re-use of knowledge within this data is impossible. Searching the product and process data considering the aforementioned attributes enables re-using the knowledge without the necessity to make the knowledge explicit and maintain this explicit knowledge base.

5 The Data Model and Knowledge-Related Structures

As shown in Chapter 4, the process of offer engineering as a knowledge intensive task is mainly concerned with the preparation of a (preliminary) bill of materials (BOM) and a working plan by conceptual planners, followed by the cost calculation.

In Fig. 3, an excerpt from the data model is presented, showing the bill of materials and its constituent parts, as well as influencing cost data and material masters. A bill of materials is composed of at least one component, which can be materials, bought-in-parts, packaging, trading goods, and assemblies. An assembly in turn consists of a complete (sub-) calculation, this way reflecting the recursive structure of the requested product.

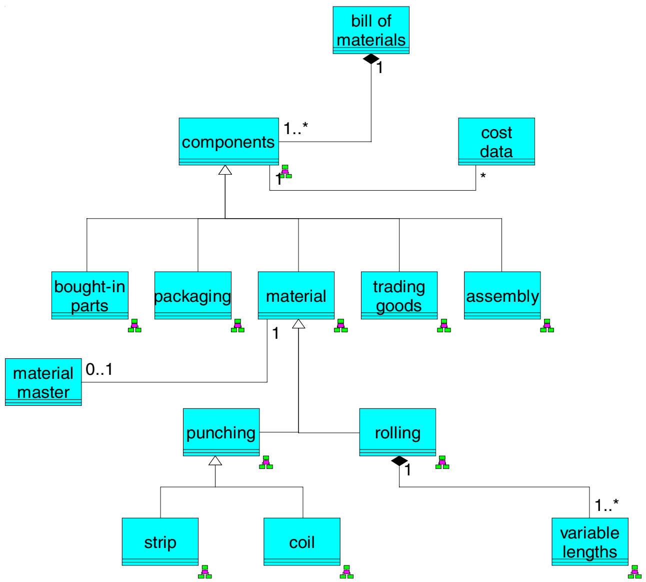

Fig. 3. Excerpt from the data model, showing the bill of materials, its constituent parts, influencing cost data, and material masters

The material appears either as punching or rolling material, where punching can further be segmented in strip and coil punching. Rolling material contains at least one variable length, meaning that in one machine cycle there can be more than one rolling material processed, as long as no further setting-up is needed. The material master can provide information about specifc materials available in the company, assuming the material in question is part of the stock. Finally, cost data influences all of the above mentioned parts of a BOM.

As a next step the knowledge-relevant attributes in the data model have to be identified to generate the search space for the algorithms. Whilst regular attributes, i.e. attributes generated in regular tasks, can be described with an explicit relation *f(input)=output* or are independent at all, e.g. attributes specified by the customer, knowledge related attributes are dependent but informal. They depent on the specific context the task is associated with, and they are informal as there can be no relation found to formalize the course of action. Taking this into consideration, every attribute of the data model can be classified as either regular or knowledge related. Furthermore, interviews with the employees involved in this particular task are of high usefullness regarding their knowledge of the process.

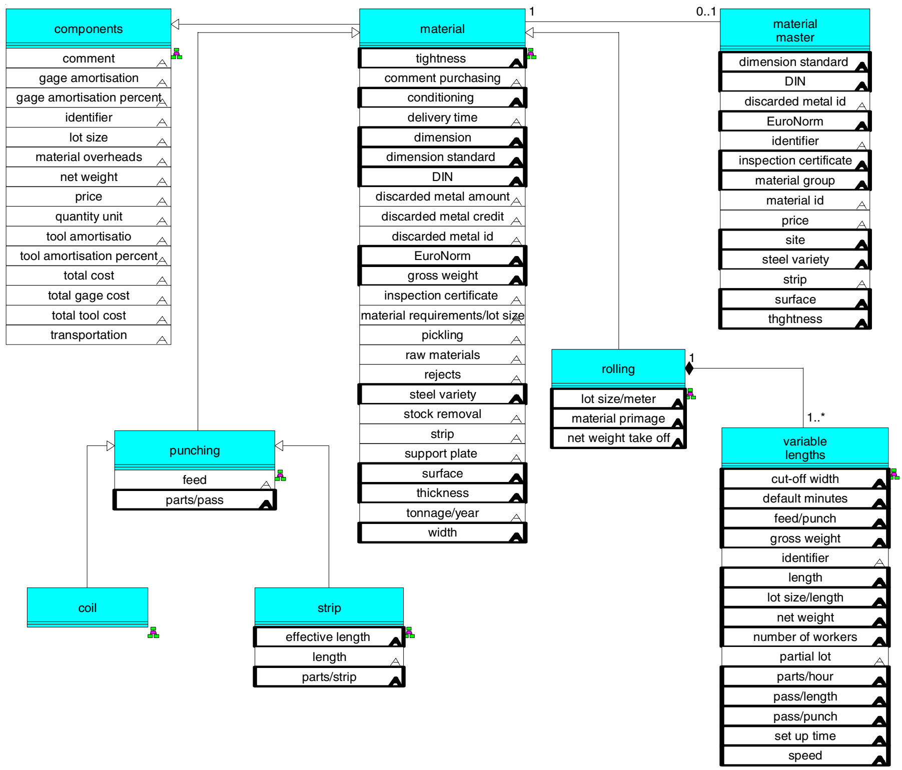

Fig. 4. Class diagram of materials' hierarchy and material masters

Fig. 4 shows the resulting class diagramm extract, restricted to the class hierarchy of materials and the influencing material master. Knowledge-related attributes appear with bold border.

6 A Solution Approach Using Search Techniques to Support Knowledge Reuse

Manual search processes for similar products performed by domain experts depend on a variety of criteria: E.g. availability and quality of provided data, an expert's knowledge and experience, the requested manufacturing technology, and the material to name just a few. This leads to the conclusion that an efficient search tool has to support a wide range of different search processes, especially it should allow arbitrary combinations of attributes with thresholds above which similarity is accepted.

In this process, two different data sources can be considered for the search: The data submitted with the customer request (textual and numerical attributes from a products class list of characteristics) and the product's geometrical properties (information within a CAD file). Analyzing the textual attributes of the request, the product name provided by the customer and the descriptions/comments added by conceptual planners are valuable search attributes. Most automotive companies describe similar products in requests with slightly different phrases. This fact leads to a search for similar products that depends on iteratively modified product names and descriptions. As a similarity measure the Edit- (or Levenshtein-) distance can be utilized for textual attributes (s. 6.1). By means of geometry, the CAD file can be analyzed with algorithms that apply different views of similarity: D2, Lightfield-Descriptor, and Spherical Harmonics (s. 6.2).

6.1 Analysis of Product Descriptions Based on the Edit-Distance

The denotation of the product description provided by the customer is mostly different from customer to customer, even if the same part is requested. E.g., one customer orders a 'front door', another one requests a 'Frontdoor', which probably is a similar product, but under the aspect of finding similar parts by means of searching for historical products under the provided description, a regular search by comparing the full text does not yield in the maximum possible results. A second point is typing errors, which can never be avoided completely. A regular text search also fails here. Therefore, a fuzzy string search is needed that generates result lists containing all strings similar to the input-string within a certain threshold.

For these reasons, the Levenshtein-algorithm [17] is applied to determine the Edit-distance between two text attributes. This algorithm computes the minimal number of edits necessary to transfer one string into another. In relation to the number of characters in the longer string, a distance in the interval [0, 1] is generated. A distance of 0 means total equality, a distance of 1 means total inequality. A similarity measure is thereby introduced for finding similar products based on textual attributes.

6.2 Analysis of Geometrical Data

The geometry of a product is specified in terms of CAD file(s). Shape and size of a product constrains the possible set of machines, manufacturing technologies, production sites etc. Therefore, one can infer from similar CAD documents to similar bills of materials respectively similar working plans.

The basis for the applied algorithms is not the CAD file itself, but a description that can be derived from the original file by extracting algorithm-specific properties. Taking into account the time-consuming procedure of building the properties, this is done offline for existing products, and the results are stored in a database. For the similarity analysis of a new CAD file, this is analyzed online and compared to the property-descriptions of existing products already calculated.

The algorithms chosen for geometrical analysis are D2, Lightfield-Descriptor and Spherical Harmonics. They differ in their approach of analyzing CAD files.

D2 [18] determines points on the surface of the object and computes distances between them which are stored in the descriptor [19]. When comparing two descriptors, a similarity measure is calculated by relating the distances of the two objects for the same points. The Lightfield-Descriptor [20] builds on the assumption that if two objects are similar, they are similar in all possible visual perspectives. Thus, it creates a descriptor by examining 20 different visual perspectives. Spherical Harmonics [21] calculates a descriptor by decomposing the object into a set of functions based on concentric spheres, while discarding orientation information. This results in an orientation invariant descriptor.

7 Prototypical Implementation

The presented procedures for similarity analysis are combined and implemented within a prototypical search tool. Due to the presented generic character of the search process it is realized as an interactive, modular, and extensible system; a user can arbitrarily combine attributes, search values, search methods, and threshold values. The search methods act as filters shortening the list of similar products successively with every attribute taken into consideration. This way, the result list can be minimized until sufficient information ist presented to the user. To support the user's evaluation of the search results, a viewer is added presenting the CAD file in question and that of the selected case for a visual comparison.In fig. 5 the dialog for evaluating the search results is depicted. The CAD document a user searches similar parts to is shown in the upper left part whereas the upper right part displays the CAD document associated to the search result selected in the list below. In this case the part on the right side was found based on geometrical similarity calculated by the Spherical Harmonics method (s. 6.2). First evaluation by domain experts demonstrates the potential of this approach to speed up offer engineering.

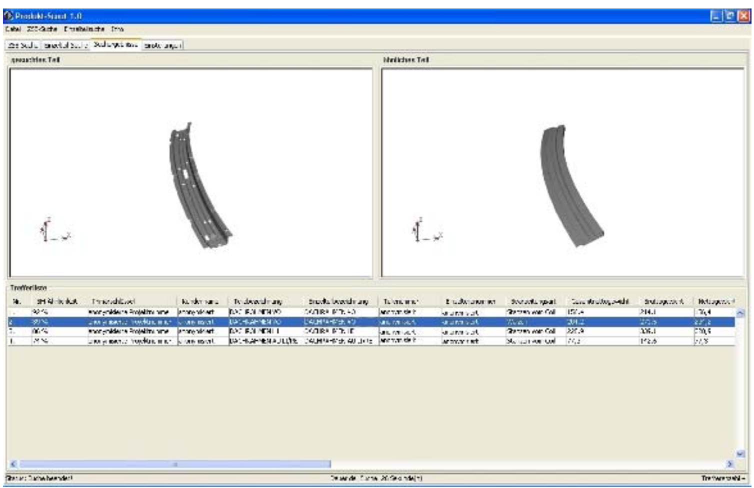

Fig. 5. Snapshot of the search tool's GUI for comparing and evaluating search results by means of geometrical, numerical, and textual data

8 Conclusion and Future Work

In this paper a process model was presented to provide support of knowledge-intensive business processes by re-use of knowledge through search for similar products in product and process data. First, konwledge intensive tasks in the business process chain as well as relevant data for their support were identified. Then, analysis methods were presented used in the iterative search process. Methods for the analysis of textual, numerical, and geometrical data can be combined to an arbitrary, iterative search process using a prototypical tool including 3D visualization of search results.

Next steps are the further development and validation of this process model in other engineering domains (e.g. in metal casting as well as tool design and construction in sheet metal forming) and additional processes to be supported (e.g. in product design and strategic planning). In order to improve the search process itself it is planned to analyze a user's search strategy (e.g. the order of attributes searched for, used similarity metrics, precision) and to adapt the search tool for future searches automatically.

Acknowledgement

Parts of the work presented here were funded by the German Federal Ministry of Science and Research (grant #01AK806C).

References

[1] Hinrichs, T.: Problem Solving in Open Worlds: A Case Study in Design, Lawrence Erlbaum Associates, 1992.
[2] Barth, T., Lütke Entrup, C., Schäfer, W.: Supporting Knowledge-Intensive Business Processes in Automotive Supplier Industry by Analyzing Product and Process Data. To appear in: Proc. Of I-Know 2006, Track on Business Process Oriented Knowledge Infrastructures.
[3] Royce, W.W.: Managing the Development of Large Software Systems: Concepts and Techniques. In: Proc. WESCON, IEEE Computer Society Press, Los Alamitos, CA, 1970.
[4] Beck, K.: Extrame Programming Explained. Addison-Wesley Professional, 2004.
[5] Scheer, A.W.: ARIS –business process frameworks. Springer, Berlin, 1999.
[6] Toward a Technology for Organizational Memories. IEEE Intelligent Systems 13(3), 1998, pp. 40-48.
[7] Remus, U.: Process-oriented Knowledge Management: Concepts and Modelling, (PhD thesis, in german), University of Regensburg, 2002.
[8] Gronau, N.; Müller, C.; Korf, R.: KMDL – Capturing, Analysing and Improving Knowledge-Intensive Business Processes. J. of Univ. Comp. Science, 11(4)2005, S. 452-472.
[9] Karagiannis, D.; Telesko, R.: The EU-Project PROMOTE: A Process-oriented Approach for Knowledge Management. In: Reimer, U. (ed.): Proc. of the Third Int. Conf. on Practical Aspects of Knowledge Management (PAKM2000).
[10] Sainter, P.; Oldham, K.; Larkin, A.; Murton, A.; Brimble, R.: Product Knowledge Management within Knowledge-Based Engineering Systems, in: Proc. ASME 2000 Design Eng. Tech. Conf. (http://www.kbe.coventry.ac.uk/moka/Documents/Papers/ dac14501.pdf, visited 2006/08/03).
[11] Kahraman, C. (ed.): Fuzzy Applications in Industrial Engineering, Springer, 2006.
[12] Navarro, G.: A Guided Tour to Approximate String Matching. In: ACM Computing Surveys, Vol. 33, No. 1, March 2001, pp. 31–88.
[13] Klösgen, W., Zytkow, J.: Handbook of Data Mining and Knowledge Discovery, Oxford University Press, New York, 2002.

[14] Aamodt, A., Plaza, E.: Case Based Reasoning: Foundational Issues,Methodological Variations, and System Approaches. In: AI Communications 7 (1), pp. 39-59, 1994.

[15] Vajna, S.: Approaches of Knowledge-based Design. In: Proc. Int. Design Conf. 2002, Dubrovnik, pp. 93-100.

[16] Murman, E. M., Walton, M., Rebentisch, E.: Challenges in the Better, Faster, Cheaper Era of Aeronautical Design, Engineering and Manufacturing. MIT, Engineering Systems Division, Working Paper Series, 2000.

[17] Apostolico, A.: General Pattern Matching. In: Atallah, A. (eds.): Algorithms and Theory of Computation Handbook. CRC Press 1999, pp. 13-5 - 13-10.

[18] Osada, R.; Funkhouser, T.; Chazelle, B. und Dobkin, D.: Matching 3D Models with Shape Distributions. In: Shape Modeling International (2001), pp. 154-166.

[19] Vranic, D.: 3D Model Retrieval. (PhD thesis), University of Leipzig, 2004.

[20] Chen, Ding-Yun: Three-Dimensional Model Shape Description and Retrieval Based on LightField Descriptors. (PhD thesis), NTU CSIE, 2003.

[21] Princeton Shape Retrieval and Analysis Group: Shape Representations. http://www.cs. princeton.edu/ gfx/proj/shape/, (visited 2006/03/23).

Common Knowledge Based Access to Disparate Semantic Spaces: The Ontology Switching Approach

Thomas Mandl and Christa Womser-Hacker

Information Science, University of Hildesheim
Marienburger Platz 22
D-31141 Hildesheim, Germany
mandl@uni-hildesheim.de

Abstract. We present the ontology switching system MyShelf which is implemented using technological standards of the semantic web. MyShelf solves the problem of semantic heterogeneity by integrating different cognitive views on one data collection. Homogenization has been the most widely applied approach to create common semantic spaces in order to overcome the limitations to sharing knowledge posed by semantically heterogeneous concept structures. Ontology switching implemented in MyShelf manages diverse ontologies by integrating them and enables the user to access them in parallel. Most important, the user can change his point of view by switching to another concept structure at any point and explore different viewpoints according to his current context. An ontology switching system dynamically reorganizes the concept structure and populates it with knowledge objects. The paper describes the design options for ontology switching systems, presents prototypes, evaluation results and points to future developments. This work integrates semantic web ideas and human computer interaction.

Keywords: Ontology switching, semantic web, ontology, semantic heterogeneity.

1 Introduction

Ontologies are structured collections of concepts which describe the world. Ontologies define a perspective on the domain under consideration and have been an important tool for knowledge management. Often, ontologies are expressed as hierarchically organized taxonomies. In this paper, we treat deal with taxonomies.

The disparity of ontologies has often been discussed as a major problem for mutual understanding and the sharing of knowledge. Most knowledge domains have been organized into ontologies several times form different perspectives. Each perspective is justified within its own context. Ontologies also differ over time. Information system engineers are faced with a large variety of ontologies for each domain. Sharing knowledge across different perspectives represented by semantically heterogeneous ontologies remains tedious. This challenge of semantic heterogeneity has led to many intellectual and technological solutions for different forms of semantic unification [7].

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 201 – 212, 2006.

Heterogeneous ontologies occur in many areas. The most typical attempt to resolve this problem is standardization and concentration on one ontology. However, this may not always be possible and aspects of and perspectives on the domain may get lost. Some of the problems arising from ontology mismatch are presented here with examples from information science:

- Challenges due to different terminology (e.g. usability, human-computer interaction, or interface design)
- Different hierarchy formation or poly-hierarchical structures (e.g. information science -> information retrieval -> information retrieval evaluation or information science -> evaluation -> information retrieval evaluation)
- Different assignment of segments (e.g. information retrieval -> user interfaces for information retrieval or user interfaces -> user interfaces for information retrieval)

Further examples can be found in many information systems. Cognitive and also cultural differences lead to different conceptual structures. In an experiment, it was shown that users from different cultural groups order recipes differently and based on different parameters [21]. In an experiment set in South Africa it has been shown that some users even draw family relationships in different ways than the traditional tree structure [19].

As a consequence, heterogeneity treatment is necessary to overcome the incompatabilities between different ordering systems [16, 7, 18]. A traditional but expensive approach is the creation of concordances. Automatic systems based on machine learning are being developed [8, 12]. Related work has also been presented as semantic interoperability.

Intellectual methods for ontology creation, adaptation and their population with knowledge objects are based on the similarity between concepts and objects. Automatic methods also try to measure the similarity between the concepts and the knowledge objects. For example, objects similar to each other are grouped together and all associated with he same concept.

Most solutions tend to create one unified ontology for the domain under consideration [3, 18]. This ontology merging process poses several problems for the user who explores knowledge objects organized under the concepts of ontologies:

- An new modified ontology created e.g. by ontology fusion is a new artificial construct which needs to be learned by the user. On the other hand, acquired knowledge about the structure of the established ontologies cannot be used anymore. This is a serious problem for users even if the new structure is superior to the old ontologies in many respects
- The possibility to explore a different perspective are lost. A user cannot enrich his understanding of a domain by exploring the perspective of another ontology and maybe learn how and why an ontology engineer has chosen this conceptual structure.

Our ontology switching approach relies on the same technological building blocks like most ontology merging methods. However, it does not try to unify the semantic representations by fusing the heterogeneous ontologies. Much rather it integrates all relevant ontologies into one user interface and allows the user to chose his current perspective (see figure 1). That means, the user can change the viewpoint form which he explores the knowledge objects.

The user selection can be changed at any point in the system. Switching causes a dynamic reorganization of the objects according to the selected perspective. The switching process invokes a process which display the new concepts, populates them with the knowledge objects and sets the current focus based on the position in the previous ontology. Switching can be done either based on a class or an object level. In the first case, the user is presented with a view on the class from the selected ontology most similar to the one he had viewed in the previous ontology. In the second case, the same object is still shown to the user, however, it has been dynamically introduced into the selected ontology. That way, MyShelf solves semantic heterogeneity problems within a heavily knowledge based system.

The remainder of the paper is organized as follows. The second section discusses the heterogeneity of ontologies. Section three presents the ontology switching approach as a theoretical concept. Section four shows the implementations of the concept in information systems. The outlook points to areas of further development.

2 Ontology Switching

The central idea of ontology switching is that the user should be enabled to change the ontology with which he wants to explore the knowledge objects in a unified collection. The user can select the appropriate perspective for his current context.

Enabling ontology switching within each concept is motivated by principles from human-computer interaction. The interaction, parameters once set by the user should be kept and not entered again. Within ontology switching, the parameter setting is undertaken by selecting concepts thus expressing interest for them. Simply changing the underlying ordering structure by selecting another ontology should not invalidate the expressed interest and force the user to search for the concept of his interest in the newly selected ontology again.

2.1 Ontologies in Information Systems

Ontologies often need to be created intellectually because they require domain knowledge. The assignment of objects to ontologies has been subject of intense research. In the recent past, ontologies have gained more attention and even the automatic construction of ontologies is now feasible. Ontology learning deals with these two aspects and can be seen as an intersection of machine learning and knowledge management. Ontology learning comprises the following research topics:

- Automatic categorization
- Assigning Objects to ontology concepts (ontology population)
- Establishing relations between ontologies [4]
- Construction of new ontologies [2, 11]
- Adaptation of existing ontologies [1]

Ontologies play an important role in information systems. Sometimes they are visible for the user and in other cases they support the system. When the ontology is a crucial tool for the access to knowledge objects, the user is confronted with a selection problem.

The user needs to decide which information system he wants to explore and he then needs to accept the governing ontology. Ontology switching integrates the collections as well as the ontologies in order to create a single point of access.

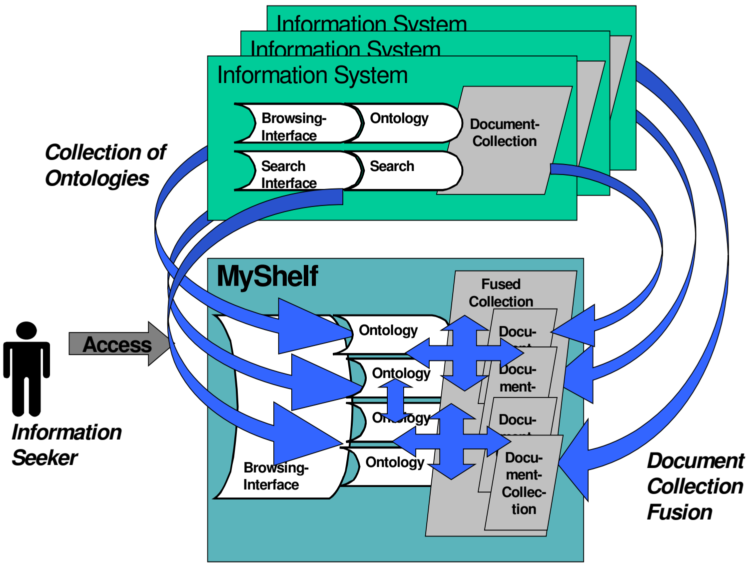

Fig. 1. Integration of Collections and describing Ontologies

2.2 Ontology Switching

Ontology switching [14] deals with heterogeneity in several levels. It integrates several overlapping collections indexed with different ontologies. Ontology switching provides a browsing system with a user interface containing all classifications. Between the ontology entries and the documents, relations need to be established where they cannot be found in the original system. For example, when a book is only available in one library, we need to index this book with terms from other library catalogues as well. Ontology switching can integrate the perspective of different departments within a company. An engineer, an auditor and a manager all take a different approach toward documents available to them.

Ontology switching results in the following value added services:

- One browsing user interface serves for several ontologies
- The reach of each ontology is increased
- Ontology switching is made possible and the user can change his perspective
- Ontology switching is based on the decision of the user and not on system limitations
- Thematic selections remain effective during switching and are not lost

Ontology switching is an application based on ontology learning. It exploits knowledge engineering and the results of the assignment of objects to ontology concepts. That way, ontology switching systems, create transparency in semantic heterogeneity.

Ontology switching also supports ontology learning. Continuous use of Ontology switching systems allow the identification of highly used ontologies as well as fractions of ontologies which are accessed more than others. These popularity measures can be exploited in further ontology refinement.

Multiple assignments are also possible within ontology switching. Often, machine learning systems assign vague membership values to objects. These vague assignments can be integrated in several ways into ontology switching systems:

- Complete ranked list within a concept
- Object is only assigned to the concept where it receives its maximum membership value
- Combinations between the two approaches above

2.3 Concepts for Ontology Switching

Ontology switching can be based on different machine learning algorithms for the mapping between the concepts and the assignment of objects to concepts. More important for the user, the switching can be based on the concept or class or the knowledge object. In the first case, it is not always sure that the same concept exists within the other ontology. Then, the most similar concept needs to be found. In the second case, where the switching takes place at the object level, it is always guaranteed that the object is also present in the other ontology. The concept to which it is assigned can be presented to the user. Problems can occur when multiple assignments are allowed and the object is assigned to more than one concept. Further experience with the prototypes will show which approach is more adequate.

3 Implementation of the Ontology Switching Approach

Ontology switching was first implemented within MyShelf, which allows access to information science content. Three different prototypes have been implemented, one static HTML system, one based on a database and PHP scripts and a last one applying semantic web technology. First, the knowledge engineering for the domain is described.

3.1 Knowledge Engineering for Ontology Switching

First, some 6000 books from the university library of Hildesheim were identified. These information science books were also searched in the library of the university of Constance. In a semi-automatic process, all books were assigned to categories in both ontologies [5]. First, keyword searches with keywords extracted from the information science part of the library of the university of Constance were conducted within the keywords assigned to books at the University of Hildesheim. The hit lists were manually evaluated and assigned to categories.

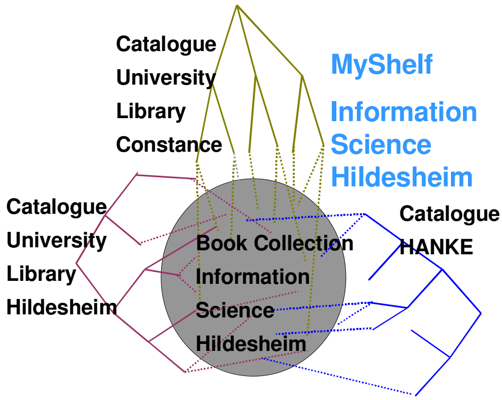

Fig. 2. Three taxonomies lead to the same collection

Furthermore, a new information science classification was designed which better models the specific teaching and research profile of the information science at the university of Hildesheim. This newly developed ontology is called HArmonized NomenKlature information sciencE (HANKE). It also mirrors recent developments in information science and information technology. As a result of this work, some 6500 documents were assigned with terms or categories from three classifications, the HANKE classification, the older classification for computer and information science called KID (cybernetics, information and documentation), developed and used by the university library of Constance, Germany and the catalogue of the university library of Hildesheim [5].

The sizes of the classification systems cannot easily be compared. The HANKE classification was dedicated to information science content identified and optimized for it. The other ontologies cover a wider range of subjects and contain more slots. Their advantage lies in the fact that they are physically represented in the shelf structure of the two libraries. The library catalogue of the University of Hildesheim encompasses 36 top level entries of which 32 were displayed because they contain information science content. The optimized HANKE classification encompasses 18 top level entries and has three more hierarchical levels.

3.2 Static and Dynamic MyShelf Prototypes

From a librarian's point of view the virtual shelf performs the function of a hierarchical catalogue. In this case the catalogue is not based on only one classification but on three (see figure 3). The basic design of the system follow the example of Yahoo's web catalogue1. The hierarchical structure of the classifications (i.e. categories and subcategories) is represented by the varying size of file symbols and fonts. The path at the top of the site ("bread crumbs") helps users navigate and orient themselves in the classifications. The prototype links to the library catalogue (see figure 3).

[1] http://de.yahoo.com

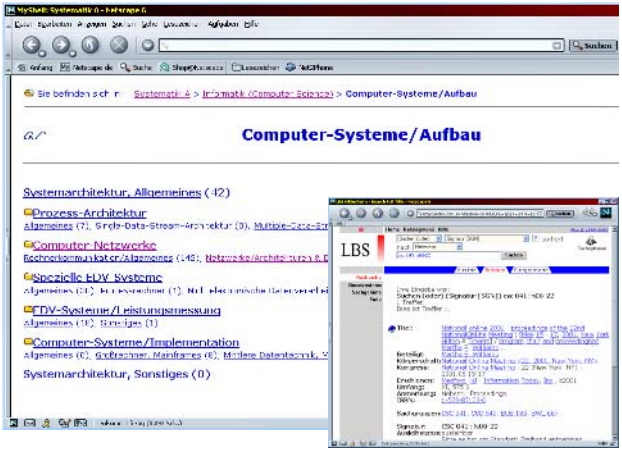

Fig. 3. MyShelf prototype 1 with link to internet library catalogue

The second prototype is mainly based on the same content as the first one. However, it is based on a relational database and applies PHP technology to display the objects assigned to the ontology concepts[2]. In addition to the library books, quality checked URLs have been included into the database. That allows the user to explore both relevant links and books within the same interface. The ontology switching applies for both links and books [20]. Most of the links have been taken from a previously constructed Clearinghouses for computational linguistics. In addition, links on information retrieval have also been identified and evaluated.

3.3 Semantic Web Prototype of MyShelf

The third prototype of MyShelf has been implemented in Java and relies on semantic web technology [9]. The ontologies have been converted to OWL. The data form the relational database contains also the assignments of the objects to the ontology concepts. This data has been converted to RDF (Resource Description Framework). Tools were implemented to parse the original data and transform it into an hierarchical document object model (DOM). After this step, RDF files could be written. As metadata scheme, Dublin Core was used.

The RDF data was stored in the Jena semantic web database[3] system which allows queries in semantic web query languages and stores the data internally in a relational database scheme.

The implementation with semantic web technology brings several advantages. The data in the system can now be easily complemented with data in the semantic web format from other sources. In the library contextt of MyShelf, we are currently accessing Amazon in order to add another taxonomy and additional information like prices and the bibliographic service DBLP[4] in order to display the content of volumes and proceedings which are available in the library.

2 http://www.vww-info.de/browse.php
3 http://jena.sourceforge.net/
4 http://www.sigmod.org/dblp/db

4 Evaluation and Results

The first prototype was evaluated by a user study, examining the following aspects:

- User behavior during the search process
- Overall usability issues
- Satisfaction of the users with the system
- Usefulness of the newly developed HANKE classification compared to the library classifications
- Acceptance of the parallel provision of three different classifications

Eleven subjects, all students of the course International Information Management at the University of Hildesheim, were recruited for the study. Detailed evaluation results can be found in [6]. The evaluation included an interview, several search tasks and a follow up questionnaire. The interview focused on the experience of test persons with library information systems. With this setup, objective and subjective measures for user satisfaction and interaction quality could be measured. The interview focused on the subjects' research experience and practices using the OPAC of the university library. It was found that the subjects usually searched the OPAC by entering a keyword. They were not used to search via browsing through classifications.

4.1 Search Tasks

Each subject was asked to perform eight search tasks after a brief introduction to the virtual library shelf. The subjects were encouraged to think aloud and comment on what they were doing. The subjects were also informed that they could spend as much time as they wanted on each task, and that they could abandon a search without a result in cases where they thought thy could not find a satisfying result.

There were two types of search tasks:

- The subjects were asked to find literature about a particular topic.
  Example: "Find publications about the programming language JAVA"
- The subjects were asked to locate a particular topic within a classification.
  Example: "Under which broader term can you find the topic multimedia within the classification?"

For the first seven tasks the subjects were told which classification to use for their search. This methodology guaranteed that each subject got to know and to use all three classifications. For task 8 the subjects were free to select the classification(s) of their choice. Thus, it could be determined which classification they preferred and whether they made use of only one classification or switched between them.

Search times varied widely and did not allow conclusions. It was more an indicator of time spent browsing or exploring than a performance measure. It should be noted, that a positive learning effect is involved in browsing during the search tasks. Comparing the task results with the defined optimal solutions was more helpful: The subjects' results of the search tasks 1-7 were checked whether they corresponded completely/partly/not at all with the optimal solutions, or whether the search was abandoned.

The subjects' search results indicated that the newly developed HANKE classification was most useful for the subjects. This finding was verified by the results of task 8 and the subjects' statements in the questionnaire.

As mentioned above, for task 8 the subjects were free to select one or more classifications. Table 1 shows which classification the subjects used, which classification was first choice, which was the only choice, and in which classification they found results.

Table 1. Which classifications the subjects used for task 8

Classifications	Used	First choice	Only choice	Results
A (Hildesheim)	6	1	0	3
B (HANKE)	10	9	4	8
C (KID)	5	1	1	2

The subjects preferred the newly developed classification (highlighted grey). The second question regarding task 8 was: did the subjects use only one classification, or did they switch? It turned out that six out of eleven subjects changed the classification at least once. There were two reasons for that behaviour:

1. Test users had found no results in one classification.
2. Test users changed the classification in order to look at the topic from a different angle.

4.2 Subjective Evaluation

A questionnaire focused on the question how the test persons subjectively evaluated the classifications in comparison and the virtual library shelf as a whole. Table 2 shows the subjects' evaluation of the three classifications.

The newly developed HANKE classification turned out to be the subjects' favourite, which confirmed the trend of the results obtained from analysing the tasks. Only with regard to structure (item 3) the HANKE classification was outperformed by the Hildesheim classification which users of the library already were familiar with.

In the questionnaire test users reported no problems in adjusting from one classification to another and did not lose their orientation in the virtual library shelf because of the large number of links.

The concluding question the subjects were asked to answer in the questionnaire was: Would you welcome the integration of the virtual library shelf into the university library OPAC? Ten out of eleven subjects asnwered yes. Four subjects supported the integration of all three classifications, six subjects the integration of one or two (the newly developed classification and the Hildesheim classification, respectively).

4.3 Interpretation

The goal of the design and evaluation of the virtual library shelf has been to examine the user search behaviour in a hypertext-based, browsing-oriented library system based on three hierarchical classifications. The results of the user study have showed

Table 2. The subjects' evaluation of the classifications

	Classification A (Hildesheim)	Classification B (HANKE)	Classification C (KID)	No difference between the classifications
Altogether, I along best with ...	3	**8**	0	0
The most clearly arranged classific-ation was	2	**7**	1	1
I could understand best the structure of	**6**	4	0	1
The names of the classes were best understandable in	3	**6**	1	1
The most useful classification for the searches was .	3	**8**	0	0

that although the users were only used to keyword search, they appreciated browsing features as well. Furthermore, they welcomed the provision of more than one classification and were able to deliberately choose between them. The HANKE classification has proved to be the most appropriate ontology for the typical search tasks for information science students. In the free search task, users actively applied ontology switching. Most users switched once and others twice. Test persons had no problems adjusting from one classification to another and did not lose their orientation in the virtual library shelf because of the large number of links. The subjects were also asked to answer whether they would welcome the integration of the virtual library shelf into the university library OPAC. All but one test person would welcome such an integration. The newly developed HANKE classification proved to be best, which indicates that the "age" of a classification can be an important factor, especially in the rapidly changing world of information science. It shows further that investment in the development of classifications improves interaction. Updates and adaptation of an ontology to a specific context makes a system easier to use for the users. In our context, adaptation of the information science classification for students increases the satisfaction with the system. However, old classifications should not be omitted but much rather integrated into a holistic system like MyShelf. In that manner, neither knowledge which users obtained with former classifications nor indexing work is lost. Offering parallel access to classifications is not only subjectively valued by users, but they also make use of the switching capabilities.

The evaluation results can be summarized in the following statements:

- The best ontology for this task has been identified
- Subjective and objective evidence shows that ontology switching is helpful

5 Outlook

Challenges for heterogeneous ontologies are also posed by multilingual systems. As one of the next steps, we in tend to include multilingual elements into the MyShelf concept. The problems and solutions for multilingual information retrieval make a good starting point for that endeavor [17].

For future projects, not only intellectually designed ontologies will be considered. Since ontology switching allows various viewpoints, we will create several automatically constructed ontologies for the domain. Tools for ontology learning construct these structures from full text documents [2, 11].

References

1. Böhm, K.; Heyer, G.; Quasthoff, U.; Wolff, C.: Topic Map Generation Using Text Mining. The Journal of Universal Computer Science J. UCS 8 (6) 2002. pp. 623-643
2. Colace, F.; De Santo, M.; Foggia, P.; Vento, M.: Ontology Learning Through Bayesian Networks. In: International Conference on Enterprise Information Systems (ICEIS) (2) 2003. pp. 430-433
3. Ding, Y.; Foo, S. Ontology research and development. Part I – a review of ontology generation. In: Journal of Information Science 28(2). 2002. pp. 123-136
4. Ehrig, M.; Sure, Y.: Ontology Mapping – An Integrated Approach. In: The Semantic Web: Research and Applications. First European Semantic Web Symposium (ESWS 2004) pp. 76-91.
5. Hanke, P.; Mandl, T.; Womser-Hacker, C.: Ein „Virtuelles Bibliotheksregal" für die Informationswissenschaft als Anwendungsfall semantischer Heterogenität. In: Information und Mobilität: Optimierung und Vermeidung von Mobilität durch Information. Proceedings 8. Intl. Symposium für Informationswissenschaft (ISI 2002) Regensburg. pp. 289-302.
6. Heinz, S; Mandl, T.; Womser-Hacker, C.: Implementation and Evaluation of a Virtual Library Shelf for Information Science Content. Proceedings of the fifth National Russian Research Conference (RCDL). St. Petersburg. 20.-31. Oct. 2003.. pp. 117-123.
7. Hellweg, H.; Krause, J.; Mandl, T.; Marx, J.; Müller, M.; Mutschke, P.; Strötgen, R.: Treatment of Semantic Heterogeneity in Information Retrieval. IZ-Arbeitsbericht Nr. 23, IZ Sozialwissenschaften, Bonn. 2001. http://www.gesis.org/Publikationen/Berichte/IZ_Arbeitsberichte/index.htm#ab23
8. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In European Conference on Machine Learning (ECML) 1998. pp. 137-142.
9. Kölle, R.; Mandl, T.; Schneider, R.; Strötgen, R. (2004): Weiterentwicklung des virtuellen Bibliotheksregal MyShelf mit semantic web Technologie. In: Information Professional 2011: Strategien – Allianzen – Netzwerke. Proceedings 26. DGI Online-Tagung. Frankfurt a.M. 15.-17. June 2004. pp. 147-153.
10. Kotis, K.; Vouros, G. : The HCONE Approach to Ontology Merging. In: The Semantic Web: Research and Applications. First European Semantic Web Symposium (ESWS 2004) pp. 137-151.
11. Maedche, A.; Motik, B.; Stojanovic, L.; Studer, R.; Volz, R.: Managing Multiple Ontologies and Ontology Evolution in Ontologging. In: Intelligent Information Processing 2002. pp. 51-63

12. Mandl, T.: Tolerant Information Retrieval with Backpropagation Networks. In: Neural Computing & Applications. Special Issue on Neural Computing in Human-Computer Interaction vol. 9 (4) 2000. pp. 280-289.
13. Mandl, T.; Womser-Hacker, C.: Fusion Approaches for Mappings Between Heterogeneous Ontologies. In: Research and Advanced Technology for Digital Libraries: 5th European Conference (ECDL 2001) Darmstadt Sept. 4.-8. [LNCS 2163] pp. 83-94.
14. Mandl, T.; Womser-Hacker, C.: Ontology Switching as Interaction Technique for the Semantic Web. In: Universal Access in HCI: Inclusive Design in the Information Society. Proceedings of the 2nd International Conference on Universal Access in Human-Computer Interaction (UAHCI), Crete, 22-27 June 2003. pp. 567-571
15. Navarro-Prieto, R.; Mike S., Yvonne R.: Cognitive Strategies in Web Searching. In: 5th Conference on Human Factors & the Web. June 3. Gaithersburg Maryland. 1999 http://zing.ncsl.nist.gov/hfweb/proceedings/navarro-prieto/index.html
16. Noy N., & Musen, M.: SMART: Automated Support for Ontology Merging and Alignment. 12th Workshop on Knowledge Acquisition, Modeling and Management. Banff, Canada. Oct. 1999 http://sern.ucalgary.ca/KSI/KAW/KAW99/papers/Fridman1/-NoyMusen.pdf
17. Peters, C.; Gonzalo, J.; Braschler, M.; Kluck, M. (Eds.): Comparative Evaluation of Multilingual Information Access Systems, 4th Workshop of the Cross-Language Evaluation Forum, CLEF 2003, Trondheim, Norway, August 21-22, Revised Selected Papers. LNCS 3237
18. Rahm, E.; Bernstein, P.: A Survey of Approaches to Automatic Schema Matching. In: The VLDB Journal 10. 2001. pp. 334-350.
19. Walton, M.; Vukovic, V.: Cultures, literacy, and the web: dimensions of information "scent". In: interactions 10 (2) special issue: HCI in the developing world. 2003. pp. 64-71
20. Wilhelm, B.: Der virtuelle Wegweiser Informationswissenschaft: Entwicklung und Implementierung eines Konzepts für die Integration eines Clearinghouse in das virtuelle Bibliotheksregal MyShelf. Master Thesis, Information Science, Univ. of Hildesheim. 2004
21. Yeo, A.; Loo, W.: Identification and Evaluation of Classification Schemes: A User Centred Approach. In: Designing for Global Markets 6: Proceedings of the Sixth International Workshop on Internationalization of Products and Systems (IWIPS 2004) 8-10 July, Vancouver. pp. 75-87.

A Meta-Model for Intellectual Capital Reporting

Martin Nemetz

University of Vienna, Faculty for Computer Science, Department of Knowledge and Business Engineering, Brünner Strasse 72, A-1210 Vienna, Austria
martin.nemetz@dke.univie.ac.at

Abstract. Since more than a decade, various conceptions of intellectual capital reporting have been presented in both the academic and practitioner's literature. As the evolving concepts of reporting can most certainly not be compared with each other, the real value of those reports cannot be unfolded by investors and stakeholders in general. When looking at the discrepancy between an enterprise's market value and book value, those reports seem to gain even more in importance. The application of the meta-modelling approach allows the generation of intellectual capital reports by using so called meta-indicators. Those generated reports are based on already existing intellectual capital reports, which leads in turn to the possibility of comparing two originally different intellectual capital reports.

1 Introduction

Since the 1990s, both the academic and the practical discussions about intellectual capital and its influence on the enterprises' performances have led to a high amount of diverse concepts and procedures for managing and also measuring an organization's intellectual capital (see e.g. [1], [2] [3], [4]). As a result, the upcoming conceptions that have been presented on various conferences and workshops are not interchangable. Moreover, the outcomes of the measurement of intellectual capital that are typically presented in the form of a report or a statement are not comparable at all as the underlying conceptions differ dramatically. This raises the following question: Does it make sense to report intellectual capital as investors and stakeholders in general cannot evaluate the positions of two competitors within the same branch based on the intellectual capital and its reports? Additonally, when looking on the obvious importance of the non-financial capital for an enterprise's success, it seems to be really important trying to find a solution for this dilemma.

The first step in the direction of the defintion of intellectual capital is its accepted structure. Hence, intellectual capital is understood as a set of three factors that are human capital, structural capital, and relational (or customer) capital [17]. Therefore, the next logical step would concern a unification and standardization of available intellectual capital approaches [5]. But as it cannot be expected that an agreement on a standardized intellectual capital reporting can be found in the near future, another approach can at least contribute a small

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 213–223, 2006.

step into the right direction. Instead of attempting to standardize an intellectual capital report, it may be more realistic to find similarities of the existing intellectual capital reports and statements and bundle them for deriving comparable meta-indicators that can be used for the comparison of indicators of different firms that rely on diverse conceptions of intellectual capital reports. A very promising approach hereby is the meta-modelling method.

The paper is organized as follows: Section 2 will describe the Intellectual Capital Reporting Framework with its two components, the Intellectual Capital Reporting Step Model and the Intellectual Capital Reporting Meta-Models, whereas the focus of this paper rests on the latter. Therefore, general concepts of the meta-modelling approach will be presented as well as the application of those concepts for the creation of meta-models for intellectual capital reporting. Section 3 will give an outlook of further tasks to do, before the conclusion will complete the paper.

2 The Intellectual Capital Reporting Framework

The reporting of intellectual capital is gaining steadily more on importance as the overall discrepancy between an enterprise's market value compared to its book value is getting bigger. When looking on the S&P 500 Companies and their ratio of market-to-book value, it can be determined that the ratio has reached in March 2001 more than five times the value of December 1977 [15]. This huge portion of value of an enterprise that cannot be seen on the traditional financial balance sheet implies a lot of risks [16]. Due to the variety of intellectual capital reports that are available, a comparison of values stated in these reports can most likely not be executed.

The here presented Intellectual Capital Reporting Framework, containing the Intellectual Capital Reporting Step Model and the Intellectual Capital Reporting Meta-Models may support a small contribution for solving the problem of the non-comparability of existing intellectual capital reporting conceptions.

2.1 Intellectual Capital Reporting Step Model

Fig. 1 illustrates the Intellectual Capital Reporting Step Model and identifies six core steps [18]. The Intellectual Capital Reporting Step Model starts with the **elaboration of the strategy and tactical targets**, wherein the firm's global strategy is defined as an intellectual capital report depends also on the company's strategic and tactical objectives. Further, step 2, the **definition of the business processes and working environments** has to be executed as the firm's business processes have to be modelled for being able to step forward to the **extraction of intellectual capital-relevant business processes and working environments**. Here, those business processes that contain intellectual capital-intensive activities have to be identified, before in step 4, the **intellectual capital processes are defined** and subsequently in step 5, the **selection of applicable intellectual capital reporting concepts** can be

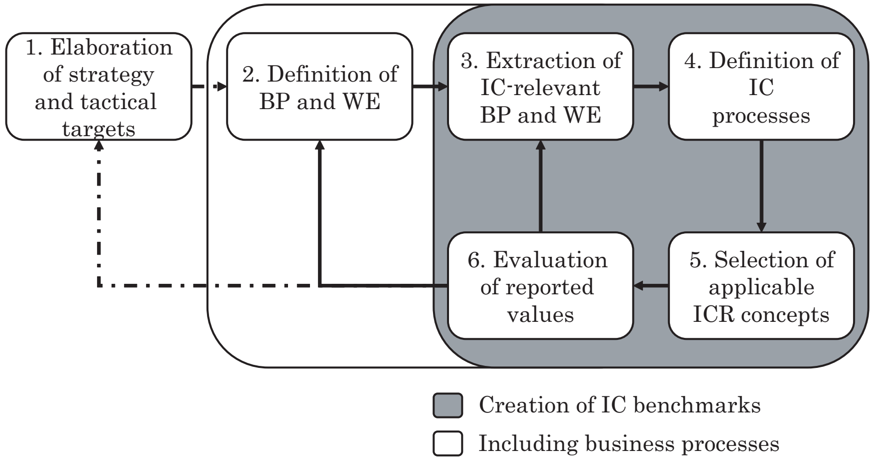

Fig. 1. Intellectual Capital Reporting Step Model

chosen. Finally, step 6 contains the **evaluation of reported values**, where it can be checked whether the company's performance is satisfactory according to its prior-set strategy [18].

2.2 Intellectual Capital Reporting Meta-Models

The conception of meta-modelling has already gained wide acceptance in diverse business areas and has also been applied in various projects (see e.g. [12], [13], [14]). In the following, the meta-modelling approach shall be presented briefly.

The Meta-Modelling Approach. The classical architecture of the meta-modelling approach is a four layered construct. Fig. 2 depicts this architecture [6], [7]. The four layer construct of meta-modelling has been established in the course of the introduction of the meta-objects facility (MOF^{TM}). It enables the definition and construction of meta-models [8].

The hereby presented infrastructure contains a hierarchy of model levels, whereas each represents an instance of the above level (with the exception of the top level that is not instantiated). Those four levels are characterized as follows (see e.g. [6], [7], [8], [9]):

- M_0: This is the bottom level that contains data, i.e. data objects that can be manipulated by the software and/or the user.
- M_1: The model level includes models of the data of level M_0.
- M_2: This layer represents the meta-model level that contains any kinds of meta-models.
- M_3: The fourth level is the meta-meta-model layer (also called meta^{2}model layer) that holds information of one or more meta-models of M_2.

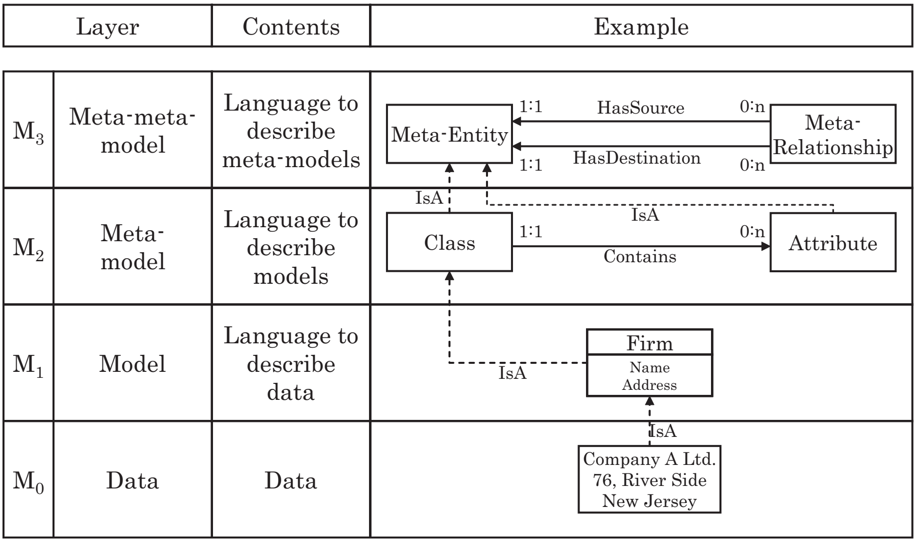

Fig. 2. Meta-Modelling Levels

Certainly, it would be possible to add other meta-layers (i.e. for instance a meta-meta-meta-model level M_4) that would describe underlying layers, but the majority of frameworks limit their numbers of layers to four [7].

When referring to fig. 2, the column "contents" contains short descriptions of languages. These (meta-)modelling languages are required for the creation of models, meta-models, and meta-meta-models, respectively [10]. Fig. 3 illustrates the interdependencies of the modelling languages and the appropriate models [11]. For the creation of a model, as a picture and/or description of an artefact of the real world, a modelling language is used. This modelling language is defined in the meta-model, which is again a model of the model of level 1, as it represents a model of the applied modelling language. Thus, the meta-model is formalized in a language, the meta-modelling language, which is defined in level 3, the meta-meta-model that is in turn created with the aid of a meta-meta-modelling language [11] [10]. This could be extended to the meta-meta-meta-model and its language, but as the meta-modelling framework contains, like afore-defined, only four layers, this shall not be elaborated anymore.

Furthermore, a modelling language is composed of a syntax, semantics, and a notation [10].

– **Syntax:** It describes elements and rules that are required for the creation of models. The syntax is described by a grammar.
– **Semantics:** Hereby, it is described the meaning of the modelling language, whereas it is composed of two separate parts:
  • **Semantic domain:** It expresses the meaning of the semantics of the modelling language by using ontologies, mathematical expressions, or equivalent expression forms.

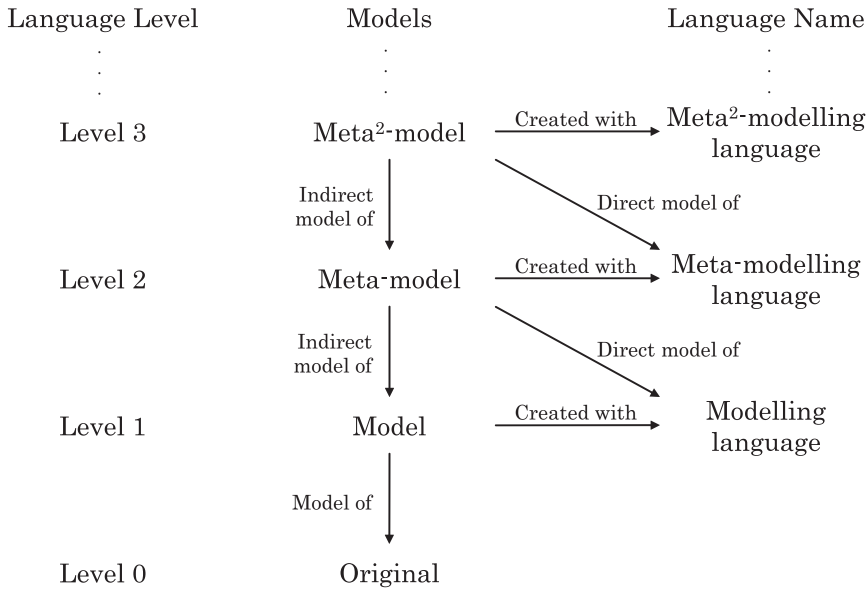

Fig. 3. (Meta-)Modelling Languages

- **Semantic mapping:** This represents the connection between the modelling language's syntax and the semantic expression, i.e. the semantic domain.

– **Notation:** Hereby, it is understood the visual representation of the modelling language.

Next to the modelling language with its syntax, semantics, and notation, the modelling procedure describes the steps that are applied on the modelling language for achieving results, as e.g. models [10].

A modelling method consists of the modelling language, the modelling procedure, as well as of mechanisms and algorithms (abbreviated M&A). The M&A are divided into three main categories [10]:

– **Generic mechanisms and algorithms:** They are implemented in a meta-meta-model, i.e. they are available for all meta-models that are instances of the appropriate meta-meta-model. Examples would be import/export functionalities that should be available for all meta-models.
– **Specific mechanisms and algorithms:** This category of M&A is implemented in a meta-model, therefore those M&A are available only for a specific meta-model. Potential applications are specific calculation algorithms that are required for a certain meta-model.
– **Hybrid mechanisms and algorithms:** The hybrid M&A are implemented in a meta-meta-model, but in contrary to the generic M&A, they are adapted especially for a certain meta-model. An example for a hybrid M&A may be an export component that is valid only for one certain meta-model.

Fig. 4 summarizes the components of a modelling concept graphically and shows the interdependencies between the modelling language, the modelling procedure and the M&A [10].

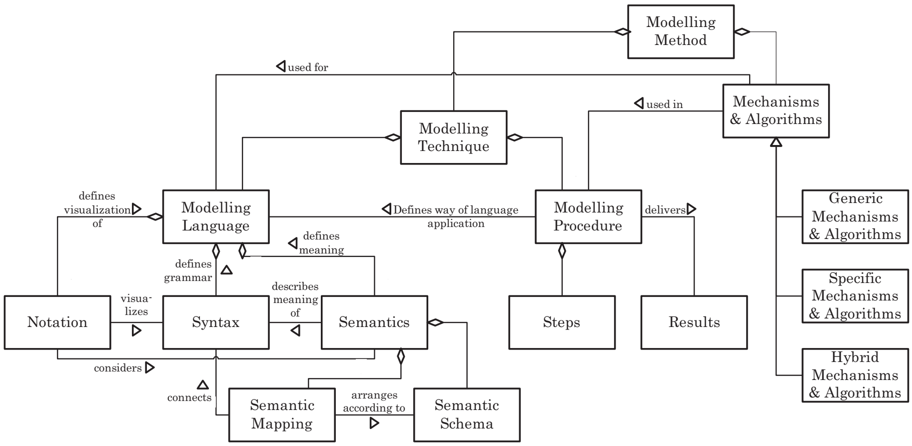

Fig. 4. Components of a Modelling Method

Meta-Models for Intellectual Capital Reporting. As has already been described before in the discussion about the Intellectual Capital Reporting Step Model, the firm's business processes play a major role when defining a meta-model for an intellectual capital report. Therefore, they have to be included in the considerations concerning a meta-model for intellectual capital reporting. In the following, fig. 5 gives an overall picture of the whole modelling method containing seven meta-models for intellectual capital reporting, one meta-model for business processes and one meta-model for working environments.

The seven intellectual capital reporting meta-models are again classified into two categories: Primary model types and secondary model types, whereas the first contain the human factor meta-model (HFM), the structural factor meta-model (SFM), and the relational factor meta-model (RFM). The secondary model types are composed of the risk meta-model (RM), the cost meta-model (CM), the competence pattern meta-model (CPM), and the patent map meta-model (PM). The primary model types contain information that is directly related to and can be found in intellectual capital reports, whereas the secondary model types are composed of data that is relevant for the linking and/or calculation of values relevant for intellectual capital reports. Thus, they represent the supporting means for the values to be included in the primary model types. As there exist a lot of interdependencies between the classes of all model types, fig. 6 provides an overview of the interdependencies between them. Furthermore, table 1 provides an overview of the meta-models with their classes as well as short descriptions of their purposes.

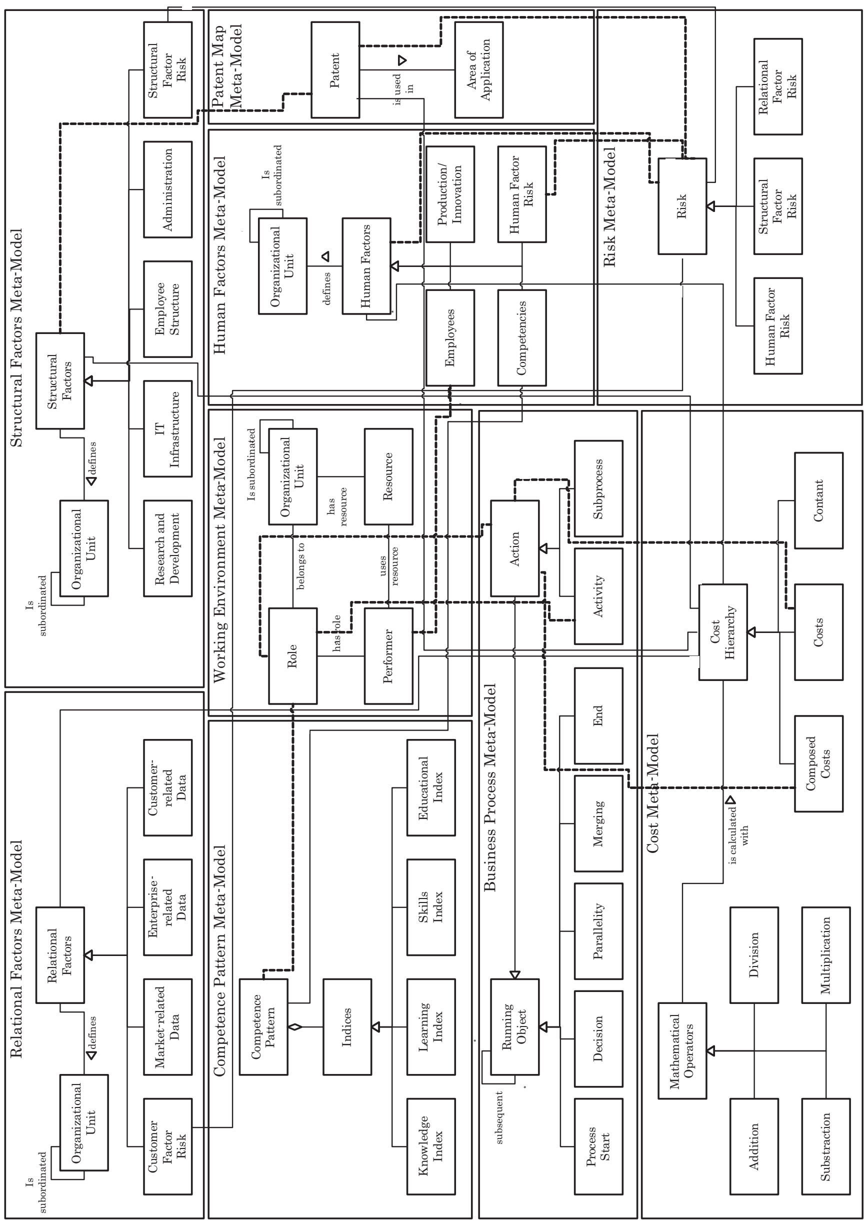

Fig. 5. Meta-Models for Intellectual Capital Reporting

Technical Realization. The meta-models have been implemented in the meta-modelling platform ADONIS® with the aid of the languages ALL and AdoScript [12]. ADONIS® applies the afore-described four layer meta-modelling approach. The meta-meta-model is already defined in ADONIS®, whereas the meta-model

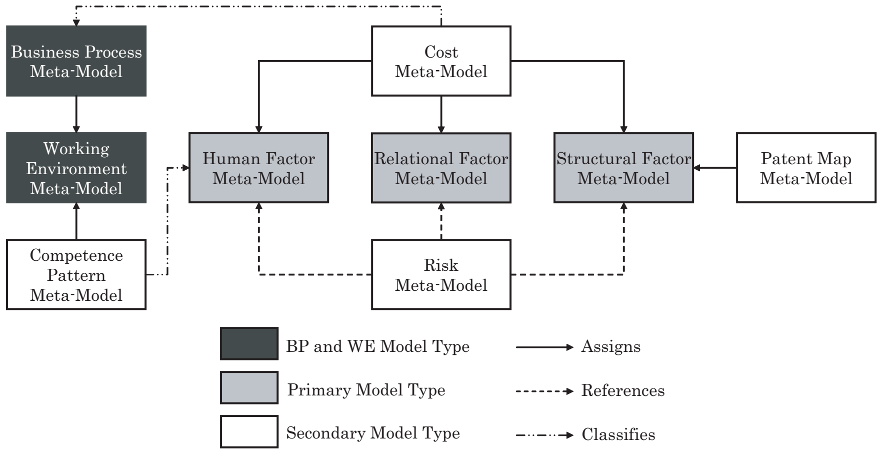

Fig. 6. Interdependencies Between the Different Meta-Models for Intellectual Capital Reporting

has been implemented for being able to create models of intellectual capital reports. Exemplarily, the following programming codes show small excerpts of the syntax as well as the notation of the human factor meta-model:

Example 1. Syntax Definition

```
NOTEBOOK
CHAPTER "Description"
ATTR "Name"
ATTR "Performer" dialog actor lines: 3
ATTR "Reference Costs"
CHAPTER "Numbers" GROUP "Number/Share of Employees"
ATTR "Number of Employees"
...
```

Example 2. Notation Definition

```
GRAPHREP
ATTR "Name" y:1.5cm w:c:2.8cm h:t
FILL color:white
ELLIPSE
x:0cm y:-.25cm rx:.3cm ry:.3cm
FONT bold color:gold h:14pt
TEXT "M" x:-.2cm y:-.55cm FONT style:times new roman
...
```

The classes of the respective model types (as e.g. the human factor model) contain attributes that are treated as meta-indicators or benchmarks [18]. Based on these meta-indicators together with the meta-modelling structure of the

Table 1. Meta-Models and Their Classes

Type	Class	Class Description
HFM	Organizational Unit	Information of the appropriate organizational unit
	Employee	Numbers, changes and types of employed people
	Competences	Reference to appropriate competence pattern model
	Productivity/ Innovation	Information on the added value per employee and innovation within the organizational unit
	Human Factor Risk	References to the risk and the cost models
SFM	Organizational Unit	Information of the appropriate organizational unit
	R&D	Costs and resources for R&D and its administrative effort
	IT Infrastructure	Describes the effort for maintenance and usage of the firm's IT infrastructure
	Employee Structure	Information of the turnover and % of teleworkers
	Administration	Number of projects and their applications as well as the overall administrative effort
	Structural F. Risk	References to the risk and the cost models
RFM	Organizational Unit	Information of the appropriate organizational unit
	Market-related Data	Quantitative and qualitative factors of the market
	Enterprise-rel. Data	Number of orders and customer contacts
	Customer-related Data	Number of acquired, existing, and lost customers as well as welfare by customer
	Relational F. Risk	References to the risk and the cost models
CPM	Competence Pattern	Serves as a variable containing the information of the competence pattern that is referenced to the HFM
	Knowledge Index	Defines the required specialized knowledge for a certain competence pattern
	Learning Index	Information on the to-pass training days per year for a certain competence pattern
	Skills Index	Contains information of required skills for a certain competence pattern
	Educational Index	Defines the required state of education for a certain competence pattern
CM	Costs	Defines the height of the costs and to which primary model type they are assigned to
	Mathematical Operators	Distinction between addition, subtraction, multiplication, and division
	Constant	Mathematical constant
	Composed Costs	A composition of several cost classes, constants, and mathematical operators
RM	Risks	Classes for human, structural, and relational factor risks that are assigned to the human, structural, and relational factor models
PM	Patent	Contains the patent's name, number, kind of protection as well as the inventors
	Area of Application	Name of relevant project, application, or the like

intellectual capital reporting models, it is now possible to compare diverse intellectual capital reports of different enterprises that use different intellectual capital reporting concepts. The key hereby is the meta-modelling structure of the intellectual capital reports that allows the assignment of meta-indicators to various (derivable) intellectual capital reporting concepts. The following figure depicts the assignment of meta-indicators that are extracted from existing intellectual capital reports (level 1) by entering the values of the meta-indicators into the classes of the meta-models of an intellectual capital report (level 2) for deriving intellectual capital reports (level 3). The generation of those reports should be handled automatically by using a so-called query component. The query component represents a control centre that allows either the selection of pre-defined intellectual capital reports and thus also the relevant indicators, or the manual selection of meta-indicators. In both cases, the result would be an automatically generated report containing the respective indicators.

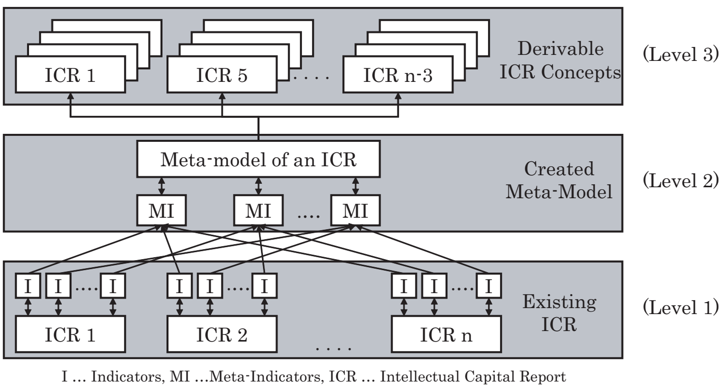

Fig. 7. Deriving Intellectual Capital Reporting by the Usage of Meta-Models

3 Conclusion and Outlook

The paper presented an approach that allows the comparison of existing intellectual capital reports. Therefore, it made use of the meta-modelling concept, with which it is possible to take the indicators of existing intellectual capital and eventually transfer them into the meta-model level. This metaization of indicators to meta-indicators allows a comparison of diverse intellectual capital reporting concepts. The technical realization of the meta-models for intellectual capital reporting has been executed with the aid of the meta-modelling platform ADONIS®, using ALL and AdoScript. The next steps include the programming of the query component that would allow the automatic generation of intellectual capital reports from the meta-models and its classes, as well as an extensive testing phase, and the extension of the current meta-model to new upcoming streams of design of intellectual capital reporting concepts.

References

1. Roos, J.: Exploring the Concept of Intellectual Capital (IC). Long Range Planning **31** (1998) 150–153
2. Stewart, T. A.: Intellectual Capital. Nicholas Brealey Publishing. London (1997)
3. Sullivan, P. H.: Value-driven Intellectual Capital. John Wiley and Sons. New York (2000)
4. Lev, B.: What Then Must We Do?. in Hand, J. R. M., Lev, B. (eds.): Intangible Assets: Values, Measures, and Risks. Oxford University Press. New York (2003)
5. Marr, B., Chatzkel, J.: Intellectual Capital at the Crossroads: Managing, Measuring, and Reporting of IC. Journal of Intellectual Capital. **5** (2004) 224–229
6. Bézevin, J., Lemesle, R.: Ontology-based Layered Semantics for Precise OA&D Modeling. in Bosch, J., Mitchell S. (eds.): OO-Technology - ECOOP'97 Workshop. (1997) 151–155
7. Atkinson, C.: Meta-Modeling for Distributed Object Environments. EDOC'97. Proceedings, First International Volume **24-26** (1997) 90–101
8. Bézevin, J., Gerbé, O.: Towards a Precise Definition of the OMG/MDA Framework. ASE'01. Proceedings of the 16th Annual International Conference on Automated Software Engineering (2001).
9. Atkinson C., Kühne, T.: Model-Driven Development: A Metamodeling Foundation. IEEE Software **20** (2003) 36–41
10. Karagiannis, D., Kühn, H.: Metamodelling Platforms. in Bauknecht, K., Min Tjoa, A., Quirchmayer G. (eds.): Proceedings of the Third International Conference EC-Web - DEXA 2002. Springer Verlag. Berlin (2002)
11. Strahringer, S.: Metamodellierung als Instrument des Methodenvergleichs. Shaker Verlag. Aachen (1996)
12. Junginger, S., Karagiannis, D., Kühn, H., Strobl, R.: Ein Geschäftsprozessmanagement-Werkzeug der nächsten Generation - ADONIS®: Konzeption und Anwendungen. Wirtschaftsinformatik **42** (2000) 392–401
13. Karagiannis, D., Bajnai J.: eduXX - The Instructional Design Platform. in Electronic Proceedings of EISTA'04 - International Conference on Education and Information Systems: Technologies and Applications. Florida (2004)
14. BOC: The Meta-Modelling Concept. http://www.boc-eu.com (last access on July 24, 2006)
15. Lev, B.: Intangibles. Brookings Institution Press. Washington D.C. (2001)
16. Karagiannis, D., Nemetz, M., Schwab, M.: Dashboards for Monitoring Compliance to Regulations – A SOX-based Scenario. in Electronic Proceedings of IGO'06 - International Conference on Integrating Global Organizations. Italy (2006)
17. Saint-Onge, H.: Tacit Knowledge. Strategy & Leadership **24** (1996) 10–14
18. Nemetz, M.: Towards a Model for Creating Comparable Intellectual Capital Reports. in Maurer, H., Tochtermann, K. (eds.): Proceedings of I-Know'06 - International Conference on Knowledge Management. JUCS. Graz (2006)

Assessment of Effective Utilization of KM Technologies as a Function of Organizational Culture

Heejun Park[1] and Duke H. Jeong[2]

[1] Department of Information and Industrial Engineering, Yonsei University,
134 Shinchon-Dong, Seodaemoon-Gu, Seoul 129-749, Rep. of Korea
h.park@yonsei.ac.kr
[2] Department of Information Management, Dongguk University,
26 Peel-Dong, Joong-Gu, Seoul 150-103, Rep. of Korea
duke@dongguk.edu

Abstract. Companies are adopting more technologies than ever to maximize the benefit of Knowledge Management (KM). However, recent global analyses of such investments highlight the fact that not all of them are necessarily successful. Too much emphasis on technology without incorporating organizational culture can easily result in failed implementation of KM technology. The key factors for the higher return on the KM technology are the choosing the right technologies for given organizational culture and the effective utilization of those technologies. The purpose of this research is to explore possible relationships between the KM technology types and organizational culture orientations regarding effective utilization of KM technology. Data used to test hypotheses derived for this research were obtained from 294 responders from the Organizational Culture Profile (OCP) survey instruments and 143 responders from the Knowledge Management Technology Profile (KMTP) survey instruments representing 29 separate organizations. The OCP provides a profile of an organization's culture orientation while the KMTP provides a profile of the organization's effective utilization of KM technologies. The results of this research suggest that organizations utilizing collaborative technologies effectively identified people-oriented culture orientation. On the other hand, effective utilization of distributive technologies does not show any relationship with organizational culture orientation.

Keywords: Organizational Culture Orientation, Knowledge Management Technology Taxonomy, Affective Unitization of Knowledge Management Technology.

1 Introduction

The Knowledge Management (KM) technology market has evolved as many organizations have been implementing KM programs, and literally dozens of products and portal solutions deliver the major functions that KM systems require [3]. Many companies implement KM technology with a half-cooked approach by convincing

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 224 – 233, 2006.

themselves, that buying a million dollars' worth of neat technologies will give them a measurable return on investment. However, not all of them are necessarily successful because some of them do not understand all the business practices and cultural and organizational changes they will have to make along with [5]. Too much emphasis on technology without incorporating the other critical elements (i.e., business strategy under leadership, and organizational structure and culture) can easily result in failed implementations of KM program [4]. The same dollar spent on the same system may give a competitive advantage to one company but only expensive paperweights to another. The key factors for the higher return on the KM technology dollar are the choosing the right technologies for given business contexts and the effective utilization of knowledge using those technologies [5].

This research will examine the relationships between organizational culture and KM technology type regarding effective utilization of knowledge using KM technologies. The following sections will examine the literature on Knowledge Management Technology and organizational culture. Then, the rational and methodology for this study will be discussed. Finally, the findings from the surveys conducted for this study will be described with the data analysis and conclusions and recommendations for the future study will be also presented.

2 Knowledge Management Technologies

The creation, retaining and sharing of knowledge within and among different knowledge communities require the coordinated management of tacit and explicit knowledge [2]. Knowledge Management (KM) technologies provide a seamless pipeline for the flow of explicit and tacit knowledge through four modes of knowledge conversion which are socialization, externalization, combination, and internalization. The KM technologies enable capturing knowledge, defining, storing, categorizing, indexing and linking digital objects corresponding to knowledge units, searching for and subscribing to relevant content, and presenting content with sufficient flexibility to render it meaningful and applicable across multiple contexts of use [7].

Processing knowledge can be segmented into two broad classes: distributive and collaborative process, each addressing different knowledge management objectives. Together, these approaches provide a broad set of knowledge processing capabilities. They support well-structured repositories for managing explicit knowledge while enabling interaction to integrate tacit knowledge [10]. Technologies used in distributive process exhibit a sequential flow of explicit knowledge into and out of the repository while technologies used in collaborative process are focused primary on supporting interaction among people holding tacit knowledge [10]. Distributive technologies maintain a repository of explicitly encoded knowledge created and managed for sequent distribution to knowledge consumers within or outside the organization. These technologies provide flexible access and view of the knowledge in a central repository. Knowledge producers and consumers interact with the repository rather than with each other directly [10]. Collaborative technologies may be a simple directory of individuals within or associated with a community of knowledge. It may also take the more interactive form of a knowledge brokerage, an electronic conference or discussion space where people may either search for knowledge by posing questions (e.g., "Does anyone know?") or advertise their expertise. The most collaborative form supports direct communication through discussion databases, computer conferences, and real-time collaboration technologies. These technologies directly support interaction and collaboration within and among knowledge-based teams, enabling "teams of teams" to form across knowledge communities [10]. The technologies supporting the main functions of Knowledge Management systems (KMS) are well suited to Zack's classification of Knowledge Management technologies (see Table 1.) [5].

Table 1. Classification of KM Technologies

Distributive(Integrative) Technology	Data Warehousing (and Data Mining tools) Database Management technologies Document Management Systems Electronic publishing Information Retrieval Systems Search Engines Intelligent Agents Enterprise Information Portal (i.e. Corporate Intranet) Decision Support Systems Business Modeling Systems
Collaborative(Interactive) Technology	Messaging or E-mail GroupWare Knowledge-mapping tools Enterprise Information Portal (i.e. Corporate Intranet) Web-based Training Help-desk applications Decision Support Systems Workflow Systems

3 Organizational Culture

There are some fundamental areas of agreement in the definition and the important role of culture in an organization, but less agreement exists about its measurement. To investigate person-culture fit, O'Reilly, Chatman, and Caldwell developed a survey instrument called the Organizational Culture Profile (OCP) [8]. This survey instrument contains a set of attribute statements that can be used to assess idiographically the extent to which certain attributes characterize a target organization. In a set of related investigations using multiple sets of respondents, researchers explored the characteristics of the OCP and demonstrated its ability to assess preferences for organizational attributes. The OCP contains 54 attribute statements that can generically capture individual and organizational attributes [8]. In order to define organizational culture orientations in this research, 44 out of 54 OCP attribute statements were mapped against two concerns of the "Managerial Grid" developed by Blake and Mouton, concern for people and concern for production. 10 out of original 54 OCP attribute statements were excluded because they do not fit any of two concerns of the Managerial Grid [9].

Each of two concerns on Managerial Grid was viewed as an organizational culture orientation which is a set of cultural attributions for this research. The original use of the Managerial Grid is to analyze interactions between significant variables of management – production and people – as consultant to understand a basic conflict in a top management group. The two dimensions of the Managerial Grid include Concern for Production and Concern for the People [1]. In each case, the term "concern for" is not addressing so much with the degree to which the needs of employees and their accomplishments are being considered, but rather the degree of interest that is presented and demonstrated by the organization's management. What is significant is how management concerns itself about production and people and how they interact. In their development of the Managerial Grid, Blake and Mouton defined these dimensions as follows [1]:

- Concern for production. The word "production" and "people" cover a range of considerations. Attitudes of concern toward production, for instance, may be seen in the type of policy decisions, the extent of creativity throughout the organization, procedures or processes, workload and efficiency demands, the quality of services, and the volume of output. The important aspect is that the meaning of production covers whatever the organization deems it important that people accomplish. At the lower level, concern for production may take the form of the number of things that can be counted, or the time it takes to meet production schedule. But at the organizational level, it may be demonstrated in the kind of policies established, the character of direction given to major programmatic efforts, or the importance applied to finding new directions or products to sustain the organization [1].
- Concern for People. In a similar fashion, concern for people can be expressed in a number of different ways. Included might be the degree of concern for personal commitment, accountability, trust versus obedience, self-esteem, good working conditions, benefit packages, security, and social relations or friendships with associates [1].

4 Research Questions and Hypotheses

Research in the field of Knowledge Management (KM) reveals that companies are adopting more KM technologies than ever to maximize the benefit of KM, but they don't take full advantage of them. Is successful implementation of knowledge management not just a combination of new technology, but also organizational culture? If so, which cultural attributes and/or orientations do have positive or negative correlation with effective utilization of a specific kind of KM technology? In determining the relationships between the KM technology type and organizational culture orientation regarding effective utilization of KM technology, two basic hypotheses will be developed and tested.

Hypothesis I

H0: There is no positive correlation between the effective utilization of collaborative technology and people-oriented cultural orientation for that organization.

H1: There is a positive correlation between the effective utilization of collaborative technology and people-oriented cultural orientation for that organization.

Hypothesis II
H0: There is no positive correlation between the effective utilization of distributive technology and production-oriented cultural orientation for that organization.
H1: There is a positive correlation between the effective utilization of distributive technology and production-oriented cultural orientation for that organization.

5 Research Methodology

5.1 Survey Instruments

For this research, the Knowledge Management Technology Profile (KMTP) was developed to assess the effective utilization of knowledge management technology by modifying slightly the Information Technology Investment Performance (ITIP) survey instrument developed by National Research Council [6]. The ITIP was developed to assess and understand patterns of behavior that could help explain why some organizations were, or were not, realizing greater payoffs from the investment in information technology [6]. Methods of determining effective utilization of knowledge management technology were researched and it was decided to use a modification to the ITIP survey instrument. The KMTP represents how effectively each of KM technology types is utilized for the successful implementation of KM program in that organization. On the other hand, The 44 statements of Organizational Culture Profile (OCP) grouped by two concerns of the Managerial Grid (see Table 2.) were used to identify cultural attributes and orientation which most likely describe a given organization [9].

Table 2. 44 Attributes of the OCP

Production-oriented Attributes	People-oriented Attributes
Adaptability	Being calm
Attention to detail	Being different from others
Autonomy	Being easy going
Being aggressive	Being thoughtful
Being carefulness	Confronting conflict directly
Being competitive	Decisiveness
Being exact	Demanding of employees
Being innovative	Developing friends at work
Being result-oriented	Enthusiasm for the job
Compliance	Fairness
Experimentation	Fitting in at work
Flexibility	Having a good reputation
High expectations for performance	Low level of conflict encouraged
Informality	Praise for good performance
Predictability	Respect for the individual’s right

Table 2. *(continued)*

Problem solving	Security of employment
Risk taking	Socially responsible
Being rule-orientated	Supportiveness of employees
Sharing information freely	Being team-oriented
Stability	Tolerance of failure
Taking advantage of opportunity	Trust
Taking initiative	Working closely with others

For the purpose of this research, reliability is not a leading concern because of the changing nature of both organizational culture and the way knowledge management technology is utilized across an organization. This research presents only a snapshot of the organization under study and the employees' feeling and perceptions about organizational culture and the implementation of knowledge management technology. An organization is a dynamic entity; conditions surrounding the operation of the business are constantly changing and thus the results from a reliability test of instrument would be expected to vary in reflection of those changing conditions [1]. The questionnaire Organizational Culture Profile (OCP) and the Information Technology Investment Performance (ITIP), slightly modified for this research, have been validated by many researchers in their previous researches.

5.2 Sample

The purpose of this research has been to determine the correlation, if any, between organizational culture attributes and the successful implementation of knowledge management technology. Data used to test the two hypotheses derived for this research were obtained from 294 respondents from the Organizational Culture Profile (OCP) survey instruments and 143 respondents from the Knowledge Management Technology Profile (KMTP) survey instruments representing 29 separate organizations. A total of 1088 OCP survey instruments and 215 KMTP survey

Table 3. Industry Type Patition

Industry Type	Number of Organizations	Percentage	Average Sample Size of KMTP	Average Sample Size of OCP
Consulting	8	30.8%	2.5	10.1
Software Development	6	19.2%	3.0	7.2
Financial/Baking/Accounting	5	19.2%	2.4	9.2
Manufacturing	3	3.9%	3.0	8.0
IT/Telecommunication	4	15.4%	2.3	8.0
Government	2	7.7%	3.0	10.0
Education	1	3.9%	2.0	4.0
Total	29	100.01%		
Average			2.6	8.7

instruments were distributed across 29 organizations. The OCP survey instruments were distributed to employees within the organization regardless of employees' function and level. The KMTP survey instruments were distributed to managers who were in a position to be knowledgeable about knowledge management technology across the organizations. A total of 294 OCP survey instruments were completed and returned from 29 organizations with the response rate of 27.0 percent. A total of 143 KMTP survey instruments were completed and returned from 29 organizations with the response rate of 66.5 percent. Table 3. summarizes industry type partition of 29 participating organizations.

6 Findings

6.1 Data Analysis

The Pearson Product-Moment correlation coefficient was considered as a method of determining linear relationship between two quantitative variables measured in interval scales – organizational culture and the successful implementation of a specific type of knowledge management technology. However, nonparametric alternative to Pearson Product-Moment correlation, Spearman's correlation coefficient, was used when replacing the data values for each variable by ranks because the variables are not normally distributed. The fact that variables are not normally distributed is due to the sample size.

The correlation between the degree to which specific type of KM technologies is being effectively utilized in an organization and that organization's OCP production-oriented and people-oriented value was examined using a non-parametric correlation analysis. A correlation coefficient of 0.66 was identified between the effective utilization of collaborative technologies and the OCP people-oriented value (see Table 4.). No other significant correlation was identified between those two variables, the effective utilization of KM technology categorized into two types, collaborative and distributive technology and organizational culture orientation.

Table 4. Correlation between effective utilization of KM technology categorized into two types and OCP Cultural Orientations

Correlations

			People-oriented	Production-oriented
Spearman's rho	Collaborative Tech.	Correlation Coefficient	.664**	.184
		Sig. (2-tailed)	.001	.699
		N	29	29
	Distributive Tech.	Correlation Coefficient	-.238	.431
		Sig. (2-tailed)	.359	.055
		N	29	29

** Correlation is significant at the .001 level (2-tailed).

Further analysis revealed other findings which are correlations between the 44 OCP attributes and the effective utilization of collaborative and distributive technologies for each organization. The non-parametric correlation analysis presented a number of cultural attributes having moderate to high positive correlation with the effective utilization of collaborative technologies while there is no obvious negative correlation between them. These attributes include working closely with others, being team-oriented, having a good reputation, and sharing information freely. Two cultural attributes having a moderate to high positive correlation with the effective utilization of distributive technologies were also identified. These attributes are being rule-oriented, and being result-oriented. Both of them are production-oriented cultural attributes. Even though there was not found any obvious correlation between the effective utilization of distributive technologies and organizational culture orientation, the effective utilization of distributive technologies seems to be more related to production-oriented orientation than people-oriented orientation. The positive correlations between the effective utilization of both collaborative and distributive technologies and the 44 OCP cultural attributes are shown with their correlation coefficients in Table 5.

Table 5. Correlation between Effective Utilization of KM Technology and OCP Cultural Attributes

OCP Attributes	Collaborative Tech.	Distributive Tech.
Working closely with others	0.67	
Having a good reputation	0.60	
Being team-orientated	0.59	
Sharing information freely	0.58	
Being rule-oriented		0.67
Being result-oriented		0.61

6.2 Analysis of Hypotheses

The Spearman's Rank Correlation coefficient (Rho) was used to determine the relationship between two quantitative variables measured in an interval scale by replacing the data values for each variable by ranks because the variables are not normally distributed. The Pearson Product-Moment correlation could be used with a sample size larger than 30 if the variables are approximately normally distributed. However, the sample size of this research (n=29) is not sufficiently large to use the Pearson Product-Moment correlation coefficient. The hypotheses were tested based on the findings from the correlation analysis with 99% confidence interval.

Table 6. Testing Hypothesis I

Culture Orientation	Correlation	t - value
People-orientated	0.66	t : 3.34 > $t_{.005,\ 25}$: 2.79

The hypothesis I postulates organizations indicating the effective utilization of collaborative technologies, would find that employees rank people-oriented attributes more positively in their assessment of organizational culture attributes than employees within organizations indicating a less effective utilization of collaborative technologies. The t-value (Table 6.) was calculated against Spearman's Correlation coefficients of people-oriented culture (r = 0.66) found from the data analysis. The-value is sufficient to reject null hypothesis (Table 6.) then reveals there is a positive correlation between the effective utilization of collaborative technology and people-oriented cultural orientation for that organization in the population.

Table 7. Testing Hypothesis II

Culture Orientation	Correlation	*t* - value
Production-orientated	0.43	$t: 2.11 < t_{.005, 25}: 2.79$

The hypothesis II postulates organizations indicating the effective utilization of distributive technologies, would find that employees rank production-oriented attributes more positively in their assessment of organizational culture attributes than employees within organizations indicating a less effective utilization of distributive technologies. The t-value (Table 7.) was calculated against Spearman's Correlation coefficients of production-oriented culture (r = 0.43) found from the data analysis. The t-value is not sufficient to reject null hypothesis (Table 7.) then reveals there is no obvious correlation between the effective utilization of distributive technology and production-oriented cultural orientation for that organization in the population.

7 Conclusions and Recommendations

Before an organization puts knowledge management technologies for a successful KM implementation, it should deal with cultural issues. The success of KM technology implementation is mediated by human behavior. While this research focused on establishing a correlation between organizational culture orientations and the successful implementation of KM technology, evidence suggests that the specific cultural attributes are the drives for or barriers to the successful KM technology implementation. Although focusing on organizational culture and change may extend the time it takes to prepare a KM program, the benefits of doing so include being better prepared for implementation and being more able to leverage existing technology.

The results of the data analysis revealed sufficient evidence to establish a correlation between cultural orientation and effective utilization of collaborative technologies. Employees of organizations, which are more effectively utilizing collaborative technology, have identified people-oriented culture. The research further identifies cultural attributes, which have moderate to high positive correlation with the effective utilization of collaborative technologies such as working closely with others, having a good reputation, being team-oriented, and sharing information freely. On the other hand, there was not found any obvious correlation between the effective utilization of distributive technologies and organizational culture orientation. However, two production-oriented cultural attributes having a positive correlation with the effective utilization of distributive technologies were identified.

This study could help researchers in the field of KM develop a better understanding of the role of cultural climate in the successful implementation of collaborative and distributive KM technology. This study could allow practitioners initiating KM programs to identify their current culture style with quantitative methodology used in this research and suggest a direction of changing it before they put technologies for a successful implementation of KM. The positive and negative cultural attributes to the successful KM technology implementation identified in this research could prove most beneficial to those organizations interested in a successful KM initiative. Organizations should provide their employees with those positive attributes through training programs and incentive systems.

References

1. Blake, Robert R. and Mouton, Jane S., The Managerial Grid III, Houston, TX, Gulf Publishing Company, 1985.
2. Davenport, Thomas, and Prusak, Laurence: Working Knowledge. How organizations manage what they know, Harvard Business School Press, MA (1998)
3. Dyer, Greg: KM Crosses the Chasm: IDC State of the Market Survey, *Knowledge Management*, March (2000) 50-54
4. KM Review survey reveals the challenges faced by KM practitioners, Knowledge Management Review, Vol. 4, Iss. 5 (2001)
5. KPMG Consulting: *Knowledge Management Research Report*, KPMG (2005)
6. National research Council, Computer Science and Telecommunications Board: Information Technology in the Service Society, National Academy Press, Washington, DC (1994)
7. Nonaka, Ikujiro, The Knowledge-Creating Company, Harvard Business Review, November-December, 1991, 96-104.
8. O'Reilly, Charles A., Chatman, Jennifer and Caldwell, David F., People and Organizational Culture: A Profile Comparison to Assessing Person-Organization Fit, Academy of Management Journal, Vol. 34, No. 3, 1991, 487-516.
9. Park, Heejun, Critical Attributes of Organizational Culture Promoting Successful KM Implementation, Lecture Notes of Computer Science, Vol. 3482, 2005.
10. Zack, Michael H., Managing Codified Knowledge, Sloan Management Review, Summer, 1999, 69-82.

Structured Knowledge Transfer in Small and Medium Sized Enterprises

Tanja Peherstorfer and Bernhard Schmiedinger

Profactor Produktionsforschungs GmbH, Im Stadtgut A2,
4407 Steyr-Gleink, Austria
{tanja.peherstorfer, bernhard.schmiedinger}@profactor.at

Abstract. Small and medium sized companies face the problem that handing over management or key positions often lead to loss of knowledge and decision competence. The loss of knowledge often causes miss-management, decrease of efficiency and also effectiveness in key processes. In the paper a structured hand over process handling detailed preparation, extensive know-how transfer and guided introduction of the successor will be introduced as well as a case study in a manufacturing SME. Major topic of the structured knowledge transfer process is to enable organizations to plan and handle leaving or retirement of key persons without loss of key competence.

1 Introduction

Within the next five years up to 20% of the SMEs[1] in Austria, Germany and Switzerland have to manage a follow-up regulation concerning owner or management [1][2][3]. Beside legal and economic factors the follow-up situation affects also transfer of knowledge and contacts (social knowledge). As there are not only follow ups within families (~60% [3]) it is also necessary to introduce external managers. Most failures in this situation are bad planning, no use of external help and insufficient know-how [3].

But not only follow-up situations in leading positions are vital for SMEs. All key positions must be kept in mind concerning personal changes. Production managers, research staff or assistants of managers carry essential knowledge in their heads, which is lost or hardly accessible for their successors without structured knowledge transfer.

2 Problem Description

In consideration of the fact that the bigger part of the know-how, used for solving complex problems, is undocumented and therefore hardly transferable or learnable, there is a great demand for alternative ways of knowledge transfer. The addressed alternate way focuses on the knowledge transfer from experts to their (know-how) successors. Therefore it's a task of transferring know-how existing as long term generated experience.

[1] SMEs = Small and Medium sized Enterprises.

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 234 – 242, 2006.

Professional knowledge is not directly addressed, the focus rather lies on transferring more specific task-, process-, product- or context knowledge, that is used unconscious by the knowledge owners (experts) [10]. This expert knowledge is prior noticeable in the following areas:

- Decision making - which parameters, persons, general conditions, etc. are considered in the decision making process?
- Processes / workflows – which procedures or approaches have been successful?
- Information evaluation – which available information is relevant?
- Information acquisition – how to get relevant information from which source?
- Competences – which organizational competences [4] are available and how can they be used to fulfil existing tasks?
- Problem evaluation – which problems exist and which are new? How are new / known problems handled?
- Networks – which communication or team structures exist and how can they be used?

These knowledge areas and the associated know-how are hard to identify for the expert [10]. Based on this it is also hard to find out the right way to transfer this experience to a successor. In organizations the awareness concerning the relevance of this knowledge is often missing. Similarly it is difficult for the knowledge receiver to substantiate his knowledge demand and to identify whom to ask in case of unclear situations. Therefore the main job of the following described transfer-coach is to support knowledge owners and receivers in identifying relevant knowledge areas and also the transfer process.

3 Transfer Coaching – Structured Knowledge-Transfer

Time is an important fact if knowledge transfer is concerned. Experienced employees (experts) and their successors have only limited time to share their know-how, which is based on the leaving date (ex. retirement) of the expert [5][7]. In most cases it is not possible to access the know-how of the leaving employee after his discharge or organisational change. To use the given time most efficient for transferring the grown experiences as complete as possible there is a great need for a target oriented transfer-process, which is managed by a third person (dialog-coach or transfer coach).

3.1 Objective

Intention of the structured knowledge transfer is to transfer knowledge as efficient as possible from the owner of knowledge (predecessor) to the recipient (successor). The affected knowledge is in most cases tacit knowledge, which requires a personal exchange of experiences [8]. Structuring the transfer process and using a dialog coach enables the accessing of intrinsic know-how of the predecessor.

3.2 Procedure of the Knowledge Transfer

Moderated transfer dialogues of owner and recipient of the knowledge are only one part of a multi-level transfer process, which consists of preparation, transfer, use and feedback phase (see Fig. 1).

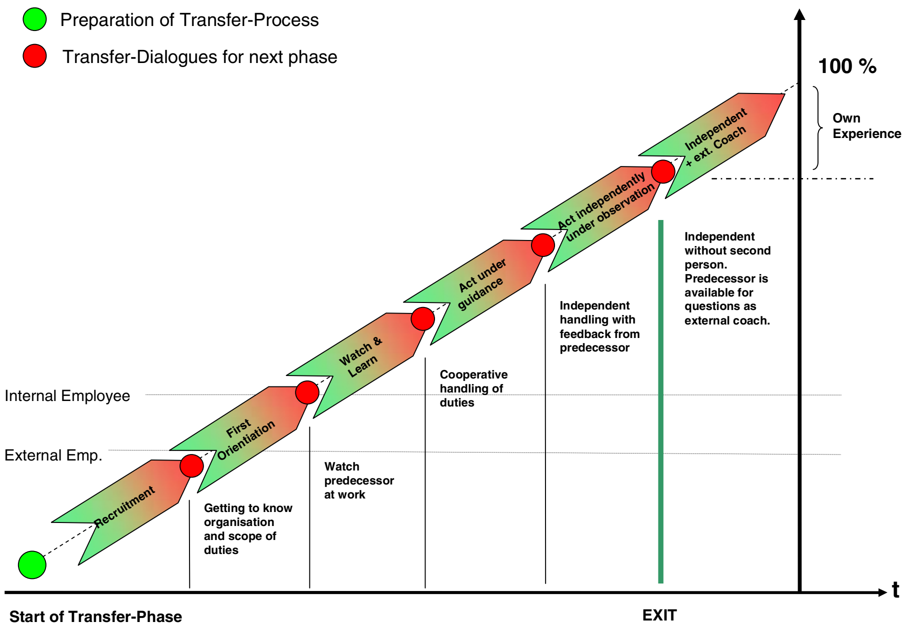

Fig. 1. Knowledge Transfer Process

Except the preparation phase all steps of the transfer process (transfer, usage and feedback) are executed several times. Knowledge transfer and feedback about using transferred know-how will take place during a structured transfer dialogue. The application of the transferred know-how is done afterwards at daily duties [9].

Preparation

During the preparation phase all persons are identified who are owner of essential know-how and who are without know-how substitute. This could be managers but also engineers or operators. If their knowledge is unique and vitally important for the future existence of the organisation there must be several efforts to keep this know-how. Another target group of the preparation phase are employees who will retire or leave within the next years and who own essential knowledge.

A development plan will be prepared in cooperation with the knowledge owner which contains details about his current duties, the transfer (start, costs, specials, etc.) and also his own future development. Together the coach and knowledge owner make a job and work description which describes the needed qualifications of the successor.

Based on this job specification the recruitment will be done internal or external. Depending on internal or external staff the successor has to pass through different phases of the transfer process (see Fig. 1 internal employee vs. external employee).

Transfer

Central point of the transfer phase is the transfer dialogue between knowledge owner (predecessor) and knowledge receiver (successor), moderated by a transfer coach or dialogue attendant.

- Preparation (transfer-plan). To structure the dialogue and to come to the wanted results, it is necessary to define a transfer plan in the run up of the process. This plan contains contents of dialogues, goals and dates. Details to each dialogue like content, project experiences, relevant questions, special documents, etc. will be prepared by each participant in advance.
- Transfer dialogue. The conversation during the transfer is a kind of a moderated dialogue. Knowledge owner and receiver exchange the prepared contents, supported by the dialogue supporter (coach) who asks concrete questions to control the process to the planned results. After the dialogue the partners (especially the knowledge receiver) make a short reflection about discussed topics to close this round of knowledge transfer.

Application of Knowledge

According the transfer process of Fig. 1 the next step after the transfer phase is the application of new gained knowledge. The degree of independent work depends on pre-existing knowledge of the successor, who will increasingly use the new knowledge. Ultimate goal of the transfer concept is the guiding of the knowledge receiver from “first orientation” to “independent fulfilment of duties”.

Feedback

This phase shall be used by the knowledge receiver (successor) to discuss open questions and problems, raised during the application of knowledge, with the knowledge owner. The feedback conversation is also moderated by the dialogue supporter. Beside the review of selected tasks/topics this step is also used to define responsibilities for the future (from current expert to successor).

3.3 Role- / Task-Arrangement

Knowledge Owner

Major task of the knowledge owner is the identification of relevant transfer topics [7]. In this context it is essential that he is willing to share the experiences identified before. To ease the transfer of knowledge the actual owner could support it by appropriate preparation like existing project documentation. This is also important for the preparation phase. In cooperation with the transfer coach (or personnel development) the development plan will be made. Further the knowledge owner supports the creation of the job profile of his successor.

Knowledge Receiver

First job of the knowledge receiver is to identify his lack of knowledge to fulfil his new duties. Provided that it is not a complete new task, he should think during the preparation about open questions, potential problems or improvements. As future knowledge owner and decision maker it is also up to his responsibility to prepare contents of the transfer dialogue. The application and resulting experiences shall be critically reflected during the feedback dialogue

Transfer-Coach / Dialogue Supporter
The role of the transfer coach / dialogue supporter could be done by external persons or by employees, who are not directly involved in the addressed topics. Ideally an employee of the HR department accompanies and coordinates the knowledge transfer. Depending on the position of the dialog supporter within the organization (ex. Personnel development) he will also manage preparation of dialogues, the generation of employee development plans and also the identification of knowledge receivers. Main job of this person is to accompany and moderate the personal knowledge transfer of expert and successor. To reach this he initiates the creation of the transfer plan and coordinates the transfer and feedback dialogues.

3.4 Results

The transfer is an iterative process containing knowledge transfer, application and feedback. All identified experiences will be transferred step by step from knowledge owner to knowledge receiver. By using the transferred knowledge the successor generates new experiences and increases his competences which make a new expert out of him, who ideally combines proven know-how with new one.

4 Case Study

Change of generation: long-time employees leave the company and take their knowledge established over decades with them. A SME of the packaging industry faced this problem. If experts leave the enterprise, the loss is even greater when these experts or their knowledge can not be simply replaced by employment of new staff members.

In the case of this SME the problem was, that these knowledge owners, who intended to leave the company within one or two years, had huge knowledge concerning the technological an economical development of the company as they were initiators and supporters of these trends. These experiences developed over decades can not be learned within a few months or within a year or two. Another challenge was the lack of training opportunities and job descriptions for the future experts. Training-on-the-job was in many cases the only way to evolve into a technical expert.

4.1 My Experience. My Knowledge

Besides lacking training opportunities the corporate culture and the expert’s attitude toward knowledge transfer played an important role. There were two ideological clusters established:

- Experts who did not want to provide their knowledge because they were afraid to lose their power and status.
- Experts who were not aware of the value of their knowledge. They had problems to identify their knowledge or their experiences and to transfer it to their successors.

As the origin of the succession problem is based on retirements, the transfer of knowledge included the transfer of decision-making authority. As a result the loss of authority posed a problem for some of these experts.

4.2 Year of Knowledge Transfer

The time factor plays an important role within producing SMEs [11]. In the case of the actual company, reaction time is their primary selling point. The ability to reduce the period between incoming order and delivery to a minimum and to increase flexibility in production is essential. Besides the importance in production, the time factor also played a prior role within knowledge transfer, as the availability of the leaving experts was limited to at most two years.

In collaboration with a representative of the human resource department and the current knowledge owners a detailed development plan including a realistic time management of the knowledge transfer has been prepared. This employee development plan included following details:

- Field of responsibility of the current knowledge owner
  - function / responsibility / decision-making authority
- Objectives of the current knowledge owner
  - leaving / availability / kind of availability
- Details of the knowledge transfer process
  - beginning of the preparation / beginning of the transfer / duration of the transfer
- Specifics during knowledge transfer
  - i.e. special projects / share or rather combine responsibility
- Potential knowledge recipient
  - future knowledge owner

By means of this employee development plan which describes the future role of the leaving experts, it was possible to define the duration of the knowledge transfer. In the case of the actual company the process of knowledge transfer was restricted to one year.

The knowledge transfer has been carried out according to the transfer process drafted above. All single steps starting with preparation or rather recruiting till the independent fulfilment of the assigned tasks ("independent with monitoring") has been divided into twelve month. The final process step – independent with external coach" – starts with the retirement of the knowledge owner" (month 12). In the case of the analysed enterprise this last step was shifted backward as the current experts are still available after their retirement. For the overall documentation it is necessary to write down all results out of the individual phases of the knowledge transfer process, including the date till particular process steps have to be completed.

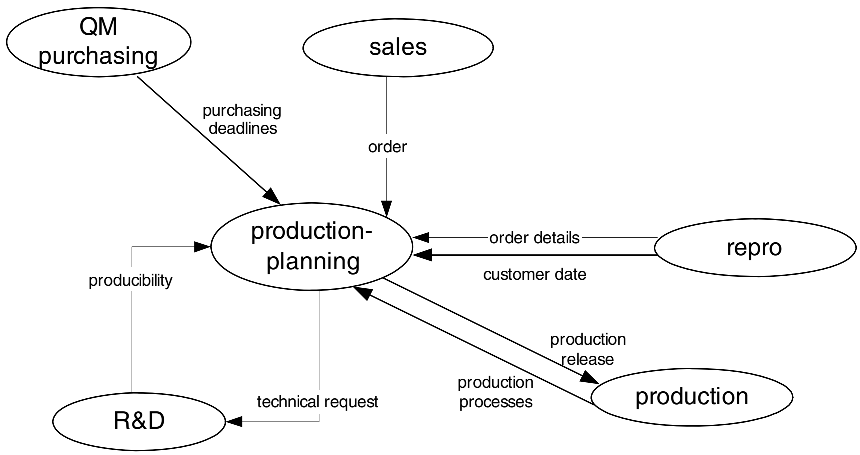

Fig. 2. Value Network - e.g. production planning

4.3 Who Are the New Knowledge Owners?

To identify those who have the potential to follow in long-time expert's footsteps is not an easy challenge. In collaboration with the human resource department and the current knowledge owners (predecessor) job and function descriptions for the potential successors have been developed. These descriptions included technical qualifications, education and work experience. Information about duties, responsibilities and decision-making authority of the future knowledge owners are important within the recruitment process and should therefore be documented in the job description of the knowledge successors. Because of the high demands on technical competences and failing of well-trained employees the company decided to fill the job in-house. Based on the developed job description and the references of the current knowledge owners the employees have been selected.

4.4 Value Network

For the experts, identified as valuable knowledge owners, putting their experiences and knowledge into words and summarize it into best practices, in order to transfer it to a third person, was not an easy job. This succession problem concerned a shift foreman, a research and a development director. All of these knowledge owners had to transfer their knowledge as soon as possible to their successors as they were leaving the company within at most two years because of illness or retirement. In order to facilitate knowledge structuring the daily work processes have been firstly pictured in terms of a Value Network[2] (s. Fig. 2) [6].

On the basis of this Value Network it was possible to identify those topics and problems which are important for the successors as they need this information for fulfilment of their future duties. The visualization of communication and knowledge

[2] http://www.alleevaluenetworks.com/ … Value Network Analysis, developed by Verna Allee, is a business analysis tool for modeling value-creating relationships.

structures was guided from the transfer coach. In case of the viewed company the human resource department acted the part of the transfer coach.

4.5 Structured Knowledge Transfer – Training Plan

Prior objective of the actual enterprise was to use of the running knowledge transfer process for purpose of developing a detailed training plan which can be used to educate current and future employees. Besides relevant topics for the particular position or role this plan included following information:

- Topic / skills / priorities. One topic requires one or more skills; the relevance of the particular skill for the future knowledge owner is defined by priorities.
- Status. The status informs about the degree of the existing skills. It provides information about the current expertise of the future knowledge owner, to what extend he is able to fulfil the associated duty on their own.

This training plan has been developed for the future knowledge owners in cooperation with human resource department and the current experts. The plan must be updated regularly within the transfer interviews or discussions between knowledge owner and knowledge receiver. Both, predecessor and successor, document with help of the transfer coach to what extend the defined training objectives have been reached, which skills are missing and adapt the current state of knowledge, concerning the particular skills.

5 Conclusion

The structured transfer of knowledge requires extensive investment into coordination of the involved knowledge owners and receivers, preparation of the transfer dialog and regular review and adjustment of the knowledge transfer process between predecessor and his successor. Besides the emotional barriers associated with the transfer - leaving the company, restart, losing decision-making authority, replacement by successors – general conditions concerning knowledge documentation and missing faith in the skills of the future knowledge owners should attract adequate consideration. The more structured the knowledge transfer process and the better the quality of the knowledge documentation the easier the whole transfer. This brings us back to the starting point – the time factor. The quality of knowledge transfer and therefore the qualification of the future experts depend on the available time and personnel resources. Consequently it is necessary to start with the planning of the knowledge transfer as soon as possible.

6 Outlook

In the actual model the competences and know-how are investigated by using the training plan. This approach shows “only” the obvious existing skills and know-how which is used in the daily business of the organisation. Latent (invisible) competences are not addressed and therefore not part of the actual knowledge transfer program. To improve the actual model it is planned to combine it with the "Competence based Business Development (CbBD)" Model [4]. This approach analyses all existing competences (used and unused) within an organization to provide a wide competence-set for future activities. Additional benefits are:

- early succession planning
- training of potential successors
- identification of know-how / competence owners
- separation of personnel successors and competence successor

The last point should strike out that the successor must not be one person but could also be a team which handles the actual existing agendas, roles and competences. Doing this enables a company to reduce dependency on the successor's competences and increases the flexibility in recruitment.

Additional the approach shall be integrated into existing succession topics (legal, economic, financial) to allow a holistic solution. To reach these goals further implementations and also an intensive exchange with organisations like the chamber of commerce will be made.

References

1. Keese, D., Ballarini, K.: Change in Hand Over Sphere - A Comparison of the Situation and Estimations of the Predecessors, RENT XVIII 2004, Managing Complexity and Change in SMEs, Kopenhagen, 24. - 26. November (2004)
2. PricewaterhouseCooper AG: Nachfolger gesucht - Empirische Erkenntnisse und Handlungsempfehlungen für die Schweiz, HSG St. Gallen (2005)
3. Pichler, H., Bornett, W.: Wirtschaftliche Bedeutung der kleinen und mittleren Unternehmen (KMU) in Österreich. In. Schauer, R., Kailer, N., Feldbauer-Durstmüller, B. (Ed.): Mittelständische Unternehmen - Probleme der Unternehmensnachfolge (2005) 117-150
4. Schmiedinger, B., Valentin, K., Stephan, E.: Competence Based Business Development - Organizational Competencies as Basis for Successful Companies, Journal of Universal Knowledge Management, Volume 0, Issue 1 (2005)
5. Maier, R.: Knowledge Management Systems, Second Edition, Springer, Berlin (2005)
6. Allee, V.: The Future of Knowledge. Increasing Prosperity through Value Networks., Butterworth Heinemann (2003)
7. Pfeffer, J., Sutton, R.: The knowing-doing gap : how smart companies turn knowledge into action, Harvard Business Press, Boston (2000)
8. Polanyi, M.: Tacit Dimenstion, Doubleday, New York (1966)
9. Staudt, E. et al.: Kompetenzentwicklung und Innovation, Waxman, Münster (2002)
10. Jetter, A.. et. al.: Knowledge Integration – The practice of Knowledge Management in Small and Medium Sized Enterprises, Physica, Heidelberg (2006)
11. European Guide to Good Practice in Knowledge Management, CEN - European Committee for Standardization, www.cenorm.be (2004)
12. PAS 1062:2006-05, Einführung von Wissensmanagement in kleinen und mittleren Unternehmen, DIN – Deutsches Institut für Normung e.V. (2006)

How to Transfer a Knowledge Management Approach to an Organization - A Set of Patterns and Anti-patterns

Anne Persson[1] and Janis Stirna[2]

[1] School of Humanities and Informatics, University of Skövde, P.O. Box 408, SE-541 28 Skövde, Sweden
anne.persson@his.se
[2] Department of Computer and Systems Sciences, Royal Institute of Technology, Elektrum 100, SE-164 40 Kista, Sweden
js@dsv.su.se

Abstract. The succesful implementation of a knowledge management approach in an organization is inherently difficult and risky. This paper presents, in the form of organizational patterns some concrete advice that will improve the possibilities of a knowledge management initiative to survive the implementation phase. The sources of the advice are a number of case studies that were carried out in private and public organizations.

Keywords: Knowledge Management implementation, organizational patterns.

1 Introduction

Knowledge Management (KM) has established itself as good management practice for modern organizations that strive to be efficient and competitive. Most large and middle size organizations have either some knowledge management activities in place or are planning some. While organizations commonly have a number of KM initiatives at various organizational levels, many of them fail to make the desired impact. KM approaches, methods and tools are tried out but the results are often unimpressive. One of the reasons for these problems is that the adoption process is too ad hoc and unplanned. The organizations attempt to follow a set of generic advices such as “start small and build-up gradually” without enough internal expertise or they rely on external consultants whose attitude is “we tell you what you want and then we will build it for you”.

Therefore, the objective of this paper is *to capture and package, in the form of organizational patterns, a set of proven good and bad practices of introducing knowledge management approaches in organizations*.

The research approach is conceptual and argumentative based on a number of case studies that were carried out in public and private organizations [16] [18] [12] [13] [19].

The remainder of the paper is organized as follows. In section 2 we provide a background to knowledge management and patterns. Templates for patterns and anti-patterns are also introduced. Section 3 contains good practices for introducing KM approaches, while section 4 contains bad practices. The advices are given in the form

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 243 – 252, 2006.

of organizational patterns and anti-patterns, which are based on experiences gained from case studies in various types of organizations. Some conclusions and future work are, finally, discussed in section 5.

2 Background to Knowledge Management and Organizational Patterns

This section provides a background to the concepts of Knowledge Management (KM) and organizational patterns.

2.1 Knowledge Management

Modern organizations need to maintain a high level of innovation in their business and products, which requires them to flexibly adapt to rapid change in their environments. Among the main driving forces in this change process are people and their knowledge. Organizations need to utilize this knowledge in the most efficient way since, in essence, it is part of their competitive advantage. It is therefore that managing experience, competence, knowledge about business processes and best business practices are so important. This knowledge is part of the organizational memory.

The Knowledge Management (KM) process as described in (Fig. 1.) covers the whole lifecycle of knowledge in an organization. The cycle is adopted from O'Dell et al [11] and is similar to the spiral of organizational knowledge creation as presented by Nonaka and Takeuchi [10]

Creating knowledge can be done in many different ways – running day-to-day business operations, improving existing work routines, restructuring the organization, planning organizational strategies for the future, etc. Often the creators of knowledge are not aware of this and valuable knowledge may therefore be lost. To prevent this, the knowledge needs to be captured in one way or another. This might require thinking in abstract terms, building models and/or mind maps, or simply writing down the experiences. Most often this should be done in a participative and collaborative way, which enhances one's individual view.

Once knowledge is captured, the organization and its employees are aware of its existence. If the captured knowledge is relevant the next step is to package and store the knowledge so that it is available and can be used by those who need it in the organization. The key element here is to make the specific knowledge *useful*. This usually requires some degree of generalization of knowledge. Furthermore, it also requires envisioning how and in which context each knowledge chunk will be used. The knowledge that is written down in some form usually resides in repositories, manuals, the intranet, etc. However, not everything can be written down. Most often the tacit knowledge is the most important knowledge. In this case we can only write down who knows what, where the knowledge sources are, and how to access it. This also becomes an important part of the organizational/memory/knowledge repository/knowledge base. After knowledge is properly documented and stored, it needs to be shared and applied. This probably is the most important task in KM. Knowledge sharing cannot be done mechanistically – it is not enough to install and fill a knowledge base and expect the organization to suddenly start sharing knowledge. Therefore, particular attention should be paid to building a knowledge sharing culture in the organization supported by organization's leadership [3] [14] [6] [17] [20].

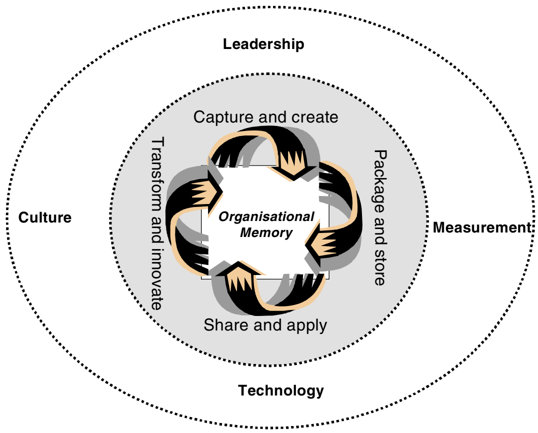

Fig. 1. The Knowledge cycle in organizations

Technology can only play a supporting role in knowledge sharing and application – it can make knowledge sharing easier and more effective. Successful as well as effective knowledge sharing and application also stimulates innovation - improvement of existing knowledge and creation of new knowledge. This essentially closes the knowledge cycle.

2.2 Organizational Patterns

Alexander et al [1] define a pattern as describing *"a problem which occurs over and over again in our environment and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice"*. Following this definition, pattern based approaches have established themselves in software programming, software design, data modeling, and in systems analysis (see e.g. [8] [7] [5]). The notions of pattern from these areas share two main ideas – (1) a pattern relates a recurring problem to its solution, and (2) each problem has its unique characteristics that distinguish it from other problems. The common objective of applying patterns is to capture, store and communicate reusable content, such as fragments of code of class diagrams.

The pattern concept has been further extended and applied in organizational development and knowledge management (see e.g. [15] [16] [14] [19]). The community of design patterns emphasizes that patterns should always be well-proven solutions to reoccurring problems. Organizational patterns are instead viewed as *generic and abstract organizational design proposals that can be easily adapted and reused*. These represent solutions to specific problems within the context of an organization, problems that are important and recurring in a variety of cases – best practices. In summary, organizational patterns encapsulate organizational knowledge in a way that facilitates its reuse.

In order to facilitate the reuse process patterns usually are structured according a pattern template. There are various pattern templates that can be found in books and on the internet. The common principle is that a pattern should define the problem that it solves and the solution to that problem. The pattern should also define when it can be reused and how.

Figure 2 shows the pattern template used in this paper.

Name	The name of the pattern
Problem	KM oriented motivation or problem it tries to solve
Context	When does the problem occur?
Goal	What is the goal for the organization or the individual?
Solution	What is the solution to the problem?

Fig. 2. The pattern template

Isolated patterns make sense only in solving small trivial problems. For problems beyond this trivial level we need to look at the relationships between the patterns in order to provide a more complete solution. The set of constructed patterns makes up a *pattern language*. Different patterns from a pattern language may be combined in different alternative ways to adopt different solution paths to different facets of the overall problem. The patterns presented in this paper can potentially be developed into a pattern language for introducing KM approaches in organizations.

2.3 Anti-patterns

A pattern is a proven successful solution to a recurring problem. An anti-pattern (see e.g. [2]) is a *bad solution* to a common problem. Besides just presenting a bad solution, a good anti-pattern should also explain why this solution looks attractive in the first place and why it, in fact, turns out to be bad.

We use a template for anti-patterns adopted from Long [9], see figure 3.

Name	The name of the anti-pattern
Problem	KM oriented motivation or problem it tries to solve
Anti-solution	How it is applied
Actual results and unintended consequences	What happened after the it was applied
Primary fallacies	What were the likely causes for failure

Fig. 3. The anti-pattern template

3 Patterns for KM Adoption

In this section we present a set of patterns that contain solutions to a given number of problems that occurred in all the case studies that were carried out.

Select champion high up in the organization	
Problem	Acquire resources to implement KM
Context	The organization has not previously used KM in a systematic manner but has identified the need to introduce a KM approach. Resources, in the form of money and man-power, are needed to effectively carry out the implementation project and to make the project survive as a long-term way of working.
Goal	Make the implementation process smooth and effective and make sure that the project survives also after the implementation phase.
Solution	Acquire the support from a respected champion high up in the organization and make sure that the support from this person is clearly visible to everyone in the organization. The KM champion should not only support KM in words but also be active in knowledge sharing, e.g. be visible in statistics about visits to the knowledge base, knowledge objects read, commented, created; participate in person KM related events.

Select champion at lower levels in the organization	
Problem	Start knowledge work, e.g. capturing and sharing
Context	The organization has decided to introduce a KM approach either in a pilot project and/or on an institutional basis, but not much practical work has been done.
Goal	Create a critical mass of activities and knowledge contents in the knowledge base.
Solution	Select people that: - understand the business need for knowledge sharing, - would personally benefit (e.g. in terms of greater work efficiency) from the adoption of the chosen KM approach. - understand the basic concept of KM, - are willing to share their knowledge as well as learn from other's, - are highly respected among colleagues in the work group, and - have time to do the practical work (e.g. creating the initial repository structure and contents, facilitating discussions, arranging meetings)

Involve the IT department	
Problem	Modern KM solutions are normally supported by software applications which need to operate in a corporate IT environment controlled by an IT department.
Context	When introducing a new KM tool it is usually the IT department's responsibility to manage the technical aspects of this process. In some cases the IT department's managers may feel that they should have been consulted about purchasing this software, or that this software is not needed because similar tools have been used in the past with little or no success. Or that the same effect could be achieved with other tools as well (e.g. HTML editors).
Goal	Introduce the IT support for KM activities early and as efficiently as possible.
Solution	Collaboration with the IT department is needed and should be established as early as possible. The IT department needs to understand that using a particular tool or system is a business decision supported by the top management. Involve the KM champion from the top level, if necessary.

Tailor the KM approach	
Problem	The organization has specific KM needs, including internal ways of working as well as requirements for the KM system.
Context	The organization has decided to adopt KM and is in the process of assessing various approaches, techniques, tools, and applications.
Goal	Choose an approach that fits organically the way of working and the overall IT environment
Solution	The KM approach should be tailored according to specific needs of the organization. External consultants and vendors should be consulted because they can often share experiences and best KM practices of similar organizations. However, the organization should have realistic expectations from this process. Demands such as “make this very simple and cheap” will not be met most likely. The people from the organization should be actively involved in the tailoring process because this allows them to learn about the approach and its various components.

Develop the knowledge base in a participative manner	
Problem	Only a few people in the organization consider that they share the ownership of the knowledge base.
Context	The knowledge base is developed by a few people on in the organization without discussing its contents with others that may contribute to its quality and usefulness.
Goal	Create a feeling of responsibility and ownership among as many people in the organization as possible, hence improving the chances of the knowledge base to survive the implementation project and spreading the knowledge about knowledge management and about the content of the knowledge base.
Solution	Develop the knowledge base in a participative manner, involving different actors that may be able to contribute. The following tasks usually benefit from working participatively: - knowledge audit and development of a commonly shared knowledge map, - identifying crucial knowledge related problems to be addressed by the joint efforts, - developing the structure of the knowledge sharing tool or repository, - developing organizational processes and activities related to knowledge work - reviewing the repository contents including the user feedback.

Develop initial content	
Problem	The KM system or knowledge base should have some content in order to attract users’ attention and create desire of using it.
Context	Once the KM system as been installed the organization should start using it.
Goal	Create a user base which will eventually span the whole organization.
Solution	The project should analyze the knowledge needs of the target users. Performing a knowledge audit and creating a knowledge map might help at this stage. The content should be needed and valuable for solving real work problems. There should be no toy examples and trivial content such as e.g. menus of the local cafeteria.

4 Anti-patterns for KM Adoption

The anti-patterns presented in this section describe undesirable situations when introducing a KM approach in an organization.

Rely on consultants	
Problem	The organization does not have people who have the interest, skill and time to develop the knowledge content.
Anti-solution	The organization hires a group of external consultants who develop the knowledge contents based on e.g. “document sampling” and interviewing. The organization’s own people have limited involvement. A variant of this is to hire students or junior employees to perform the task.
Actual results and unintended consequences	The knowledge content that is developed and made available in the knowledge sharing tool is too generic resembling a textbook on the subject, there are few real life examples. On the whole the repository content does not address the knowledge needs of the knowledge users of the organizations and nobody uses it.
Primary fallacies	The assumption that creation reusable knowledge content of high quality and value is a trivial task that can be outsourced by the actual knowledge bearer.

Toy Pilot	
Problem	How to select a suitable pilot? In most cases the new technology (or method, way of working, approach, tool, system, portal) is tried in a pilot project. Most adoption guidelines suggest doing a pilot on the basis of which the suitability of the approach is then assessed to make the decision for institutionalization.
Anti-solution	Select a simple case concerning a few people who have “spare” time in a remote department because their “testing” of the new technology will not disrupt the “normal” way of doing business.
Actual results and unintended consequences	The team does its best and develops a case, an example, or a prototype within the given time and resource constraints. The problem addressed usually is peripheral tackling problems that only a few others care about. The result is presented in a presentation and while everyone is “welcome” to have a look, only a few actually do. The rest of the organization quickly decides that this new “technology” does not help them to do business.
Primary fallacies	We do not want to experiment with things that turn out to be useless for us. In addition, in strive for efficiency the organizations try to do everything as quick and cost-efficient as deemed possible. As a result they fail to understand that KM technologies are to be tested in real life situations on a sufficiently large scale, involving a serious number of employees.

Be overly cautious when selecting the knowledge content	
Problem	The organization is afraid that the knowledge content will not be interesting and valuable to the knowledge users. A variant of this is to adhere to imaginary rules and regulations restricting information spreading and publishing, or to attempt to mitigate imaginary risks.

Anti-solution	Be extremely and overly cautious when selecting knowledge chunks to be included in the knowledge sharing tool. A formal and usually time consuming approval procedure involving senior management representatives is set up.
Actual results and unintended consequences	Being overly cautious about which content to include can make the organization set up unnecessary and resource consuming control functions that may hinder the knowledge sharing process. In fact, the cautiousness can result in what the organization is afraid of: knowledge content that is not interesting and valuable to its users.
Primary fallacies	The risk in this case is the strive for perfection, which tends to cause paralysis.

Select champions by what they say	
Problem	The organization needs to identify champions that will promote the KM project implementation project.
Anti-solution	Identify and select champions mainly using the criterion of positive statements towards KM and the actual KM implementation project.
Actual results and unintended consequences	If it turns out that the positive statements were only word of mouth and were not followed by actual action, e.g. in terms of putting in resources like time and effort, the project does not have a champion at all. If the supposed champion is at the management level, the project will probably not impact in the organization and will not have the necessary resources. If the supposed champion is at a lower level, e.g. in the work-group where the knowledge sharing tool will be maintained, the group will not put in the needed effort and the project will not survive the implementation phase.
Primary fallacies	We want to believe what people say is actually true. We should also look at what people actually do to support the KM effort.

5 Discussion and Conclusions

In this paper we have presented a number of patterns and anti-patterns that aim to capture good and bad practices when introducing a KM approach in organizations. These patterns can be related among themselves which leads to forming an initial version of a pattern language (see figure 4). It shows two groups of patterns (rectangles) – (1) addressing who to involve in the KM adoption process and (2) addressing the issue of the initial development of the structure and content of the knowledge repository. Recommendations given in the patterns are also related to a number of pitfalls that are presented as anti-patterns (octagons). This pattern language will be improved further by introducing additional patterns to cover more aspects of the KM adoption process.

It is our experience from the cases that our work is based on, that a careful and well planned implementation strategy is critical for the success of KM implementation. The last of the case study is one which addresses knowledge sharing in and between health-care organizations about the treatment of leg ulcers. Here we can observe a significant improvement in comparison to previous cases, when it comes to the chances of survival of the knowledge base after the implementation project. It is fair to say that we have been able to resolve the main portion of the challenges addressed by the patterns and anti-patterns presented in the paper.

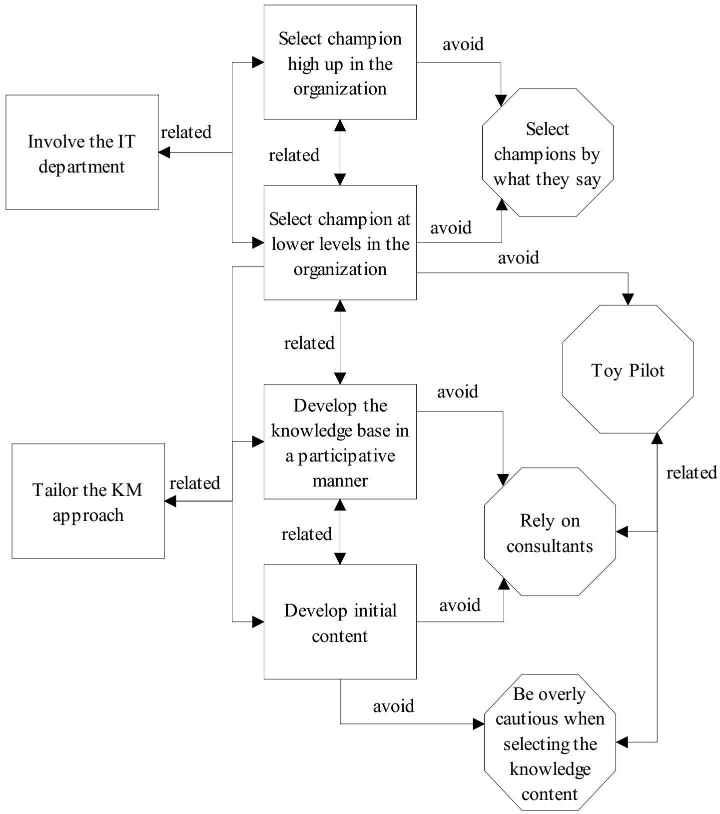

Fig. 4. Relationships between patterns

As can be observed from the patterns, the main problems are related to people and to organizational culture. This is due to the fact that it is people and not technology that actually carry out KM. Changing the attitudes of managers and employees as well as convincing them that the KM effort will benefit them personally is a difficult task that needs to be addressed from a number of different perspectives.

The presented patterns are far from a complete set. In the ongoing case study, that is now approaching its final stages we will improve and expand them as we continue to collect experiences and compare with our previous case studies. We also plan to construct an on-line pattern language from all the resulting patterns.

References

1. Alexander C., S. Ishikawa, M. Silverstein, M. Jacobson, I. Fiksdahl-King, S. Angel, A (1977) Pattern Language, Oxford University Press, New York
2. Brown, W.J.; Malveau R.C., McCormick III R.C., and Mowbray T.J. (1998). AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis, John Wiley & Sons. ISBN 0471197130.

3. Busch, P and Richards, D (2004) Tacit knowledge and culture The Proceedings of the 1st International Conference on Knowledge Management (ICKM'04), 13-15 December 2004, Singapore.
4. Chua, A. and Lam, W. (2005) Why KM projects fail: a multi-case analysis Journal of Knowledge Management, Vol. 9, No. 3, 2005, pp. 6-17
5. Coplien J., and Schmidt D. (Eds.) (1995). Pattern Languages of Program Design, Addison Wesley, Reading, MA.
6. Davenport, T.H. & Prusak, L. (1998) Working Knowledge Harvard Business School Press Boston.
7. Fowler M. (1997). Analysis Patterns: Reusable Object Models, Addison-Wesley.
8. Gamma E., Helm R., Johnson R. and Vlissides, J. (1995). Design Patterns: Elements of Reusable Object-Oriented Software, Addison Wesley, Reading, MA.
9. Long, J., (2001) Software Reuse Antipatterns, Software Engineering Notes vol 26 no 4, ACM SIGSOFT,
10. Nonaka, I. and Takeuchi, H. (1995) The Knowledge-Creating Company, Oxford University Press.
11. O'Dell C., Grayson J. jr, Essaides N. (1998) If only we knew what we know: the transfer of internal knowledge and best practice, The Free Press, ISBN 0-684-84474-5.
12. Persson, A. and Stirna, J. (2002), "Creating an Organisational Memory Through Integration of Enterprise Modelling, Patterns and Hypermedia: The HyperKnowledge Approach", In Kirikova et. al. (eds.), Information Systems Development – Advances in Methodologies, Components and Management, Kluwer Academic, New York, USA pp. 181-192.
13. Persson, A. and Stirna, J. (2003), The EKP approach to building organisational memories – experiences from two EU projects, in Eder, J. and Welzer, T. (Eds.), CAiSE' 03 Forum – Information Systems for a Connected Society, The 15th Conference on Advanced Information Systems Engineering, Klagenfurt/Velden, Austria 16-20 June, 2003, pp. 185-188.
14. Persson, A., Stirna, J., Dulle, H., Hatzenbichler, G. and Strutz, G. (2003), Introducing a Pattern Based Knowledge Management Approach - the Verbundplan Case, The 4th International Workshop on Theory and Applications of Knowledge Management (TAKMA 2003), Prague, Czech Republic, September 2003.
15. Prekas N., Loucopoulos P., Rolland C., Grosz G., Semmak F. and Brash D. (1999). Developing patterns as a mechanism for assisting the management of knowledge in the context of conducting organisational change, In Proceedings of 10th International Conference and Workshop on Database and Expert Systems Applications (DEXA'99), Springer.
16. Rolland, C., Stirna, J., Prekas, N., Loucopoulos, P., Persson, A. and Grosz G (2000), "Evaluating a Pattern Approach as an Aid for the Development of Organisational Knowledge: An Empirical Study" The 12th Conference on Advanced Information Systems Engineering, Stockholm, Sweden, June 2000, pp. 176-191, Springer.
17. Sandelands, E (1999) Learning organizations: a review of the literature relating to strategies, building blocks and barriers Management Literature in Review Vol. 1, 1999.
18. Stirna, J., Persson, A. and Kaindl, H. (2002), Evaluation of the trial applications, project deliverable IST project 2000-28401, Hypermedia and Pattern Based Knowledge Management for Smart Organisations (HyperKnowledge).
19. Stirna, J., Persson, A., and Aggestam, L. (2006), Building Knowledge Repositories with Enterprise Modelling and Patterns - from Theory to Practice, Proceedings of the 14th European Conference on Information Systems, Gothenburg, Sweden, June 2006.
20. Sun Yih-Tong, P. and Scot, J. L. (2005) An investigation of barriers to knowledge transfer, Journal of Management Vol. 9, No 2, 2005, pp. 75-90.

Developing a Model for Linking Knowledge Management Systems and Intellectual Capital Measurement

Mário Paulo Pinto[1], Maria Filomena Lopes[2], and Maria Paula Morais[2]

[1] ESEIG, Polytechnic Institute of Porto, Portugal
mariopinto@eseig.ipp.pt
[2] Portucalense University, Porto, Portugal
{flopes, pmorais}@uportu.pt

Abstract. Knowledge management systems (KMS) and intellectual capital (IC) measurement seek to increase the knowledge assets and the knowledge activities that bring competitive advantage to organizations. However, generally KMS ignore the IC measurement. This paper presents a model for linking these issues, showing the contribution of KMS to the IC measurement and their impact to organizations value creation. The model outlined in this paper should offer valuable guidelines to measuring the intangible assets through the knowledge wrapped in different KMS.

Keywords: Knowledge management, knowledge management systems, intellectual capital measurement.

1 Introduction

Today, it is widely recognized that organizational knowledge is the main source of competitive advantage and value creation. Organizations have recognised that their competitiveness is strictly related with the ability to create, store, distribute and apply their knowledge assets, in order to increase innovation, competitive advantage and future sustainability [11], [24].

IC measurement focuses the intangible assets from a strategic perspective, with the aim of showing their impact in value creation and their benefits to the organizations [42]. It covers such non-financial assets as, for instance, innovation capability, employee's creativity or customer's satisfaction, and it is oriented towards the future, focusing on the value creation and the core capabilities that bring competitive advantage. From this perspective, the IC measurement is helpful to verify the organization ability to achieve its strategic objectives. On the other hand, knowledge management (KM) focuses mainly on managing organizational knowledge with the aim of maximize knowledge-related effectiveness. In this context, KMS play an important role by supporting and enhancing the organizational processes of knowledge creation, storage and retrieval, distribution and application [1], [22]. However, KMS and IC measurement are generally viewed in a separate way, without connections and linkages [40]. The existing KMS normally don't produce IC indicators, creating a gap between KMS and IC measurement. According to Smith

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 253 – 266, 2006.

[40], this gap is one of the most critical shortcomings in the current practices of measuring and managing IC in organizations.

The purpose of this paper is to present a model to fill this gap, linking KMS and IC measurement. The model offers support to the IC measurement through KMS, showing the contribution of these systems to the value creation. It results from the reviews on IC measurement and KMS, and also from a survey conducted to one hundred of the biggest Portuguese organizations, with the aim of identifying the KMS categories used, the IC measurement practices and the metrics used to measure IC.

Research about IC measurement has produced several approaches and models over the last few years. Section two of this paper provides a brief literature review of these models. A systematization of KMS categories is made in the third section with the purpose of highlighting the different categories that are normally used to classify KMS, and section four presents the results of the survey conducted in Portuguese organizations. The model proposed in this paper is described in section five, as well as its components, the relationship between them and the measures that KMS can provide to support the IC measurement. Section six describes the model validation based on an expert panel composed by researchers and practitioners, while section seven provides some conclusions and draws some directions for future research.

2 Intellectual Capital Measurement Models

IC refers to intangible assets that can generate future economic benefits, i.e., value creation. Those assets are the key of competitive advantage and they are characterized by their invisibility, the difficulty in quantifying and acquiring them, without a monetary nature and without physical substance [28]. Its measurement reflects the value added by knowledge to the organizations [26], enables to monitor the performance of the knowledge assets and their related activities [32] and produces insights into how the organizations are managing, developing and using their knowledge assets [14], [24], [30], [34].

A review of IC measurement models was made with the purpose of identifying the main components used to measure IC. Table 1 summarizes these IC measurement models identifying the components specified in each one. The models describe different components like human capital, structural capital and relationship capital, as well as social capital, R&D capital, corporate identity, environment capital or others, depending on their own characteristics. However, human, structural and relationship capital are the most referred components [2], [9], [14].

- Human capital is concerned with individual capabilities, knowledge, skills, experience and abilities to solve problems. It represents the employee's competence, attitude and intellectual agility [38]. Competences include skills and education, while attitude covers the behaviour of the employees. Intellectual agility enables to think on innovative solutions and to change practices in order to solve problems [8].
- Structural capital is concerned with systems, organizational processes, technologies, concepts and models of how business operate databases, documents,

patents, copyrights and other codified knowledge. According to Roos [38], structural capital is what remains in the company when employees go home for the night.

- Relationship capital is concerned with alliances and relationships with customers, partners, suppliers, investors and communities. It also includes brand recognition, organization image and market position. The relationship capital represents the knowledge embedded and the value added from the relationships with other entities [3].

Table 1. A systematization of IC measurement models and their main components

IC Measurement Model	Human Capital	Structural Capital	Relationship Capital	R&D Capital	Social Capital	Environment Capital	Corporate Identity
Balanced Scorecard [23]		X	X	X			
Chen, Zhu & Xie Model [42]	X	X	X	X			
Citation-Weighted Patents [7]				X			
Danish Guidelines [32]	X	X	X	X			
Heng Model [18]	X	X	X	X			X
IC Rating [13]	X	X	X	X			X
Inclusive Valuation Methodology [29]	X	X	X				
Intangible Assets Monitor [41]	X	X	X				
Intangible Value Framework [2]	X	X	X		X	X	X
Intelect Model [16]	X	X	X				
Intellectual Capital Index [38]	X	X	X	X			
Intellectual Capital Rating [21]	X	X	X	X			
Intellectus [19]	X	X	X	X	X	X	
Meritum Guidelines [31]	X	X	X				
Nova Model [10]	X	X	X	X			
Skandia Navigator [15]	X	X	X	X			
Technology Broker [9]	X	X	X	X			
The 4-Leaf Model [24]	X	X	X				
The Value Explorer [5]	X	X		X			
Total Value Creation [4]	X		X				
Value Added Intellectual Coefficient [37]	X	X					
Value Chain Scoreboard [25]		X	X	X			

A systematization of the metrics proposed by each one of the IC measurement models reviewed was also made, with the aim of identifying a set of valuable metrics to measure the intangible assets.

3 Knowledge Management Systems

KMS refer to a class of information systems applied to manage the organizational knowledge [1]. They are based on information technologies (IT) and focused on supporting the organizational processes of knowledge creation, storage and retrieval, distribution and application [1]. Their main purpose is to enable an environment that facilitates the creation and usage of knowledge, and also the communication and collaboration among the organization.

Many authors have written about the use of various types of KMS [6], [12], [17], [20], [27], [33]. The classifications referred by these authors are based on different assumptions: some of them are based on technological issues, others on related functionalities; others yet join these two criteria in the same classification. On the other hand, some of those classifications don't make a clear distinction between KMS and traditional information systems [36]. The diversity of KMS classifications based on different approaches, takes us to develop a systematization of KMS categories regarding their addressed issues, capabilities and functionalities. This systematization encompasses the following KMS categories [36]:

- Business intelligence systems
- Collaboration systems (groupware)
- Competence management systems
- Corporative portals
- Document management systems
- E-learning systems
- Expert systems
- Knowledge discovery systems
- Knowledge maps
- Workflow systems

4 KMS and IC Measurement in Portuguese Organizations

A study was conducted in Portugal based on a survey sent to one hundred of the main Portuguese organizations in May 2005, with the aim of to know the current practices of the Portuguese organizations relating KMS usage and IC measurement. The organizations were selected from an annual publication that classifies them according their value creation. The survey questionnaire was structured in three main sections:

- Organization identification: it includes the organization name and its business area.
- Knowledge management systems identification: It comprises the identification of KMS categories used in the organization. The questionnaire presents the ten KMS categories identified in the previous section and the organizations could select the adequate categories or add new ones.
- Intellectual capital metrics identification: It comprises the identification of the metrics used in the IC measurement by Portuguese organizations. The questionnaire contains a comprehensive list of qualitative and quantitative metrics, grouped by human, structural and relationship capital, resulting from an extensive review of IC measurement models. This list includes 40 metrics for human capital, 71 metrics for structural capital and 33 metrics for relationship capital. However, the respondents could also complete this list, adding new IC metrics.

Fourteen valid questionnaires answers were received, corresponding to a response rate of 14%. All of the respondent organizations have selected a set of KMS categories, but only seven organizations have advanced with a set of metrics for measuring IC. According to the questionnaire answers, the remaining organizations don't proceed to a systematic IC measurement.

Table 2 summarizes the various KMS categories mentioned by respondents in the questionnaire, presenting the respective occurrence rate in industry and service organizations.

Table 2. Knowledge Management systems usage in Portuguese organizations

KMS categories	Industry	Service
Business Intelligence	75,0%	50,0%
Knowledge Maps	25,0%	25,0%
Document Management Systems	75,0%	50,0%
Collaboration Systems (groupware)	37,5%	50,0%
Workflow systems	62,5%	25,0%
Expert Networks	12,5%	25,0%
Competence Management Systems	75,0%	75,0%
E-learning Systems	12,5%	50,0%
Knowledge Discovery Systems	62,5%	100,0%
Corporative Portals	62,5%	75,0%

Table 3 summarizes the metrics referred by respondents, grouped by IC component: human, structural and relationship capital. The symbol X, in Table 3, identifies the metrics selected by industry and service organizations.

Table 3. Summary of IC metrics

	Metrics	Industry	Service	Metrics	Industry	Service
Human Capital	▪ % Employees of full-time	X	X	▪ Initiative capacity	X	X
	▪ % Employees of part-time	X	X	▪ Innovation capability	X	X
	▪ % male/female	X	X	▪ Investment in training per capita	X	X
	▪ % Specialized employees	X	X	▪ Leadership index	X	X
	▪ Absenteeism rate	X	X	▪ Motivation index	X	X
	▪ Average age of employees	X	X	▪ Nº of employees	X	X
	▪ Average IT literacy	X		▪ Nº of expert employees	X	X
	▪ Average of staff literacy	X	X	▪ Nº of managers	X	X
	▪ Years on company service	X	X	▪ Number of temporary employees	X	X
	▪ Average years with company	X	X	▪ Profits by employee	X	X
	▪ Distribution by age group	X	X	▪ Employees satisfaction index	X	X
	▪ Employee turnover	X	X	▪ Time in training (days/year)	X	X
	▪ Employees alternation	X	X	▪ Value added per capita	X	X
	▪ Experience index	X	X			

Table 3. *(continued)*

	Metrics	Industry	Service	Metrics	Industry	Service
Structural Capital	▪ Information technology capacity	X	X	▪ Investment in training		
	▪ Administrative expense/employee	X	X	▪ KM initiatives	X	X
	▪ Administrative expense/total revenues	X	X	▪ New solutions/products/business	X	X
	▪ Quality performance (ISO 9000)	X	X	▪ Certified products (#)	X	
	▪ Development cost of new products or services	X	X	▪ Investment in new methods and processes	X	
	▪ Expert employees/total employees	X	X	▪ New business opportunities (#)	X	
	▪ Hours in development	X	X	▪ New products (#)	X	
	▪ Hours in training	X	X	▪ Projects with partners (#)	X	X
	▪ Information availability	X	X	▪ Upgrading projects (#)	X	X
	▪ Innovation capability	X	X	▪ Employees until 40 years (#)	X	X
	▪ Innovative employees (#)	X	X	▪ Protocols with innovation entities	X	X
	▪ Investment in IT	X	X	▪ PCs by employee (#)	X	X
	▪ Investment in IT development	X		▪ Process efficiency index	X	X
	▪ Investment in new competences	X	X	▪ Productivity rate	X	X
Relationship Capital	▪ IT Capacity	X	X	▪ Investment in IT	X	X
	▪ Small/medium/high customers (%)	X	X	▪ Investment in marketing	X	
	▪ Administrative costs per customer	X		▪ Market share in segment	X	X
	▪ Annual sales per customer	X	X	▪ New customers/customers lost	X	X
	▪ Average duration of customer relationship	X	X	▪ Business alliances and partnerships (#)	X	X
	▪ Customer visits to the company(#)	X	X	▪ Employees that generate revenues	X	X
	▪ Customer satisfaction index	X	X	▪ Company image	X	X
	▪ Expert employees/total employees	X	X	▪ Customer relationship investment	X	X
	▪ Customers distribution	X	X	▪ Nº of customers claims (#)	X	X
	▪ Nº of customers (#)	X	X	▪ Revenues per customer	X	X

The results obtained from the survey show that there aren't strongly differences on industry and service organizations. The KMS usage and the IC metrics selected are almost the same in both business sectors.

The findings of this survey will be used in the model purposed in this paper to clarify the potential contribution of KMS in the IC measurement.

5 A Model for Linking KMS and IC Measurement

The main objective of the model proposed in this paper is to link KMS and IC measurement, showing the contribution of these systems to the value creation in organizations. The model can also facilitate the selection of appropriate KMS according the organization needs, aligning the KMS selection with the strategic objectives and the intangibles assets that bring competitive advantage. It provides an integrated view of intangible assets, covering the strategic and the operational perspective of KM.

The model is structured in three components, as is shown in Figure 1, which are: IC measurement model, knowledge management systems and IC measurement system. The objects defined in each component and the relationship between them is also described in Figure 1.

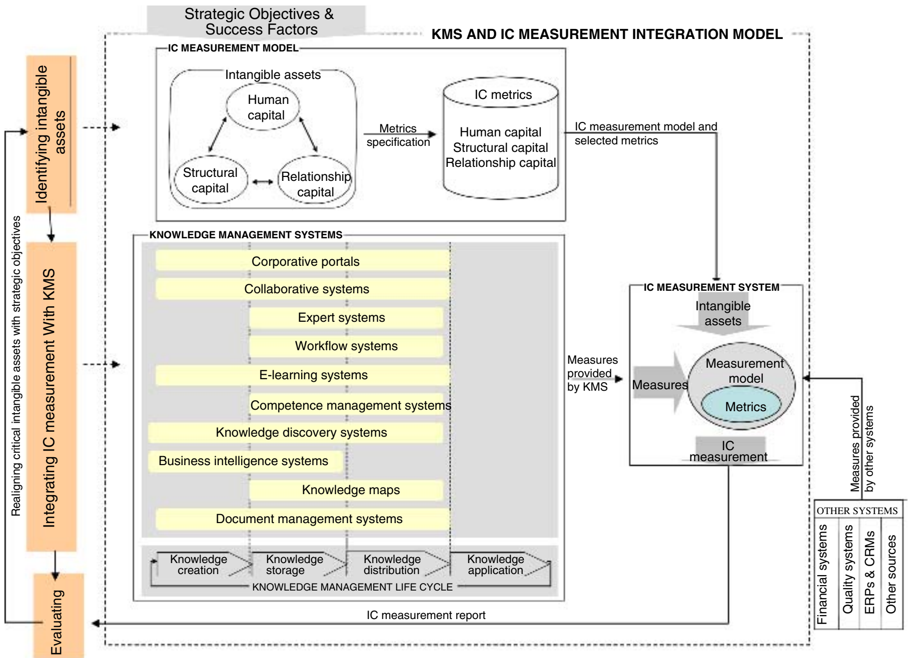

Fig. 1. A model for linking KMS and IC measurement

5.1 The IC Measurement Model Component

The aim of this component comprises the identification of the intangible assets that bring competitive advantage and assure the development of the organization core competences. It also comprises the specification of appropriate metrics in order to assess these intangible assets.

The IC measurement model component starts with the identification of the mission and the corporate strategic objectives, allowing the specification of business drivers, critical success factors and the intangible assets that can create wealth to the organization. The purpose of linking IC with the strategic objectives is to ensure that the organization gets competitive advantages from its IC and KMS usage [42]. The intangible assets can be grouped in different categories, according their own characteristics: human capital, structural capital and relationship capital.

This component can be implemented through one of the reviewed IC measurement models, like for instance the *Skandia Navigator*, *Balanced Scorecard, Intangible*

Assets Monitor, Intelect Model, or other. It is important to note that the model outlined in this paper is independent of the IC measurement model adopted, in order of not reduce their applicability.

5.2 The Knowledge Management Systems Component

The aim of this component is to provide a set of measures, useful to quantify the IC metrics specified in the first component of the model. KMS can offer a relevant contribute to the IC improvement, by supporting and enhancing processes related with organizational knowledge crucial to the organization [35]. However, this contribute can only be achieved if KMS are focused on the intangible assets that bring value added and competitive advantage to organizations, according their strategic objectives [42]. From this perspective, the KMS can contribute to quantify a set of metrics useful to the IC measurement through the knowledge wrapped in these systems, linking in this way the KMS and the IC measurement. This component should also facilitate the identification of the most appropriate KMS categories, according to the organization needs and strategy, and aligning the KMS selection with the intangible assets that bring competitive advantage.

This component is described in Figure 1 through a structure that represents the role performed by the different KMS categories, according their own characteristics and functionalities, in supporting KM processes, namely knowledge creation, storage, distribution and application [35]. This relationship is based on the assumption that KMS, as technological systems focused on managing the organizational knowledge, do not support knowledge application; in fact, only people are able to apply knowledge. However, the KMS could facilitate the development of an environment that enables the knowledge usage and application, from the organization people.

5.3 The IC Measurement System Component

The aim of this component is to support the IC measurement, according to the model and the metrics selected in the first component. The IC measurement system component uses a set of measures provided by the different KMS categories to quantify the specified IC metrics. It can also use some measures provided by other sources to complement the IC measurement, like for instance financial applications, quality systems or ERPs, Enterprise Resource Planning Systems.

This component establishes a linkage between the IC measurement model and the KMS, showing the contribution of the different KMS categories to the intangible assets development. The outcome of the IC measurement system consists on a report that facilitates the evaluation of intangible assets and it is a starting point to identify weaknesses and strengths in terms of organizational knowledge development. Evaluating the intangible assets can facilitate the redefinition and realignment of business drivers, success factor and intangible assets that bring competitive advantages to the organization.

5.4 Measures Provided by KMS to Quantify IC Metrics

Each organization tries to select the most appropriate metrics to measure the intangible assets worthiness and the KM initiatives, according to their needs and their strategic objectives [39]. A wide variety of metrics are developed as an attempt to recognize and evaluate the value of intangible assets. From the literature review on IC measurement models and the survey conducted to the Portuguese organizations, it was possible to systematize the metrics more pursued in the IC measurement [36].

Considering the different KMS categories referred in this paper and their addressed issues and main functionalities, one can argue that a large number of these IC metrics could be quantified through KMS, i.e., through the knowledge wrapped in the KMS. Thus, to clarify the potential contribution of these systems to the IC measurement, a significant number of KMS were analysed, from different categories and suppliers. As a result of this study, Table 4 summarizes a set of measures that could be provided by the different KMS categories to quantify the IC metrics, clarifying the contribution of the KMS to the IC measurement.

It is also important to note that these measures could be complemented with others provided by other sources, like for instance financial systems, ERPs, CRMs, as well as survey questionnaires, as is illustrated in Figure 1.

Table 4. Measures provided to quantify IC metrics through KMS

KMS	Measures provided by KMS to quantify IC metrics
Document management systems	Accesses to organizational knowledge base (#) Contributions to organizational knowledge base (#) Time (average) to request Rate of knowledge accessed/reutilized
Knowledge maps	Accesses to knowledge maps (#) Contributions to knowledge maps (#) Rate of knowledge accessed/reutilized
Collaboration systems	Projects in collaboration with external entities (#) Projects in collaborations with other workgroups (#) Rate of best practices diffusion Questions reported in forums (#)
Workflow systems	Processes completed without errors (#) Processes in compliance with manuals (#) Processes upgraded (#) % of processes completed in time % of automated business processes
Business intelligence systems	New business opportunities (#) Rate of investments in new markets Market share in the segment Geographic customers distribution % of new customers/customers lost Rate of sales to new markets Rate of sales to new customers Customer satisfaction index Employees satisfaction index

Table 4. *(continued)*

KMS	Measures provided by KMS to quantify IC metrics
Expert systems	Experts with specialization degree (#) Managers with specialization degree (#) FAQs accesses (#) Contributions from experts: rolls, best practices, advices, suggestions (#) Rate of expert knowledge accessed/reutilized % of new experts
Competence management systems	Average time in training per employee (days per year) Cost per capita in training programs Average years of service in organization Average age of employees % of employees by group age % of female and man employees Rate of absenteeism Rate of employees rotation Employees satisfaction index % employees with advanced degrees
E-learning systems	E-learning training programs (#) Hours spent in e-learning programs (#) % employees that complete with success e-learning training programs Employees with specialization based on e-learning training programs (#)
Knowledge discovery systems	New patents (#) Patents in registration (#) Years average of registered patents Rate of knowledge reutilized in new contexts New ideas to upgrade products, services or processes (#) New products, services or processes generated by innovation processes (#) Certified processes (#)
Corporative portals	Rate of knowledge distributed/applied

6 Model Validation

The model proposed in this paper has been evaluated by an expert panel, in order to test is validity. The point of view and the suggestions provided by the expert community can also help to improve the model in such aspects.

6.1 Method

Using a questionnaire, the expert panel was invited to analyse such aspects as the model completeness, coherence and comprehensiveness. The expert panel was also asked to clarify if they agree with the objects defined in the model and the suggested metrics that can be quantified through the different KMS categories. The point of view expressed by an expert community can contribute to the model validation and can also generate new ideas and new perspectives about the proposed model, contributing to the model improvement.

A questionnaire survey was sent to forty experts including researches, practitioners and consulters in the KM and KMS fields, requesting them to express their point of view about the proposed model. The questionnaire was structured in two sections:

i) the first section with nine questions using a five-point scale (totally, much, little, nothing, don't know) with the aim of evaluate the model from different perspectives; ii) the second section asking the expert panel about suggestions that could be interesting to improve the model. The questionnaire was sent by email with a synthesis describing the aim of the work, the proposed model and its objects. Two follow-ups were carried with a delay of one month between them. Fourteen valid answers were received, corresponding to a response rate of 35%.

6.2 Results Discussion

From the fourteen responses received only one was not complete, answering exclusively to the second section about suggestions to improve the model. Table 5 summarizes the questions addressed in the questionnaire and presents the occurrence rate addressed in the five-point scale for each question.

Table 5. A synthesis of the questionnaire results

Questions \ Response options	Totally	Much	Little	Nothing	Don't know
1. Do you consider the model useful?	46%	54%	0%	0%	0%
2. Do you consider the model comprehensive?	15%	77%	8%	0%	0%
3. Do you consider the model complete?	0%	50%	33%	0%	17%
4. Do you agree with the model components?	8%	84%	8%	0%	0%
5. Do you consider the model structure coherent?	8%	92%	0%	0%	0%
6. Do you agree with the objects defined in the *intellectual capital measurement model* component?	31%	61%	8%	0%	0%
7. Do you agree with the objects defined in the *knowledge management systems* component?	15%	85%	0%	0%	0%
8. Do you agree with the objects defined in the *intellectual capital measurement system* component?	8%	84%	8%	0%	0%
9. Do you agree with the suggested metrics and do you think it is possible to get them from KMS?	8%	76%	8%	0%	8%

Based on the above statistical results, it is possible to conclude that all of the respondent experts consider the model totally or much useful. The proposed model is totally or much comprehensive for 92% of respondents, and it is complete for 50% of inquired experts. Almost all expert panel (92%) agree totally or much with the components defined in the model and all of them consider the model structure coherent. It is also possible to conclude that almost all respondent experts (92%) agree totally or much with the objects defined in the intellectual capital measurement model; the same result was obtained when we asked about the objects defined in the intellectual capital measurement system. On the other hand, all of the expert panel agree (totally or much) with the objects defined in the knowledge management systems component. Finally, 84% of the expert panel agree (totally or much) with the suggested measures that KMS could provide to quantify the IC metrics.

The results obtained from the questionnaire show that the expert community considers the model useful, comprehensive and agree with the defined components. It is important to note that all respondents consider the model totally or much useful, which clearly indicates the utility and the relevance of this research field. The panel also agrees with the objects defined in each one of the model components. The most critical aspect focused from the expert community, in the questionnaire, was the completeness of the model. 33% of respondents have considered that the model was little complete and 17% referred that they don't know.

The second section of the questionnaire, inviting the expert community to express some suggestions to improve the model, has produced some results, described in Table 6.

Table 6. Summary of suggestions expressed by the expert community

Suggestions
In addition to the comprehensive list of metrics, I would also consider softer "percentual survey" measures that are critical to KM, such as the employees trust, commitment and ability to share knowledge.
The relationship between the KMS categories and the KM processes supported by each one of them is not very clear.
It is difficult to establish a strong relationship between the metrics and the KMS quality/effectiveness. But it is a starting point and such metrics can be refined over time.
It is important to take into consideration some important issues, such as: the need to "close the loop" through evaluation, the subjective assessment of the KMS and cultural factors, like transparency and trust.
The metrics suggested are, almost all, activity metrics and not result metrics.
It is difficult to obtain some of the suggested metrics.

Analysing these suggestions, it is possible to draw some comments:

- Almost all suggestions are related with the metrics suggested as able to be quantified through KMS. In fact, the metrics are simply suggestions and they can be redefined according to the organization context, needs and strategic objectives.
- The relationship between the KMS categories and the KM processes is not very clear probably because the synthesis sent with the questionnaire is very short. A more complete description was made in some articles [35], [36].
- The assessment of some important issues, like trust, transparency and cultural factors is important but very difficult and normally it isn't supported by KMS.

7 Conclusion

The complementariness between KMS and IC measurement is largely recognized, enforcing the necessity of fill the existing gap between them. The model presented in this paper is an attempt to fill this gap, showing the contribution of KMS to the IC measurement. The model establishes a relationship between the intangible assets that bring competitive advantage and the KMS that can support the development of these assets, i.e., it facilitates the selection of the most appropriate KMS according the organization needs and strategy. On the other hand, it enables a more automated and systematic measurement of intangible assets through measures that can be provided by KMS or other systems to quantify the selected metrics.

Considering that the purposed model shows the impact of KMS in the value creation and their benefits to the organization, it contributes to evaluate the success or failure of KMS initiatives. In fact, many projects are viewed as failures or abandoned as a result of the difficulty in measuring their benefits.

Future work will focuses on the model validation through case studies.

References

1. Alavi, M. and D. Leidner: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. MIS Quarterly, Vol. 25, Nº 1 (2001) 107-136
2. Allee, V.: The Value Evolution. Journal of Intellectual Capital, Vol. 1, Nº 1 (2000) 17-30
3. Allee, V.: "An Emerging Perspective of Wealth and Value. Research Paper", www.vernaallee.com/secondarypages/intangiblevalueframework.htm, accessed in 2004/6/12
4. Anderson and McLean: "Total Value Creation", www.totalvaluecreation.com, accessed in 2004/7/3
5. Andriessen and Tissen: Weighless Weight - Find your real value in a future of intangible assets. Pearson Education London (2000)
6. Baroni, R.: Aplicações de Softwares de Gestão do Conhecimento: tipologia e usos. UFMG University (2000)
7. Bontis, N.: There´s a price to on your head: managing intellectual capital strategically. Business Quarterly, Summer (1996)
8. Bontis, N., W. Keow, et al.: Intellectual capital and business performance in Malaysian industries. Journal of Intellectual Capital, Vol. 1, Nº 1 (2000) 85-100
9. Brooking, A.: Intellectual Capital: Core Assets for the Third Millennium Enterprise. Thomson Business Press, London, United Kingdom (1996)
10. Camisón, Palácios, et al.: "Modelo Nova", www.gestiondelconocimiento.com, accessed in 2004/5/27
11. Chen, Z. e. X.: Measuring Intellectual Capital: a new model and empirical study. Journal of Intellectual Capital, Vol. 5, Nº 1 (2004)195-212
12. Davenport, T. and L. Prusak: Working Knowledge: How Organizations Manage What They Know. Harvard Business School Press, Boston (1998)
13. Edvinsson, L.: "IC Rating", www.intellectualcapital.se, accessed in 2004/5/19
14. Edvinsson, L. and M. Malone: Intellectual Capital: Realizing your Company's True Value by Finding its Hidden Brainpower. Harper Business, New York (1997)
15. Edvinsson, L., J. Roos, et al.: Intellectual Capital: Navigating in the New Business Landscape. Mcmillan, New York (1998)
16. Euroforum: "Proyecto Intelect. Medición del Capital Intelectual", www.euroforum.es/intelect/modelo_intelec.htm, accessed in 2005/2/7
17. European KM, F.: "KM Technologies and Tools IST Project Nº 2000-26393", www.knowledgeboard.com/library/ekmf_framework_technologiesandtools.pdf, accessed in 2004-7-14
18. Heng, M.: Mapping intellectual capital in a small manufacturing enterprise. Journal of Intellectual Capital, Vol. 2, Nº 1 (2001) 53-60
19. IADE and CIC: "Modelo Intellectus: Medición e Gestión del Capital Intelectual. Documento Intellectus nº 5", www.iade.org, accessed in 2004/7/14.
20. Jackson, C.: "Process to Product: Creating Tools for Knowledge Management", www.brint.com/members/online/120205/jackson/, accessed in 4-11-2004

21. Joia, L.: Measuring Intangible Corporate Assets. Journal of Intellectual Capital, Vol. 1, Nº 1 (2000) 68-84
22. Kankanhalli, A. and B. Tan: A Review of Metrics for Knowledge Management Systems and Knowledge Management Initiatives. In: Proceedings of the 37º Hawaii International Conference on System Sciences, Hawaii, (2004)
23. Kaplan, R. and D. Norton: The Balanced Scorecard: translating strategy into action. Harvard Business School Press, Boston, Massachusetts, USA (1997)
24. Leliaert, Candries, et al.: Identifying and managing IC: a new classification. Journal of Intellectual Capital, Vol. 4, Nº 2 (2003) 202-214
25. Lev, B.: Intangibles: Management, Measurement and Reporting. Brookings Institute Press, Washington DC (2002)
26. Liebowitz, J. and C. Suen: Developing knowledge management metrics for measuring intellectual capital. Journal of Intellectual Capital, Vol. 1, Nº 1 (2000) 54-67
27. Lindvall, M., I. Rus, et al.: Software systems support for knowledge management. Journal of Knowledge Management, Vol. 7, Nº 5 (2003) 137-150
28. Lopes, I. and M. Martins: The New Business Models in the Knowledge Economy: the Strategic Way to Value Creation. The Electronic Journal of Knowledge Management, Vol. 4 (2006) 159-168
29. McPherson and Pike: Accounting, empirical measurement and intellectual capital. Journal of Intellectual Capital, Vol. 2, Nº 3 (2001) 246-260
30. Martí, J.: In Search of an Intellectual Capital General Theory. Electronic Journal of Knowledge Management, Vol. 1, Issue 2 (2003) 213-226
31. MERITUM: "Meritum Project's - Guidelines for Managing and Reporting on Intangibles", www.fek.su.se/home/bic/meritum, accessed in 2004/7/3
32. Mouritzen, J., P. Bukh, et al.: "Intellectual Capital Statements - The New Guideline", www.videnskabsministeriet.dk/fsk/publ/2003/guideline_uk/html/1_0, accessed in 2004/7/23
33. Nantel, R.: Knowledge Management Tools and Technology 2004: 35 Systems to Maximize Your Organization's Intellectual and Human Capital. Brandon- hall.com, (2003)
34. Pike, S. and G. Roos: Mathematics and Modern Business Management. In:25th McMaster World Congress Managing Intellectual Capital, Hamilton, Ontário, Canadá, (2004)
35. Pinto, M., A. Lopes, et al.: A Framework for Characterizing Knowledge Management Systems. 6th European Conference on Knowledge Management, Limerick, Ireland (2005) 442-450
36. Pinto, M., F. Lopes, et al.: Knowledge Management Systems and Intellectual Capital Measurement. Proceedings of XVI ISPIM Annual Conference, Porto, Portugal, (2005)
37. Pullic, A.: "VAIC - An Accounting Tool for IC Management", www.measuring-ip.at/papers/ham99txt.htm, accessed in 2004/7/3
38. Roos, J., N. Dragonetti, et al.: Intellectual Capital: Navigating in the New Business Landscape. Macmillan, New York (1997)
39. Smith, H. and J. McKeen: Developments in Practice XVII: A Framework for KM Evaluation. Communications of the Association of Information Systems (AIS), Vol. 16 (2005) 233-246
40. Smith, P.: Systemic Knowledge Management: Managing Organizational Assets for Competitive Advantage. Journal of Knowledge Management Practice, Vol. 1 (1998)
41. Sveiby, K.: "Intellectual Capital and Knowledge Management. Research paper", www.sveiby.com/articles/IntellectualCapital.htm, accessed in 2005/4/3
42. Zhou, A. and D. Fink: The Intellectual Capital Web: a systematic linking of intellectual capital and knowledge management. Journal of Intellectual Capital, Vol.4,Nº 1 (2003) 34-48

Synergizing Standard and Ad-Hoc Processes

Andreas S. Rath[1], Mark Kröll[1], Keith Andrews[3], Stefanie Lindstaedt[1], Michael Granitzer[1], and Klaus Tochtermann[1,2]

[1] Know-Center Graz
Inffeldgasse 21a/II, 8010 Graz, Austria
{arath, mkroell, slind, mgrani, ktochter}@know-center.at
[2] Knowledge Management Institute
Graz University of Technology
Inffeldgasse 21a/II, 8010 Graz, Austria
[3] Institute for Information Systems and Computer Media (IICM)
Graz University of Technology
Inffeldgasse 16c, 8010 Graz, Austria
kandrews@iicm.edu

Abstract. In a knowledge-intensive business environment, knowledge workers perform their tasks in highly creative ways. This essential freedom required by knowledge workers often conflicts with their organization's need for standardization, control, and transparency. Within this context, the research project DYONIPOS aims to mitigate this contradiction by supporting the process engineer with insights into the process executer's working behavior. These insights constitute the basis for balanced process modeling. DYONIPOS provides a process engineer support environment with advanced process modeling services, such as process visualization, standard process validation, and ad-hoc process analysis and optimization services.

Keywords: process modeling, knowledge utilization, ad-hoc process mining, process engineer services, knowledge capturing, process visualization.

1 Introduction

In a rapidly changing world, organizations of all kinds strive for standardization, control, transparency, and quality assurance. Workflow Management Systems (WFMS) have become quite widespread to support the progress and development of organizations. It is generally accepted that these systems have made a significant contribution to increased productivity [8].

The key discriminating feature of WFMSs is the flexibility they provide to deal with changes [17]. This adaptability is especially required in a knowledge-intensive business environment, where workers perform their work in a creative way (as opposed to a routine way). Knowledge workers are guided by goals instead of tasks and prefer significant freedom in structuring their own activities [12]. However, by allowing knowledge workers freedom for creativity, organizations may decrease the possibility to standardize and control working procedures. This results in a

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 267–278, 2006.

dilemma of contradictory ambitions, between the organizational need for standardization on the one hand and the essential freedom needed by knowledge workers on the other hand.

In the middle of this dilemma are the process engineers, who are the representatives of the organizational needs. Their task is to model the processes of organizations and to refine them if changes are required. In fulfilling their task they face following challenges (amongst others):

– *Information Gathering*
  To model a process the *process engineers* need information about what kind of work is done and how work is done in the organization. A popular technique for collecting this information is interviewing key persons in the organization, such as project managers, regional directors, and team leaders. Data from WFMSs or pre-existing *how to* documents (for example, how to write a requirements document, how to start a project, how to organize a meeting, etc.) can be used as an information source. A typical problem which arises is that many varying descriptions of the same workflow are collected. The challenge for the process engineer is to identify the best information sources for retrieving workflow information.
– *Large Amount of Information*
  Extensive collection of information from various sources is a good starting point for the process analysis step. In this step the collected information is aggregated and possible process descriptions are elaborated. The challenge for the process engineer is to extract process descriptions based on the collected information.
– *Defining Standard Processes*
  Standard processes can be modeled in various process description languages such as BPML (Business Process Modeling Language), BPEL (Business Process Execution Language), UML (Unified Modeling Language), and XPDL (XML Process Description Language) and with various applications such as Microsoft Visio, Microsoft PowerPoint, ERP (Enterprise Resource Planning) Systems, and process modeling tools built into WFMSs. The challenge is to decide which modeling language to use for the process descriptions, because the descision has consequences for the further usage of the process descriptions in specific WFMS.
– *Process Change Detection*
  Since the business environment changes continuously, organizations are requested to continually adapt their processes to new conditions. In addition to process changes arising from external factors, there are natural process deviations. Natural deviations from standard processes are also referred to as *ad-hoc processes*. These deviations can be a shortening, an extending, or more generally a structural change to a standard process. Ad-hoc processes happen when standard processes are not performed in the intended way. There can be several reasons for a deviation from a standard process, for example it is easier, more comfortable, or more efficient. The challenge for

process engineers is to detect if and where deviations from standard processes occur and to adapt standard processes if necessary.

– *Ad-Hoc Process Capture*
  Ad-hoc processes happen steadily in organizations. The first step for process engineers is detecting in which element of the process chain deviations occur. The next step is to capture the composition of the ad-hoc process itself. The information about the ad-hoc process can then be used in a refinement step by the process engineer. If an ad-hoc process outperforms the existing standard process (for example, in terms of time efficiency, information flow, or resource flow) the standard process can be modified accordingly or replaced by the the ad-hoc process.
– *Enhancing Ad-Hoc Processes to Standard Processes*
  Ad-hoc processes are not necessarily deviations from standard processes. They can also arise from the composition of new tasks carried out by process executers. Information captured about ad-hoc processes can act as a valuable base for the process engineer when modeling new standard processes. Sometimes, an ad-hoc process can advance to become modeled as a standard process.
– *Modeling Knowledge-Intensive Work*
  Knowledge work is described as work with a large amount of creative activity [12]. Creative activities are very hard to model in advance by a process engineer. Since knowledge workers often reuse their existing knowledge to manage the complexity of their work [2], reuse patterns and resources can be used as a starting point to model knowledge-intensive work. A common method of reuse is called *templating*, where past processes and their resources are used as templates for the knowledge worker's current work [2].

These challenges lead us to the objective of this paper, which is the presentation of the process engineer support environment in the DYONIPOS (DYnamic ONtology based Integrated Process OptimiSation) research project. DYONIPOS aims to support the two crucial roles in a knowledge-intensive organization, the process executer and the process engineer, by synergizing the organizational need for standardization, control, and transparency (standard processes) with the essential need for creative freedom for knowledge workers. The approach of DYONIPOS incorporates the development of solutions based on automatic and semi-automatic knowledge management methods and technologies such as knowledge discovery, semantic systems, knowledge flow analysis, and process visualization. For a comprehensive overview of the DYONIPOS project see [12].

This paper is structured as follows: Section 2 discusses top-down and bottom-up approaches to the challenges process engineers face in process modeling. A comparison of these approaches motivates the need for a new approach. The hybrid approach of DYONIPOS is presented in Section 3. Section 4 discusses the event, task, and process model (semantic pyramid) used by DYONIPOS. Section 5 outlines the support services DYONIPOS provides for the process engineer.

2 Top-Down and Bottom-Up Approaches

The common process modeling approach, where processes are modeled manually based on available process data or information, is called the top-down approach. Data and information are usually obtained from interviews, existing WFMSs, observations during site visits, document inspection, or (if available) previous process descriptions. The various information sources and the retrieved data need to be structured and aligned manually by the process engineer. Based on the analysis of the collected information the process engineer models the processes. The choice of process modeling language and process description is based on the intended further usage of the process model.

Supplying a specific WFMS with the process model is usually the next step after modeling, validation, and refinement of processes. WFMSs have become quite popular for managing complex organizational processes, but fail in supporting knowledge-intensive and agile processes [9]. The problem with this kind of process is that they cannot be modeled in advance. Further problems of WFMSs are their limited ability to deal with dynamic changes [15] to the implemented, static process models. Weakly-structured workflows address this insufficiency by suggesting lazy and late modeling or interleaving process modeling with process execution [16].

Detection of process changes is limited in standard WFMS, because refinement and deviations of standard workflows are usually not allowed and hence no workflow logs about the deviation exist. If process engineers want to validate the actuality of existing processes, a new round of time-consuming and costly information gathering has to be initiated. For the process engineer, it is also unclear when a process refinement has to take place, because in a WFMS there are no indicators for a process change.

The contrasting approach to process modeling is the bottom-up approach, which means that the information originates from process executers instead of process engineers [7]. The bottom-up approach is also referred to as process mining [4,14,18]. In this approach, the process model can be derived from workflow, task, and/or event logs. In order to transform the monitored data stored in the logs into tasks, information retrieval, mining and monitoring techniques, and advanced algorithms [13] are needed. The advantages of this approach are the intensive data and information gathering possibilities and the continuous refinement and enhancement of the calculated processes as the number of cases increases.

Event log mining [4] has the advantage of providing fine-grained data to the mining step in comparison to [13] where tasks from workflow logs are used as a basis. Event logs incorporate data about the executions of standard and ad-hoc processes and hence event log mining considers both types of processes when calculating the process model. On the other hand, there is no differentiation between standard and ad-hoc processes and hence a change in or a deviation from standard processes can not be detected, which is the same problem as in the top-down approach. Since remodeling, i.e. a recalculation of the process model, is done automatically, the generation of a new process model can be done easily.

Table 1. A comparison of the top-down and bottom-up approaches to modeling business processes

Challenges	**Top-Down**	**Bottom-Up**
Information gathering	Interviews, document sighting, organization visits, WFMS, old process descriptions.	WFMS logs, event and task logs.
Large amount of information	Manual extraction of processes	Algorithm based process creation based on logs.
Defining standard processes	Process engineers select appropriate tool and process description language (PDL) based on further usage requirements.	Only algorithm supported PDLs, automatic generation of a suggested process description
Process change detection	Only based on manual observation, new information gathering step required.	No comparison of standard and ad-hoc processes, i.e. no change detection.
Ad-hoc process capturing	Only if directly observed by process engineer.	Completely stored in event logs.
Enhancing ad-hoc processes to std. processes	No data available.	Suggestion for new standard processes based on a recalculation of the process model.
Knowledge-intensive work modeling	Limited by process engineers inspections.	Uses templates from event logs, limited when using WFMS logs.

Taken individually, neither the top-down nor the bottom-up approaches provide an adequate solution to the challenges a process engineer faces in modeling processes. A comparison of the top-down and bottom-up approach is given in Table 1.

3 The DYONIPOS Project

The DYONIPOS (DYnamic ONtology based Integrated Process OptimiSation) project strives to ameliorate the dilemma of the organizational need for standardization and control on the one hand and the day to day creative freedom needed by a knowledge worker on the other hand. The research project DYONIPOS addresses this dilemma by following a hybrid approach, i.e. a combination of bottom-up and top-down approach. The left part of Figure 1 shows the hybrid approach of DYONIPOS. The inductive, bottom-up approach uses monitored interactions of the process executer for further analysis. The top-down approach in DYONIPOS is represented by the consideration of standard processes, for example those originating from WFMSs.

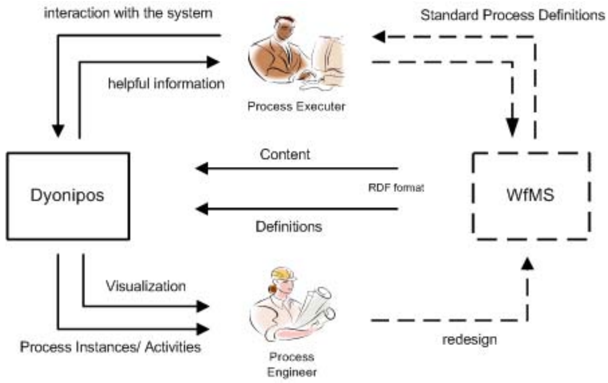

Fig. 1. The scope of the DYONIPOS project

In the business process environment, DYONIPOS will support both the process executer and the process engineer. Process executers are provided with support to find, perform, and record ad hoc processes within their work environment such that the ad hoc process retrieval, application, and definition take place within the executer's current work context. Hence, based on recent interaction with the system, guidance is featured through the daily work which is enhanced by providing supportive resources. Process engineers will be supported in reviewing and analyzing recorded ad-hoc processes. Standard processes can be validated by analyzing process instance frequencies and compared to newly created ad-hoc processes. If ad-hoc processes outperform standardized processes DYONIPOS provides means to enhance ad-hoc to standard processes.

4 Semantic Pyramid

In this section we introduce the semantic pyramid which describes the continuous evolution of contextual information through the different, semantic layers. The pyramid is illustrated in Figure 2 starting at the bottom with events that are executed by one knowledge worker and ending with processes where many knowledge workers can be involved. Each level of granularity (events, event blocks, tasks and processes) provides a different representation of the data regarding the semantic quality. By passing through the layers and ending at the process level the semantic quality is permanently enhanced. In the following it is described how the information gathering concerning each individual layer is carried out.

The steps required to collect information about the process executer's actions and thus obtain the context, start with the recording of all events, i.e. the entire user interaction. Events belonging to a logical unit are grouped together into event blocks. Event blocks form semantic sets and are eventually assigned to the knowledge worker's tasks. Hence, a task is represented as a sequence of event blocks. A task is modeled as a large graph containing event blocks as nodes.

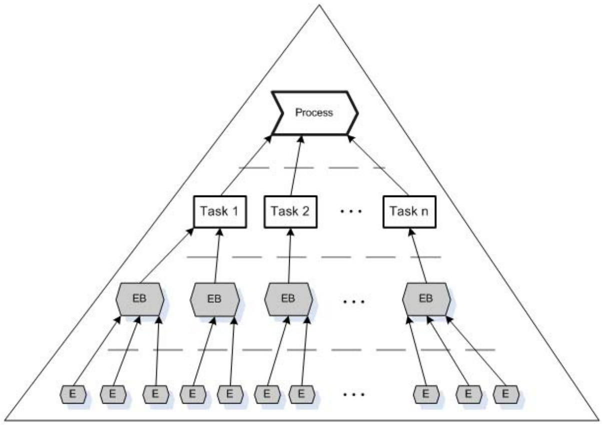

Fig. 2. The semantic pyramid comprises four layers: *events*, *event blocks*, *tasks* and *processes*. Starting from the bottom, each level of the semantic pyramid is obtained by sensibly grouping together elements from the preceding layer. Moving upwards through the layers, the semantic quality is continually enhanced.

Each level of granularity (events, event blocks, tasks, and processes) provides a different representation of the data in terms of semantic quality (see Figure 2). Moving through the layers from bottom to top, semantic quality is continually enhanced.

4.1 Events

The data on which DYONIPOS operates consists of the monitored interactions between the user and the computer. Key logging software, referred to as the *event logger*, records all *events* which occur on the user's computer. Events are user inputs, such as mouse movements, mouse clicks, starting a program, creating a folder, or opening a file. A similar approach is described by Fenstermacher [4]. All recorded events are stored in the *event log*. To ensure security and privacy, the user has the ability to modify the event log and to delete events.

Our work in this area builds upon the results of the MISTRAL project [11], which aims to extract semantic concepts from text, audio, and video data. It is even conceivable to incorporate talks amongst knowledge workers into the DYONIPOS project in order to enrich the individual user profile. The *TaskTracer* project [2] follows a similar approach, where telephone conversations are recorded and further processed by means of speech to text applications.

4.2 Event Blocks

The typical knowledge worker produces a considerable amount of data in the course of a typical day's work. Since the event logger monitors fairly low-level events, a huge amount of data is recorded. To cope with the sheer amount of data, three strategies are used: *filtering*, *relation analysis* and *aggregation*. Filtering involves removing irrelevant data from the event log. Relation analysis deals with finding dependances and similarities. Aggregation involves the grouping of sequences of events in the event log into *event blocks*.

Event blocks are formed using predefined static rules, which map a set of events to an event block. For example: The user moves the mouse over a program icon, double clicks the icon, and the system starts a program. This set of events can be combined to an event block called *starting a program*. An interesting open question here is to what extent it is possible to automatically find a mapping based on the data in the event log, i.e. automatically generating mapping rules.

Since not all events and consequently not all event blocks of a knowledge worker's daily work can be captured automatically, the user has the ability to manually add event blocks. Event blocks of this kind might be a meeting appointment, a conversation with a colleague, or signing a report.

The knowledge worker's privacy is ensured by law. Thus, a natural dilemma arises when trying to gather as much information as possible about a worker's interactions with the system while staying within the law. No user interaction remains hidden from the system. Hence, any data to be stored needs explicit permission from the user. Moreover, event blocks are transfered into an abstract form containing the essential data in encrypted form. The level of encryption is tunable and could involve term vector representation or hash coding.

4.3 Tasks

Event blocks are combined into *tasks* by grouping together similar event blocks into semantic sets. Thus one resulting set represents one task of the process executer. However, the knowledge worker is not bound to remain within a given task from start to finish. Switching between tasks might be necessary or be more efficient. In other words, event blocks may be interleaved and have to be grouped together according to their topic. By analyzing the event blocks and which documents were written or read, event blocks exhibiting similar content are identified. The degree of similarity indicates the affiliation to certain event block sets. Standard text mining algorithms provide us with the means to extract keywords and compare textual contents. A high-quality semantic description of the process executer's tasks is thus obtained.

In DYONIPOS we focus on the knowledge worker's (user's) context. The user context describes who the user is (*organizational context*), what the user does (*work context*), how the user does it (*behavioral context*), with whom the user collaborates (*social context*), and which resources the user uses (*document context*). Further contexts addressed in DYONIPOS are the *process context*, describing the position of the knowledge worker within a business process, and the *environmental context*, capturing the location of the knowledge worker (for example, computer desktop, meeting room, or corridor).

All these different contexts are used to provide highly supportive information for the knowledge worker. A further application of the contextual information is to identify different and similar tasks. The idea here is to analyze the user's context for context switches, which may indicate switching from one task to another. Context switches could potentially be used as indicators for an update of any supportive information. Since contextual retrieval is application specific [5], further research has to be done in applying contextual retrieval in the area of knowledge-intensive business environments.

4.4 Processes

A *process* is formed by aggregating tasks performed by a number of different knowledge workers using process mining. Instead of only one person, several persons performing well-defined steps within the process are involved. The different contexts described in Section 4.3 are merged, allowing crucial insight into the semantics of business processes. Based on the derived information, DYONIPOS provides a number of services to support the knowledge worker.

5 Process Engineer Support

In this section the services DYONIPOS provides for the process engineer are introduced. DYONIPOS visualizes both ad-hoc and standard processes and enables the process engineer to review and analyze recognized ad-hoc processes and compare them to pre-defined standard processes. Process visualization serves as crucial preprocessing step for further analysis and optimization. Provided sevices are listed in the corresponding categories.

5.1 Visualization

Both ad-hoc and standard processes are modeled in DYONIPOS as RDF graphs. These graphs can be visualized using standard techniques from the field of graph drawing [1], such as layered Sugiyama-style graph drawing [10] and force-directed placement [3]. For DYONIPOS, the most recent versions of these algorithms will be used [6]. Processes and subprocesses can be displayed at different levels of granularity. Hence, the knowledge worker is not bound to be an expert on the whole process: it is possible to select only that part of the process within the worker's scope of knowledge. Color coding can be used to distinguish between old and new, and between standardized and ad-hoc processes.

The frequency of paths traversed by process executors is denoted by the thickness of lines connecting parts of processes. Well-beaten paths can be easily identified, simplifying the task of selecting processes for further analysis. A further module will allow visual comparison of two graphs, for example an ad-hoc process of interest and a similar standardized process.

– *Multiple Levels of Granularity*
  Entire processes and parts of processes, i.e. subprocesses, can be displayed individually allowing different views of granularity.
– *Frequent Process Paths*
  Many newly created processes occur only with low frequency, in other words, processes that can be disregarded. However, the process engineer is interested in ad-hoc processes that happen more frequently, hence, representing

a potential alternative to standard processes. Exactly these processes can be visualized. Beaten paths can be easily identified thus lightening the task of selecting processes for further analysis.

5.2 Analysis

The analysis of subprocesses and entire processes is empowered by the corresponding visualization. The process view can be adapted according to the frequency with which ad-hoc processes occur. Low frequency processes are usually not taken into account by the process engineer and are faded out. The ad-hoc processes in question can thus be easily detected and can be partially or entirely compared to traditional processes. DYONIPOS enables the process engineer to simulate standardized and ad-hoc processes by means of petri nets. Knowledge flows can be pursued, critical paths can be identified, and time delays are indicated. It is possible to identify inefficient procedures such as processes of varying duration which perform the same task. Bottlenecks can be detected by looking for parallel sequences interrupted by sequential steps.

- *Similarity Measuring*
  Within this service two processes can be compared to each other. Different levels of granularity, i.e., which subprocesses or tasks they have in common, allow insights into levels of the organizational structures.
- *Standard Process Validation*
  Standard processes are validated by comparing them to new, ad-hoc processes. The degree of accordance of frequent process paths could serve as an indicator for the acceptance of the standard process amongst the employees. Standard processes that have few paths in common are predestined to be further analysed.

5.3 Optimization

Once inefficient procedures have been detected, the process engineer can take steps to improve them. The most efficient process can be used as a model for a new standardized process. Bottlenecks can be defused by replacing sequential steps by parallel ones.

- *Process Deviation Analysis*
  Changes in standard process flows can be detected. These deviations can be analyzed regarding occurrence frequency, time efficiency and the order of tasks. The usage of the analysis information can lead to the defusion of bottlenecks by replacing sequential steps in a process by parallel ones and to an increase concerning time efficiency by comparing various deviations. Hence, the corresponding standard process can be enhanced.
- *Suggesting Processes for Standardization*
  Ad-hoc processes can be analyzed and compared to standardized processes by making usage of above mentioned services. Such a newly created process is suggested for standardization if it accurately defines an organizational procedure so far neglected or it it outperforms an existing traditional process.

6 Concluding Remarks

In this paper we presented the Process Engineer Support Environment of the research project DYONIPOS. In this context we discussed the DYONIPOS approach to the challenges a process engineer faces and the supported services. Currently we are exploring event logging software applications, graph mining techniques and semantic technologies. We are planning to investigate task and process mining techniques.

Acknowledgements

The project results have been developed in the DYONIPOS project (DYnamic ONtology based Integrated Process OptimiSation). DYONIPOS is financed by the Austrian Research Promotion Agency (http://www.ffg.at) within the strategic objective FIT-IT under the project contract number 810804/9338.

The Know-Center is funded by the Austrian Competence Center program Kplus under the auspices of the Austrian Ministry of Transport, Innovation and Technology (http://www.ffg.at), by the State of Styria and by the City of Graz.

References

1. Giuseppe di Battista, Peter Eades, Roberto Tamassia, and Ioannis G. Tollis. *Graph Drawing: Algorithms for the Visualization of Graphs.* Prentice Hall, New Jersey, 1999.
2. Anton N. Dragunov, Thomas G. Dietterich, Kevin Johnsrude, Matthew McLaughlin, Lida Li, and Jonathan L. Herlocker. Tasktracer: a desktop environment to support multi-tasking knowledge workers. In *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, pages 75–82, New York, NY, USA, 2005. ACM Press.
3. Peter Eades. A heuristic for graph drawing. *Congressus Numerantium*, 42:149–160, 1984.
4. Kurt D. Fenstermacher. Revealed Processes in Knowledge Management. In *Professional Knowledge Management - Experiences and Visions, Contributions to the 3rd Conference Professional Knowledge Management - Experiences and Visions, April 10-13, 2005, Kaiserslautern, Germany*, pages 443–454, 2005.
5. Norbert Fuhr. Information Retrieval — From Information Access to Contextual Retrieval. In M. Eibl, Ch. Wolff, and Ch. Womser-Hacker, editors, *Designing Information Systems. Festschrift für Jürgen Krause*, pages 47–57. 2005.
6. Fabien Jourdan and Guy Melancon. Multiscale hybrid MDS. In *Proc. Eighth International Conference on Information Visualisation (IV'04)*, pages 388–393, London, UK, July 2004. IEEE.
7. U. Riss, A. Rickayzen, H. Maus, and W. van der Aalst. Challenges for Business Process and Task Management. volume 0, pages 77–100, 2005.
8. Uwe Riss. Knowledge, Action, and Context: A Process View on Knowledge Management. In *Professional Knowledge Management - Experiences and Visions, Contributions to the 3rd Conference Professional Knowledge Management - Experiences and Visions, 2005, Kaiserslautern, Germany*, pages 555–558, 2005.

9. S. Schwarz, A. Abecker, H. Maus, and M. Sintek. Anforderungen an die Workflow-Unterstützung für wissensintensive Geschäftsprozesse. In *WM 01*, Baden-Baden, Germany, 2001.
10. Kozo Sugiyama, Shojiro Tagawa, and Mitsuhiko Toda. Methods for visual understanding of hierarchical system structures. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(2):109–125, February 1981.
11. K. Tochtermann, M. Granitzer, V. Sabol, and W. Klieber. MISTRAL: Service-orientierte Cross-Media Techniken zur Extraktion von Semantik aus Multimedia Daten und deren Anwendung. In *Proceedings Semantics 2005, Reich S., Güntner G., Pellegrini T., Wahler A. (Hrsg.) Trauner Verlag*, 2005.
12. Granitzer M. Lindstaedt S Tochtermann K., Reisinger D. Integrating Ad Hoc Processes and Standard Processes in Public Administrations. In *Proceedings of the OCG eGovernment Conference, Linz (Austria)*, 2006.
13. W. van der Aalst, A. Weijters, and L. Maruster. Workflow Mining: Discovering Process Models from Event Logs, 2003.
14. W. van der Aalst and A. J. M. M. Weijters. Process mining: a research agenda. *Comput. Ind.*, 53(3):231–244, 2004.
15. W. van der Aalst and Mathias Weske. Case handling: a new paradigm for business process support. *Data Knowl. Eng.*, 53(2):129–162, 2005.
16. L. van Elst, F. R. Aschoff, A. Bernardi, H. Maus, and S. Schwarz. Weakly-structured workflows for knowledge-intensive tasks: An experimental evaluation.
17. G. De Michelis W. van der Aalst and C.A. Ellis. Workflow management: Net-based concepts, models, techniques, and tools. In *Computing Science Report 98/7*, Eindhoven,The Netherlands, 1998. Eindhoven University of Technology.
18. Lijie Wen, Jianmq Wang, Zhe Wang, and Jiaguang Sun. A Novel Approach for Process Mining Based on Event Types. 2004.

Increasing Search Quality with the Semantic Desktop in Proposal Development

Mark Siebert[1], Pierre Smits[2], Leo Sauermann[3], and Andreas Dengel[3,4]

[1] Siemens Business Services GmbH und Co OHG, Munich, Germany
Mark.Siebert@siemens.com
[2] BTU Cottbus, Germany
psmits@informatik.tu-cottbus.de
[3] KM Dept, DFKI[1], Kaiserslautern, Germany
{sauermann, dengel}@dfki.de
[4] CS Dept, DFKI, Kaiserslautern, Germany

Abstract. Quicker response times and less production costs of proposal development require further automation of sales assistant functionality in CRM environments. Automation still struggles with the handling of abstraction and the subjective character of knowledge. Based on the knowledge creation framework the paper outlines and tests the increase of search quality with Semantic Desktop technology. The discussion of peer-to-peer settings and semantic concepts illustrates the influence of individual perspectives on search quality. It reveals first potentials and benefits for process-integration, like semantic CRM and illustrates approaches to increase knowledge worker's productivity.

1 Introduction

Quicker response times to proposal requests and *less production costs for standard proposals* without quality reduction are current CRM requirements within increasing competition and market dynamics in the IT service market. This aligns with Davenports petitions to increased knowledge worker's productivity [5]. He enhances the traditional optimization triangle of processes, IT and people by their physical working environment and their personal networks.

Responding to customer requirements sales managers today either search for similar, existing, and successful standard proposals or ask an assistant to come up with a good draft. Other than proposals in product business, service proposals require a value proposition derived from and designed to the individual customer needs rather a value proposition of the product characteristics.

Existing knowledge management tools or Proposal Automation Tools [15] already support general functionality like document handling and proposal generation. They lack deeper process integration, higher quality of search and respect of the individual characteristics of knowledge. With its new on-demand CRM platform SAP [11] provides a virtual sales assistant guiding the user through the steps within the sales process (e.g. creating a value proposition or analyze competitor's products) and

[1] German Research Center for Artificial Intelligence, http://www.dfki.de/km

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 279 – 290, 2006.

offering personalized information (like my reports, contacts, appointments, tasks, etc.). This provides a sufficient process integration and personalization of the interface but does not support the knowledge creation (e.g. a proposal) sufficiently. Sales assistants though still require extensive search capabilities to manage different information sources and to translate the customer requirements into searchable key words.

Interviews with SBS practitioners name the following reasons:

- Heterogeneous storage paths with wording differences between peers (e.g. Sales Manager uses customer - and Proposal Manager uses organizational language),
- Inexistence of knowledge assets in the central knowledge base and insufficient meta-data (kept on local desktop due to high publishing efforts),
- Insufficient dialog between roles due to communication hurdles misses respect of different perspectives, increases proposal risks and lowers quality.
- Roles are only designed from a process-related point of view defining tasks. Intentions and backgrounds (expert level) are not respected.

Recent research provides with the *Knowledge creation framework – KCF –* [14] a base for understanding the knowledge creation process as a combination of perspective taking and making. It describes six steps to develop knowledge assets (e.g. a management summary of a proposal) from an individual point of view - from gathering and mapping search requirements based on full-text (Receiver), over classifying and categorizing them through ontologies (Interpretator) up to consistent alignment through inferences (Analyzer), content selection (Reconciler), verification (Verificator) and production (Producer). The paper uses the Semantic Desktop prototype "Gnowsis" [7] to explore the possibilities of semantic search and peer-to-peer technology. It supports the three perspective taking steps of the framework in practice at Siemens Business Services: Receptor using full-text search, Interpretator exploring concepts and Analyzer applying inference rules. The perspective making process requires further investigation into reasoning and problem-solving capabilities.

The KCF works on different levels of abstraction. Therefore, Johnson-Laird suggests three levels of representation [10]: first *propositional representations*, which are pieces of information resembling natural language; second *mental models, which* are structural analogies of the world; third *mental imagery*, which are perceptual correlates of models from a particular point of view. Our approach of representing information in a Personal Information Model uses the first level of abstraction along the KCF.

2 Semantic Desktop "Gnowsis" Enabling KCF

In further development of full-text and associative search like Google and LiveLink, semantic technologies like Ontoprise[2] or the brainFiler[3] explore hidden, sub-symbolic relations and make them explicitly usable. Gnowsis [12, 13] additionally investigates peer-to-peer networks in enlarging the central knowledge base like SWAP [6] Based on an index service - Receiver - it combines organizational and individual elements

[2] http://www.ontoprise.com
[3] Developed within research project EPOS at DFKI: http://www.dfki.de/epos

within a Personal Information Model (PIM) [1] – Interpretator - and integrates rule-based (Analyzer) different information sources (e-mail, files, ontologies, web, etc.). Sauermann et al [13] define Semantic Desktop as a device in which an individual stores all her digital information like documents, multimedia and messages. All data is accessible as RDF graph.

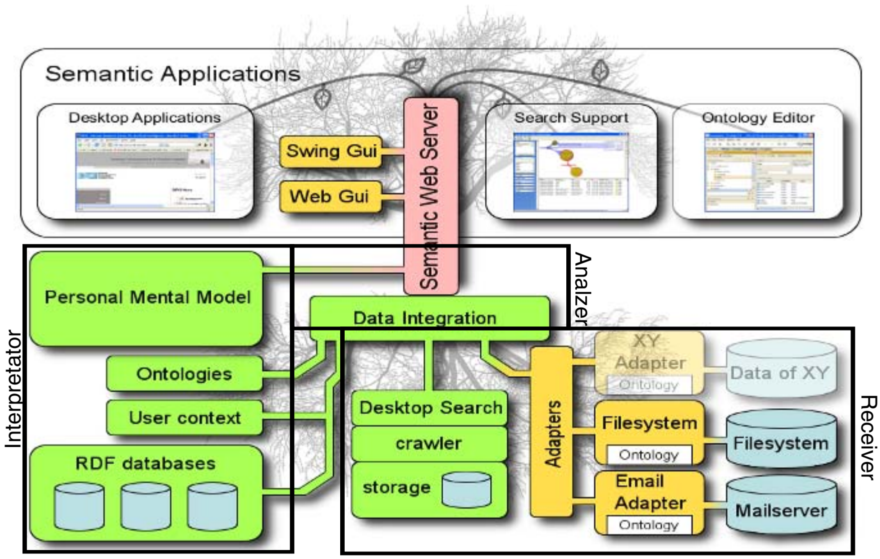

Fig. 1. Semantic Desktop Architecture [13]

Receiver: The Semantic Desktop uses the software brainFiler [8] providing the index service for all clients and determining structure and concepts of the indexed files. The brainFiler therefore creates term vectors of each file and cluster documents initially with their respective origin folder - see [1] - to improve the precision of a search result.

Interpretator: Information can be found in structured and unstructured forms with manual or automated annotations [11]. Information without knowledge structure or meta-data can be retrieved through fulltext search (e.g. Google Desktop). LiveLink uses manual annotations and meta-data to structure the content. Semantic technologies, like Semantic Miner or brainFiler use semi-automated annotations to automatically structure the content. With the automation, they respect the fact that information is stored and retrieved based on the individual preferences and priorities. These preferences lead to different perception of identical content, duplicated information in different categories of folders and information in a one-dimensional folder structure. Due to different expectations about the content of the document, they are hard receivable.

Analyzer: The architectural flexibility allows replacing the brainFiler by any other index engine. BrainFiler and Gnowsis apply rules via the Jena inference engine (http://jena.sourceforge.net/inference/) and SPARQL (http://www.w3.org/TR/rdf-sparql-query/) queries on RDF models to infer from categories. Other semantically enabled products use F-Logic to integrate rules.

3 Increasing Search Quality with the Personal Information Model (PIM)

SBS produces many thousands proposals a year for its whole service portfolio, ranging from outsourcing to solution design and system integration projects. Therein sales teams consist of different roles, like proposal managers and sales managers, with different backgrounds, expert levels and functional tasks. Sales managers feed the first rough information, like request for proposal, together with the approach (e.g. price, competitive environment, etc.) into the proposal factories to gain a first draft story – like a management summary. The result is based on existing information and references, leading to open topics and requirements for further refinement.

Proposal managers work with organizationally pre-designed and re-usable content structures (see Fig. 2, left) in a central repository. Sales managers often work in individual settings (see Fig. 2, right) according their customer requirements and area of responsibility (e.g. sales requirements from public sector are different from e.g. software business) mostly on their desktop. Nevertheless, both work on same or similar documents, value propositions and rely on similar information pools (e.g. LiveLink) with about ten thousands of knowledge objects.

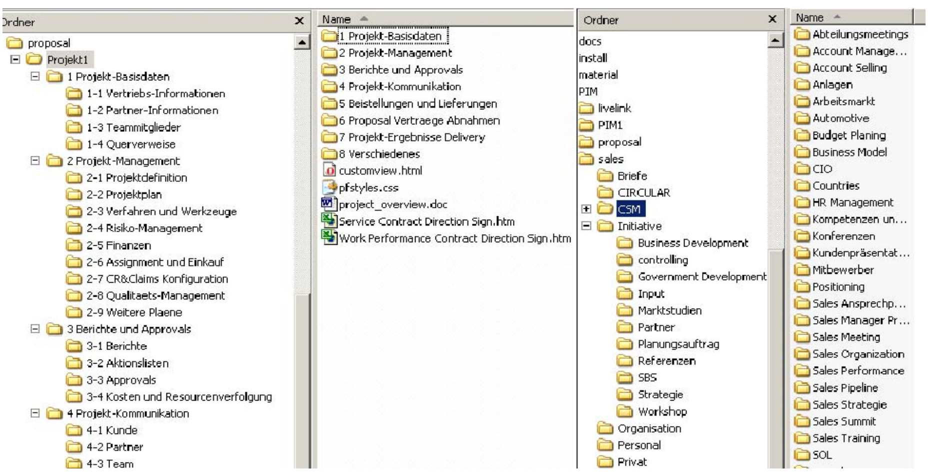

Fig. 2. Folder structure of proposal manager (left) and sales manager (right)

According to Elst et al. [7], a combination of different ontologies represents the organizational setting: information-, customer-, organization-, domain- (customer-, product-) and workflow-related ontologies. Transferred to the SBS example, role (intention) and process (situation) define a knowledge space. It includes beside the customer (e.g. industry), workflow (e.g. templates) and document (e.g. title, creator, and publisher) agreed organisational knowledge domains (e.g. document type or portfolio) and personal information structures (folder structure e.g. customer-, region-, event-based).

Together with the organizational structures, the PIM combines both folder structures (Fig 2) and represents the users' role and expert level as a model of the users' subjective perspective. The folder structures will be represented as PIM ontology in Protégé (http://protege.stanford.edu/) and integrated in Gnowsis. One user might have different roles, which end up in a mixture in his personal workspace structure. To avoid this and not to divert the results of this work, it is assumed that one user has one role and his personal workspace represents the perspective and intention of this role.

However, the model we use (PIM) is not restricted to representing a single role of a person. Instead, using a context-aware approach that activates a certain role of the user when the user is acting in this role is in principle possible. This is based on the subjective view on above ontologies from the organizational setting: the PIM is a personal view on these. It is a mediator between the mental model of the user, and the documents of the company. The current role of the user is modelled using the PIM and RDF, and then the relations of documents and projects to this role can be captured using RDF links. For our experiment, only a very limited PIM was constructed, it is a step towards representing the user's mental model.

In the Analyzer the PIM could be expanded with rules and inferences to further retrieve information about or adjust the display of the search hits. Here, deriving from the SBS proposal requirements, we identified four simple rules:

- If a project was found, determine project leader and author as a contact
- If an entry, fitting the current customer requirements, was found, determine the relevant project(s) as a good reference
- If an entry, fitting the currently required solution, was found, determine the referenced project
- If a project was found, determine the project documents as a possibly good source document for the current task

Those rules are stored in Gnowsis using forward-chaining rules:

```
# Example for reatrieving the project manager
as expert contact
(?hit retrieve:item ?project),
(?project rdf:type org:Project) ->
querySparql('
CONSTRUCT {
?x1 org:HasProjectmanager ?m. ?m rdfs:label
?labelm. ?m
rdf:type ?typem.
?x1 retrieve:todoRelateHitTo _:hit .
_:hit rdf:type retrieve:InferedHit .
_:hit retrieve:item ?m .
_:hit retrieve:textSnippet \'Projektleiter\'.
} WHERE { graph ?g {
?x1 org:HasProjectmanager ?m. ?m rdfs:label
?labelm. ?m
rdf:type ?typem.
} }
', ?project).
```

In SBS practice usually the simple rules appear context-related in combination, like: "If current role is sales manager and current task is proposal development and the found document is stored on a desktop of, or written by, a user identified as an expert about the searched topic, determine further details (e.g. assigned project, source documents and co-authors) about the document".

A full implementation in practice would have to model those combinations based on the PIM ontologies and classes. Therefore, for each class and ontology its relation to others has to be pre-defined (e.g. expert-level → show only certain document types).

The Gnowsis web interface, described in [12] is used for accessing the local and peer knowledgebase. It provides a search field for inputs and checkboxes to select peer or local search. The result page shows search results as a headline summary in their respected classes (e.g. concept, document, project, event, persons, etc.) and as detailed list with key word summary. For each item the "browse" and "link" buttons provide additional information, like members or manager of a project, in a popup box.

4 Information Retrieval Analysis Methods to Test Semantic Desktop

Different forms of collaboration exist between sales and proposal managers depending on the combination of peers (different roles and perspectives) as well as availability and similarity of objects (different peers and knowledge objects). They help to outline four use case scenarios, following process operations at Siemens Business Services. The mix between full-texts, concepts and rules in each of the scenarios influence the search results. Their differences will be investigated and discussed postulating that role and perspective will have a specific influence on the search quality and information relevance.

The following chapters will investigate and test the influence of different concepts and rules with key word examples (e.g. Help Desk, Cost Reduction, etc.) from the use case of drafting management summaries for proposal development.

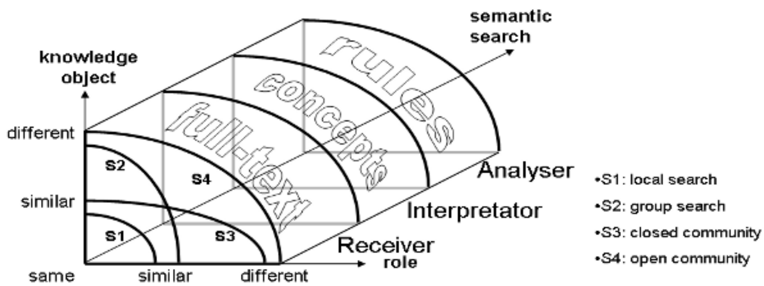

Fig. 3. Test scenarios in dependence of roles and knowledge objects

S1 - Local search: In direct comparison to Google Desktop one role (e.g. sales manager) searches his own desktop and the organizational database, now applying his native structures within the PIM. He can benefit from improvement of the precision and relevance of his results.

S2 – Group search: Similar roles working with mostly different topics and files (e.g. customers), within the same domain (e.g. calculation, trends, solution design, etc.). To optimize the process a similar vocabulary and working structure is organizationally applied. They could benefit from each other offering complementary information, which might influence one's own work.

S3 – Closed community: Different roles working with similar topics and files (e.g. around business development within a customer community). Especially customer communities are weak structured working frames. They tie together people with different roles (sales manager, solution designer, project manager, etc.), all dealing with the same customer. Some documents might be stored centrally but most information will be found on individual desktops, as they are not seen worthwhile sharing from an individual perspective. Everybody still works primarily in his own working space. Some information is shared within the community space. They could benefit from each other offering complementary information and increase of relevance as similar documents and information might be labelled differently in other peers.

S4 – Open community: Different roles working with different topics and files (e.g. Internet, Intranet). Everybody works primarily in his own working space. Information is shared from an organizational point of view. They could benefit from each other offering complementary information and increase of relevance as similar documents and information might be labelled differently in other peers. The size of the database and multiple roles might reduce relevance of search results.

Performing searches with different typical queries from SBS operations like "help-desk", "call center", "customer centralization", etc., "cost reduction" seems to be a good example for demonstrating the particularities of the Gnowsis search. The results of "cost reduction" will be base for findings and discussion. Language and document relationships have to be considered in manual pre-selection:

1) Core organizational wording is mostly identical in English and German written documents. As no translation functionality is included yet, the test will focus on English search queries.
2) Three principles on how to structure and search for information, are used. First, a document is related to different categories, second, a document is related to characteristic words or phrases and third, a document is related to other documents.

For the test, SBS-internal non-restricted data (102 files) downloaded from the LiveLink knowledgebase is used completed with 23 anonymised SBS management summaries (representing proposal stories) and 30 manually created files (project plans, calculations, contacts and references). These parts represent the basic set of documents. The file structures are part of the user's subjective view derived from directory, email, bookmark etc. [4, 9]. The individual file structures of the respective roles with the given basic set of documents are filled according to different scenarios.

5 Influence of the PIM on Search Quality

Evaluation of retrieval systems uses according to Brünken [4] the following sets as base and measures search quality in recall and precision: M is the set of all relevant objects as part of all system objects, P is the set of retrieved documents and objects (search result), Ma is the set of retrieved relevant documents and objects (relevant search results).

Using these assumptions for PIM, Gnowsis increases search quality [9] through

1) availability of information (**recall**) identifying all of the currently required documents, which are not labelled with similar search key words (e.g. find the

project "customer 1" when searching for a topic called "cost reduction" because the project is related to the topic)

r = |Ma| / |M| with 0 r 1

2) retrieval of no unnecessary documents (**precision**) respecting users' role and intention in ranking and selection of appropriate object and assets (e.g. receive results from peers, who all labelled them similar, as higher ranked as those, who are labelled differently by different peers)

p = |Ma| / |P| with 0 p 1

With this Fig. 4 illustrates the test results with different key words across the scenarios as influence of semantic concepts on recall and precision.

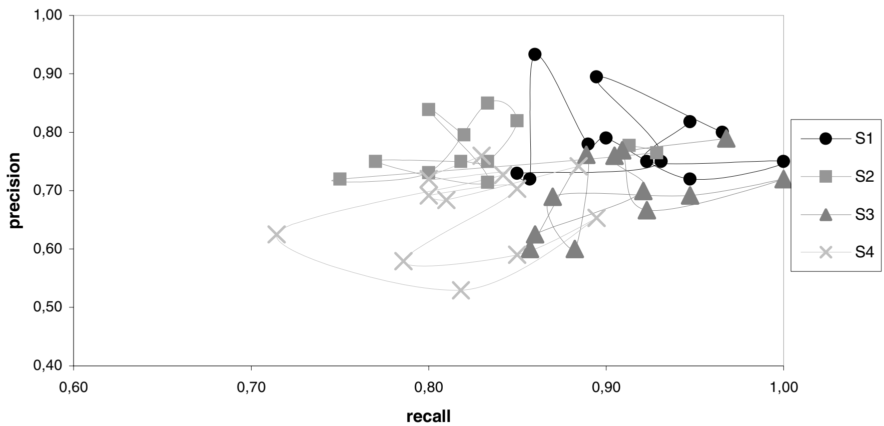

Fig. 4. Influences in precision and recall

Each point represents a key word. The key words are the same across the scenarios. The results form four different areas according to the scenarios. But overall, the tests proofed the trend that the Semantic Desktop increases search quality of existing full-text or associative searches in recall AND precision in an automated form. Full impact could be achieved in S1, whereas the results of S2 and S3 explain that a peer-to-peer approach has either high influence on recall OR precision due to an increased document base. The positive PIM and peer-to-peer effects seem to level each other out in S4. Further research would be required for this context.

S1: Local search

A system, like LiveLink, enables the user to retrieve ranked documents containing the exact search phrase and documents located in a folder containing the required phrase. Important meta-tags (like authors) of the documents are presented as well. The test with the local search of brainFiler for "Business efficiency" and "call-center" shows a higher recall-value (r=1,0) as Livelink (r=0,25), which might be caused by a good document index. However, this recall difference is valid for other key words as well.

If we apply the PIM, the test shows continuously high results in precision but lower results in recall. Reason is that the ontology only allows additional relevant hits but adds a big amount of possible documents.

Using rules within Gnowsis, the tests confirm with few exceptions (RFID, Communication) an increase of recall values with constant precision results. It indicates that rules bring additional value in search quality.

The illustrated effects with higher recall and precision require though a combination of full-text, concepts and rules - retrieving results (receiver: item about "cost reduction"), putting them into context (Interpretator: item is a "customer requirement") and applying rules (Analyzer: customer requirements are assigned to projects, "cost reduction" → "customer 1")

S2: Group Search and S3: Closed community

The value of the influences in S2 and S3 is determined by the search situation, requirements and goals.

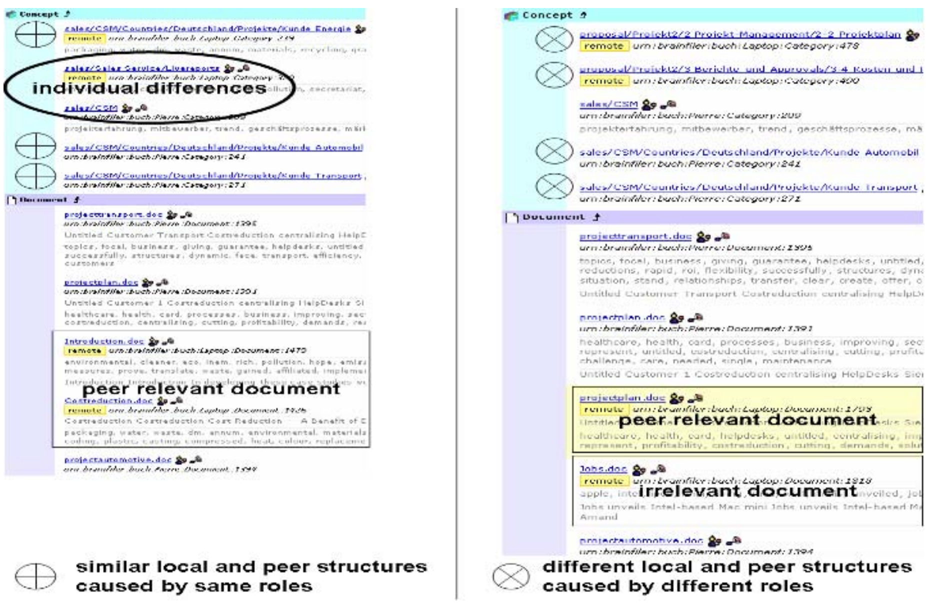

Fig. 5. Results of peer search S2 (left) and S3 (right)

For S3, for instances, it would already be helpful in the collaboration of a proposal and a sales manager to benefit from the knowledgebase of the other role in retrieving more relevant information from the "shared" knowledge space. For S2, for instances, it would already be helpful to receive better documents and rankings using the other's perspective but accepting an increase of irrelevant documents.

Users working in an organizational predefined structure (like proposal managers in S2) can take advantage of the previously mentioned local functionality. They can expand their benefits by searching peers with the same roles, based on similar PIMs as a comparison among a group of SBS proposal managers showed. Accessing different peers enlarges the total amount of possibly relevant documents and classifications. Fig. 5 illustrates the differences between local and global search. Accessing other (similar) peers returned more relevant documents stored in similar structures. Thus, the tests show that searching peers on similar organized desktops can reveal additional, unknown information compared to local search, having positive effects on precision by "knowing" the perspective of the accessed desktops.

Changing the role (perspective) in S3 means changing the PIM assuming that the same relevant files are shared among the peers (sales and proposal manager). With this, the files are seen within different contexts, experiences and perspectives. This allows the system to either expand the search results as somebody linked the file to a relevant concept or rank them higher as most of the peers linked the file into a homonymous concept. Searching peers increases recall caused by a larger total knowledgebase and consideration of different perspectives/PIMs. Contrary to S2 it might increase recall, due to a smaller set of knowledge objects. Both scenarios illustrate the influence of PIM and the database (peer-to-peer) but show that they are reliant on each other and could only improve significantly one of the search quality indicators recall OR precision. Through the PIM it is possible both to model the role of the user in a way that can be reused across the involved systems (gnowsis and brainfiler). The subjective view that was created by one user (according to classifications that map to his mental model) can be correlated to the views created by others, again based on their role or categorization metadata. In the company setting where people agree to share such metadata, we see that the recall values are increased.

S4: Open community

Searching desktops with different files of users with different roles (sales and proposal managers) additionally could broaden the "mind" of the semantic desktop network e.g. by having access to many classifications of a document. Thus, a required document could be retrieved via several search queries. As S4 is a combination of the previous scenarios, the results are a combination of the list retrieved in the previous situations. It is a combination of all relevant and irrelevant documents. It underscores the expectation from S3 that peer search has still positive impact on recall and precision when searching peers of users with a different role (perspective). The positive impact on precision could be improved by further integration of the roles (abilities, skills, expert level and desktop structure) of each peer user into Gnowsis.

In summary, the role concept and individual perspective are major drivers to increase precision over decentral and different knowledge bases. Further optimization of the balance between recall and precision would be required. Precision depends on the degree of publicity of the searched roles within the local environment. Recall depends on the amount of objects within the knowledge base and increases less with the increased size of the knowledge base.

As the individual preferences are impossible to determine within the given situations they are assumed constant for further optimization of search quality. Additionally other criteria like the amount of knowledge objects is set constant as well, because they are mainly determined by given facts, which can't be influenced by the Semantic Desktop Gnowsis. Integrating the role / perspective into the search process is though the top criteria for improving search quality.

The results show that the semantic desktop can improve precision by considering the role of an individual workspace (S1 and S2). Deeper integration within a social semantic desktop could expand these to an even larger variety of roles (S3 and S4).

6 Conclusion and Summary

The tests proof the positive influence of the Personal Information Model, representing roles and perspectives, on search quality. Qualitative analysis furthermore emphasizes the impact of complete publicity of the roles information model on search quality. Semantic Desktop with this avoids additional editorial or communication efforts, which are required today without semantic search in proposal development.

Semantic search technology and modeling framework enables Johnsons-Lairds's [10] first level of abstraction and are a good base to integrate personal views into process-related, task- and competence-oriented role concepts from an informational point of view respecting the subjective character of knowledge. Future research will have to provide solutions for enabling second and third level of abstraction, like *mental models* and *mental imagery.*

This might add a piece to the shift from a reactive to an *active search support* and information provision, e.g. within the CRM Sales assistant from SAP ("related documents"). Displaying search hits in a process-dependent template structure of a management summary (market trends, business scenarios and customer requirements, compelling events, cost driver, solution, benefits) provides a pre-structured base for further content retrieval from the found documents. Based on the meta-data "document type", e.g. market information, could be shown in the category "market trends".

In total, semantic (handling of abstraction, translation and similarity engine), technology (agents, mash-ups, user observation) and architecture (handle peer-to-peer, content management, workflow systems) are the three elements driving further research in this area leading to better search results and reduced response times and lower process costs.

Acknowledgments

Parts of this work have been supported by a grant from the German Federal Ministry of Education, Science, Research and Technology (EPOS, FKZ ITW–01 IW C01). We gratefully acknowledge Heiko Maus for his review and discussion support.

References

1. Abecker, A.; Bernardi, A.; Maus, H.; Sintek, M. & Wenzel, C. Information Supply for Business Processes - Coupling Workflow with Document Analysis and Information Retrieval, in: Knowledge-Based Systems, Special Issue on AI in Knowledge Management, Elsevier, (2000), 13 , 271-284
2. Bähr, J.T., Dannenmann, P., van Elst, L., Hust, A., Lauer, A., Maus, H., Schwarz, S.: EPOS - Milestone 1. Document D-04-01, DFKI (2003)
3. Berners-Lee, T.: WWW Past&Future(2006) http://www.w3.org/2003/Talks/0922-rsoctbl/
4. Brünken, R.: Automatische Rekonstruktion von Inhaltsbeziehungen zwischen Dokumenten. Benutzeradaptives Informations-Retrieval in Wissensbasen, Shaker, (1998)

5. Davenport, T.: Thinking for a Living, Harvard Business School Press, 2005
6. Ehrig, M., Tempich, C., et al.: Swap: Ontology-based knowledge management with peer-to-peer, in: Izquierdo, E., ed.: 4th European Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'03), London, World Scientific (2003) 557-562 - http://swap.semanticweb.org
7. Elst, L., Abecker, A., Maus, H.: Exploiting user and process context for knowledge management systems, in: Workshop on User Modelling for Context-Aware Applications at User Modeling, Sonthofen, Germany (2001)
8. Hees, H.: Nextbot knowledge framework. brainbot Technologies AG (2004) (http://brainbot.com/site3/dokumente/nextbot_knowledge_framework_whitepaper.pdf)
9. Holz, H., Maus, H., Bernardi, A., Rostanin, O.: From lightweight, proactive information delivery to business process-oriented knowledge management, in: Journal of Universal Knowledge Management (2), (2005) 101-127
10. Johnson-Laird, P. N.: Mental models:Towards a cognitive science of language, inference, and consciousness. Cambridge, MA: Harvard University Press, 1983.
11. SAP, Video-Presentation CRM on demand system http://www.sap.com/solutions/business-suite/crm/crmondemand/index.epx, March (2005)
12. Sauermann, L.: The gnowsis – using semantic web technologies to build a semantic desktop. Diploma thesis, Technical University of Vienna (2003)
13. Sauermann, L., Bernardi, A., Dengel, A.: Overview and outlook on the semantic desktop, in: Stefan Decker, Jack Park, D.Q., Sauermann, L., eds.: Proceedings of the 1st Workshop on The Semantic Desktop, ISWC 2005 Conference. (2005) (http://www.gnowsis.org)
14. Siebert, M.: Knowledge creation framework - enabling just-in-time information delivery, in: Althoff, K.D., Dengel, A., Bergmann, R., Nick, M., Roth-Berghofer, T., eds.: Wissensmanagement, DFKI, Kaiserslautern (2005) 642-647
15. Wilson, G.: Proposal automation tools, in: Proposal Management Journal, (2001) 67-73

Managing Many Web Service Compositions by Task Decomposition and Service Quality Evaluation

Yuya Takabayashi[1], Harutaka Niwa[1], Mitsuharu Taneda[1], Naoki Fukuta[2], and Takahira Yamaguchi[1]

[1] Keio University 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, 432-8011 Japan
{yuuya, haru_tk, tanem, yamaguti}@ae.keio.ac.jp
[2] Shizuoka University 3-5-1 Johoku, Hamamatsu-shi, Shizuoka, 223-8522 Japan
fukuta@cs.inf.shizuoka.ac.jp

Abstract. As Web Service technology becomes popular in organizations, we happen to compose many kinds of Web Services; software components from private UDDIs and programmable Web APIs from portal Websites. Thus we should manage many Web Service compositions to select just practical ones from many candidates. Here is discussed how to do it by task decomposition with task repository and service quality evaluation for many composed Web Service candidates. Our case studies with business trip show us that our system should select practical Web Service compositions from many candidates.

1 Introduction

As divisions and organizational structures in a company are now dynamically changing to reflect rapid changes of the market, it is necessary for the adaptation to integrate heterogeneous information services over them. Merging companies is also big pressure to change organizational structures in short periods. It is also crucial to integrate information resources outside of the company, since now there are many useful hot information resources on the Web.

On the other hand, Web Service technology [1] is a key technology to inter-connect independent systems on the Internet or Intranets. The number of Web Services is rapidly growing since the tools and standards have been widely spread with private UDDI and programmable Web APIs from portal Web Sites such as Google and Amazon.

Looking at the above both hands, it turns out to be significant for organizations to adjust business models depending on market change and implement them by heterogeneous information service integration based on Web Service technology. However, since Web Service compositions tend to be so combinatory huge with many different kinds of Web Services, we should manage many Web Service compositions to select just practical ones from many candidates.

Semantic Web Service (SWS) technologies[2][3] have been proposed for automated 'logically correct' compositions of Web services by using semantic metadata of data and services. However, in conventional SWS technologies, it is taken into less consideration to use task knowledge for better service integration.

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 291 – 302, 2006.

Furthermore, in SWS technologies, it is assumed that quality issue is considered as just a constraint of logical correctness such as keeping acceptable response time. Here is discussed how to manage many Web Service compositions by task decomposition with task repository and service quality evaluation for many composed Web Services candidates. The remaining sections describe system structure, how to compose Web Services, how to evaluate composed Web Service candidates, case study and conclusions.

2 Web Service Composition System

In this section, we show the outline of our system. Figure 1 shows the overview of our system. The system consists of two major parts. One is Task-decomposition part. The other is semi-automated service composition part. In this paper, we focus on the semi-automated service composition part. The details about Task-decomposition part is described in [6].

2.1 Structure of Task-Decomposition Part

As shown in Figure 1, Task-decomposition part consists of five parts; demand-description part, Task-decomposition part, search-Web-service part, execute-Web-service part, and Knowledge-Resource (KR) part. These 4 parts (excluding KR) are functional parts. So tasks are decomposed to Web Services in the following flows. Each functional part refers to Knowledge-Resource part. Knowledge-Resource consists of Fact, Coarse-grain-size repository, Middle-grain-size repository, Fine-grain-size repository, and object ontology. In the follow section, we describe detail of each grain-size repository.

2.1.1 Coarse-Grain-Size Repository

Coarse-grain-size repository contains the basic directory of user's tasks. In this paper, the relation defined by IS-A and HAS-A relations are given as the relations between tasks based on the structure of MIT Process Handbook [7] a well-known process ontology of business tasks. The IS-A hierarchy is defined by Atomic Tasks and Complex Tasks. An Atomic Task is a simple procedure such as hotel retrieval task, and it can be achieved directly by a single Web service. A Complex Task is an abstract task that shows the purpose of the task. It can be decomposed to atomic or complex tasks. (For instance, “Train retrieval” generally has the purpose “arrangements of transportation”).To make complex task executable by Web Services, abstract task to one or more concrete tasks. Both roles are described by the HAS-A relation. For instance, in the travel task hierarchy case, “Travel” task is composed of two subtasks “Arrangement of the way to go to the destination” and “Arrangement of the way to go back from the destination”. This is an example of decomposing “Travel” task by the procedure that is constituent of it. The task “Train retrieval for the go way” is given as HAS-A to “Finding out the way to go to the destination” that is an element that the task “Train travel” HAS-A. This means the train retrieval is given as a method to arrange the train.

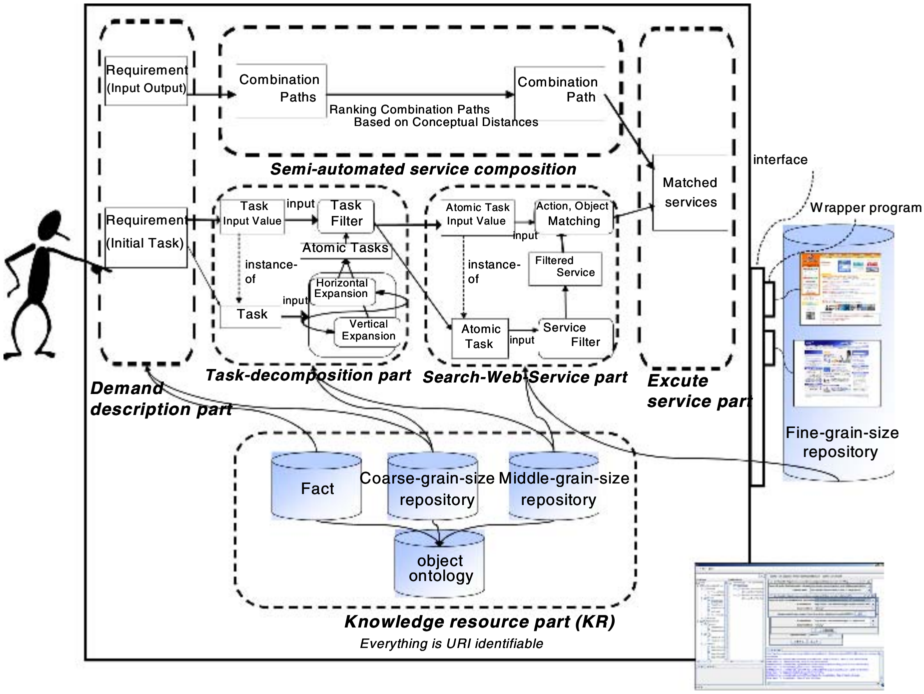

Fig. 1. The overview of web service composition system

2.1.2 Fine-Grain-Size Repository

In Fine-grain-size repository, the Web Services and their service semantics are stored. There are two types of semantics for describing Web services, one is called “Light Semantics” and the other is called “Heavy Semantics”. In a well-known SWS framework WSMO[3], those are distinguished strictly. Light Semantics describes the I/O concept and the action (retrieval, delivery, and purchase, etc.) of service as a meaning of service. In Heavy Semantics, Precondition and Effect are described as semantics of the Web service. In WSMO, they try to integrate both of them in a single framework. The mechanism of service discovery using both approaches can achieve more flexible discovery. However, in real service discovery cases, since the cost of preparing such metadata is not negligible, it is necessary to consider a trade-off between the description cost of metadata and the flexibility of the discovery.

In this paper, we use the Light Semantics for the description of Web services. This is meaningful since in this research we narrow the domain and the task so that a task is not associated to massively large amount of services.

2.1.3 Middle-Grain-Size Repository

Middle-grain-size repository stores the model of Web services. In this paper, the model of Web services is described using the role concept and basic concept. Role concept [9] is the concept that is used as abstracted I/O of tasks [8]. For example, in "Route retrieval", "departure" and "Destination" are the role concepts, and these do not have instance.

It is necessary to assign a Web service to the task to accomplish the task. For instance, "Jorudan Route Search" Web service will be assigned to use with the task "Route retrieval".

The actual I/O of a Web service should be instance (basic concept). Therefore, it is necessary to give a instance for the role concept to invoke the Web service with a certain task. Middle-grain size repository defines basic concepts that can give the instance for I/O role concepts of the task.

Such association puts the meaning for both the Fine-grain-size side and the Coarse-grain-size side. From the Fine-grain-size side, similar Web Services can be modeled more generic than the service provider originally constructed, and the essence of the Web service will be described. For example, the inputs of the task "route retrieval" are "departure" and "destination". But "Jorudan route search", that is a Web Service that achieves the task "route retrieval", has also "Time" as input, while it is not essential to accomplish the task. By removing such service specific information and describing only essential information to accomplish the task, the understanding of Web services will become easier.

From the Coarse-grain-size side, instances can be assigned to role concepts. By using middle-grain-size repository, Web services can be found to accomplish the task.

2.1.4 The System Flow of the Task-Decomposition Part

Figure 2 shows our implementation of Task-decomposition part. Here, following are ordinal user operations for task decomposition using task repository. First, user selects a task from is-a or has-a hierarchy tree then, drag-and-drop the task to the Task-combination-Editor window. The system displays the inputs which are necessary to achieve the task by using Coarse-grain-size repository. Next, the user enters these inputs and press the Task-decompose button, then the system decompose the complex tasks to the atomic tasks by using user inputs and Coarse-grain-size repository. The User confirms the tasks and inputs, and press Service-invoke button. Then these atomic tasks are associated with the Web Services by using Fine-grain-size and Middle-grain-size repositories. The Web Services are executed manually by pressing Service-execute button, and finally the user will obtain the result.

2.2 Structure of Semi-automated Service Composition Part

In Semi-automated service composition part, the needed task is given by the user. The system analyzes the input task, and finds out the input values and the output values of the task. Next, the candidate combinations of the Web service ("paths") are generated based on I/O relationships among tasks and Web services. Here, object ontology is used for specifying such relationships. Then, the quality of generated paths are tested by using the evaluation function we will show later, and then the paths are sorted by the value and displayed to the user (Figure 3).

In the rest of following subsections, we describe the details of generating and evaluating paths by using the conceptual model of WSMO.

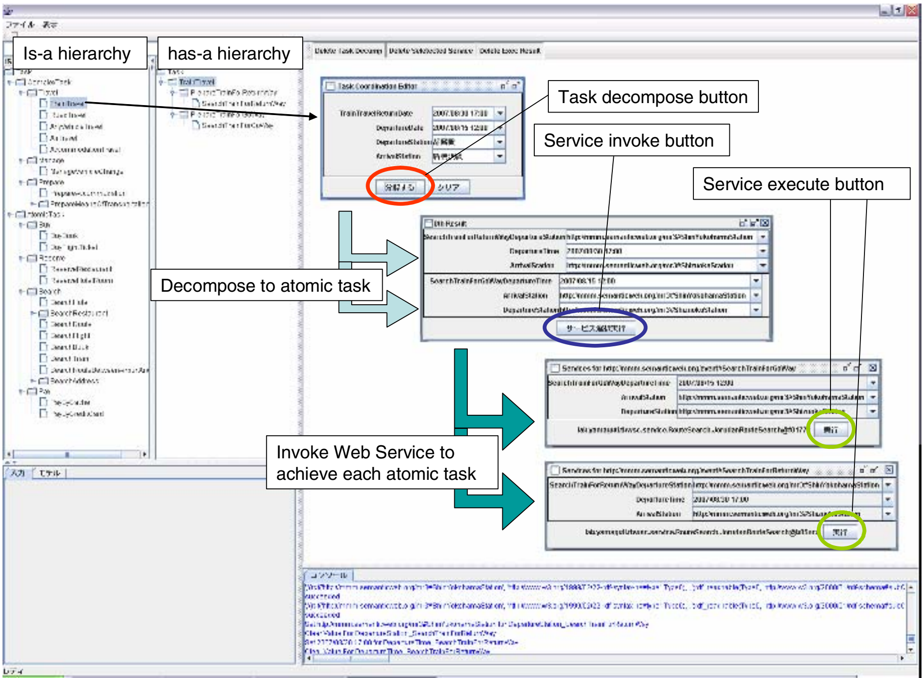

Fig. 2. Screenshot of the Task-decomposition part

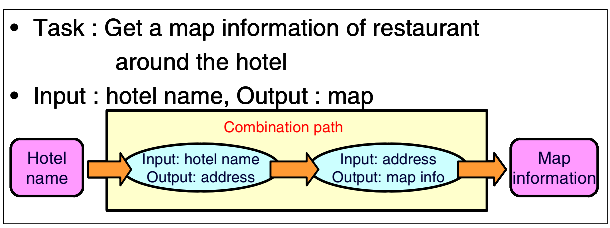

Fig. 3. An example of Semi-automated service composition part

2.2.1 Web Service

Since there are few Web services available on the internet now, in this paper, we use Web service wrapper to treat some ordinal Web applications as Web services to increase available Web services in our case study. By using wrapper program, each Web application follows the model of WSMO. In the experiment and the evaluation part, it was necessary to have an enough number of Web services to evaluate the appropriateness of the evaluation criteria. So we added a lot of pseudo services which have only I/O concepts for these experiments.

2.2.2 Goals

In this paper, we consider the user's demands can be classified into two cases. One is the complex demands which are not executable by one service (ex. traveling to somewhere etc.). The other is the ad-hoc demands to know about something that are related to previous results of services. (ex. To search the restaurant on the side of the hotel). The Task-decomposition part treats the former, and the Semi-automated service composition part for service treats the latter.

2.2.3 Ontologies

In the Semi-automated service composition part, we prepared object ontologies for inputs and outputs of services to represent Web service ontologies. Here, we used classes (objects) with restrictions of properties (attributes) to denote I/O objects of Web services. For example, there are a service that has "address of somewhere" as input and "address of restaurant" as output. In this case, the input "address" is restricted as an address but no property restrictions are applied. However, the output "address" is limited to the one for a restaurant. Here, it is possible to define a new class that represents an address of a restaurant. However, it increases the number of classes used in the ontology that will make it more difficult to compare semantic distances of each objects. To simplify the ontologies for semantic matching, we use classes with an attribute restriction instead.

EDR electronic dictionary [4] was used to compute the conceptual distances of those I/O objects on our semantic matching mechanism.

2.2.4 Mediators and Semantic Matching

Mediator is a general mechanism to ease a lot of problems that will occur when Web Services are composed by using existing Web services. In WSMO, four types of Mediators are defined for semantic Web service composition. Each Mediator is classified as follows.

- OO-Mediator - provides mappings for different ontologies, to allow multiple ontologies for describing Web services (and also often used for importing other ontologies in WSMO).
- GG-Mediator - provides mappings to the user's goal and the goals that can be solved by Web services.
- WG-Mediator - provides mappings of Web services and goals, and resolves mismatches between the Web Service and the goal.
- WW-Mediator - presides mappings from 'conceptual' Web services to real Web service instances.

In this paper, we used WW-Mediator in these 4 mediators. The following two methods are used for matching the I/O of Web services. The first one is strict matching. In this method, the input objects of a service that will be executed should be subsets of output objects of services that have already executed. The other is extended matching. In this method, direct subclasses of output objects are allowed for the next input objects. At the extended matching, we restricted this mapping for only direct subclasses of output objects to keep accuracy of matching in a expected level.

2.3 Service Selection with Quality Evaluation Criteria

The system generates a lot of candidate compositions (paths) when there are many similar services to accomplish the user's goals. Therefore, the selection of most relevant service compositions is crucial. On WSMO, this service selection issue is considered to be solved in the WW Mediator, and not focused on providing detailed mechanisms for service selection from semantically consistent candidates. In this paper, we select relevant service compositions based on the score that are considered with QoS of the composed paths that are followed with the model of WSMO. Our previous research [5] used the following criteria.

- Value of provided information by the service (I) - The total value of provided information of the Web service. The value was calculated based on the coverage of backend databases used in the service.
- Heuristic Usability Index (U) - Heuristic value for the service that denotes the quality of service. It includes accuracy of the provided results, and also considered the usability including response times of the service.
- Length of composed paths (L) - Number of Web services included in the path
- Type of matching (M) - strict matching or extended matching
- Suitableness of Combination (C) - A Heuristic value that denotes suitableness of combining two services as a set.

In this paper, we try to use parameters that do not need any heuristic values that are assigned prior to the service composition process.

In case we use only L and M for evaluation criteria, the composed paths are grammatically valid, but the web services that are used in the paths may not relevant to the user’s tasks. In this paper, we propose new criteria based on conceptual distances among the generated paths for the Web service and the description of user's tasks.

2.3.1 Service Selection Based on Conceptual Distances

We propose a method based on conceptual distance between the intermediate results of the paths and the concept extracted from user’s task descriptions. The intermediate results shown in Figure 4, are output of some services but that will be passed as input for following services in a composition). In Figure.4, “address”, “ZIP code”, and “Amount of money” are intermediate results.

The key concepts of user's tasks are extracted by following steps. At the beginning, User's task is described as a single sentence in natural (Japanese) language. Relevant words are extracted from the task description by applying Japanese morph-analyzer. And then, a pair of two words is associated sequentially. Here, we use the key concepts to be a pair, since the model of I/O parameters of Web services are represented as a pair of “attribute” and “object”.

The k -th key concepts of user’s task is defined as $(KO, KA)_k$. KO is the key concept “object” and KA is the key concept “attribute”. And the l -th intermediate result of web services is defined as $(IO, IA)_l$. IO is the intermediate result “object” and IA is the intermediate result “attribute”. Then d_k which is conceptual distance between k -th key concept pair and all intermediate results in the path is defined in (1)

$$d_k = \min \forall l\{d(KO_k, IO_l) + d(KA_k, IA_l)\} \tag{1}$$

And the conceptual distance between the task (t) and the path (p) is defined in (2)

$$D_{t,p} = \sum d_k \tag{2}$$

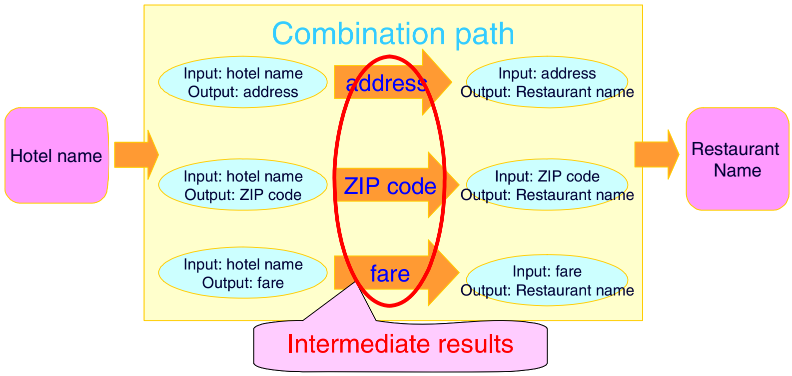

Fig. 4. The Concept of Intermediate Results

2.4 Evaluation Function

The evaluation function is defined by the following (3).

$$V_{t,p} = \frac{\sum_{i=1}^{|p|-1} m(p_i)}{|p| + D_{t,p}} \tag{3}$$

$V_{t,p}$ is the evaluation value of t and p. Here, t denotes the task sentence, and p denotes the combination path. $|p|$ is the number of Web Services contained in the combination path. $D_{t,p}$ is the conceptual distance between the task sentence and the combination path. $m(p_i)$ is the type of matching used in p_i. Here, p_i is the combination of i-th Web Service and $i+1$-th Web Service.

3 Evaluation

3.1 Overview

In this section, we will show the experimental results for the proposed service composition mechanism shown in section 2.In this experiment, we prepared two real tasks that are needed to combine two or more services to accomplish them. And then, we evaluated appropriateness of the generated service compositions for those tasks.

3.1.1 Tasks

We prepared two tasks to be accomplished by using our system. Each task is represented as a natural language description in a single sentence. The key concepts and initial inputs and expected outputs, that are used for generating service compositions in the system, are extracted from the each task description. Following are the task descriptions used in the experiments.[1]

Task 1:"I want the map of a restaurant around the hotel."
Task 2: "I want to know the name of the station near the restaurant where the dish of a certain menu was served."

The Natural language processing technology is needed for the key concept extraction from the task description. In the experiments, we used the system that has a simplified the natural language processing module, since we focused our experiments on the evaluation of service composition mechanism rather than concept extraction from a natural language text. Here, from the task 1, we can get four concepts: "hotel", "around", "restaurant", and "map" by using morphological analysis engine. And then two key concept pairs: "Hotel + Around" and "Restaurant + Map" are generated. And, from the task 2, we can get six concepts: "menu", "dish", "restaurant", "around", "station", and "name" by using morphological analysis engine. And then three key concept pairs: "Menu + Dish", "Restaurant + Around", and "Station + Name".

3.1.2 Prepared (Pseudo) Web Services

Due to the limitation of number of actually usable Web services on the Internet, in the experiments, we used abstracted pseudo Web services that are not in the actual use. On the preparation of pseudo Web services,, the following five were considered.

(1) Should be Include the Web services that can be used to accomplish the tasks.
(2) Should be Include the Web services not related to the tasks but they are possible to used with services indicated in (1).
(3) Should be Include the Web services that can be combined with the services indicated in (2).
(4) To keep reality, the prepared pseudo Web services could instantiated into real services.
(5) Prepare a lot of services as many as possible while keeping reality.

We prepared total 50 pseudo Web service definitions. An example of the pseudo Web services are shown in Table 1("*" in the table shows that the object is good anything).

3.2 Experimental Results

Task 1: "I want the map of a restaurant around the hotel."

For Task 1, totally 54546 combination paths were generated by the system. The highest five paths ranked by equation (3) are shown in Table 2.

Task 2: "I want to know the name of the station near the restaurant where the dish of a certain menu was served."

[1] Actually, the task descriptions are written in Japanese texts in the experiments. Here, for better readability, we showed concepts and task descriptions that are translated into English texts. The meanings of task descriptions and extracted concepts shown here might be slightly different from original meanings.

Table 1. Services inputs and outputs

Input		Output	
Object	Attribute	Object	Attribute
hotel	name	hotel	address
hotel	name	hotel	zipcode
.*	address	restaurant	address
.*	zipcode	restaurant	address
.*	address	.*	map
.*	amount	restaurant	name
station	name	restaurant	name
cooking	kind	restaurant	name
restaurant	name	restaurant	address
.*	address	station	name
author	name	book	name
hotel	name	stock	amount
hospital	name	hospital	address
book	name	book	number
company	name	stock	amount

Table 2. Result of task1

rank	V	combination			
		turn	input		output
1	0.1	1	hotel,name	→	hotel,address
		2	hotel,address	→	restaurant,address
		3	restaurant,address	→	restaurant,map
2	0.1	1	hotel,name	→	hotel,address
		2	hotel,address	→	restaurant,phone
		3	restaurant,phone	→	restaurant,map
3	0.1	1	hotel,name	→	hotel,address
		2	hotel,address	→	restaurant,zipcode
		3	restaurant,zipcode	→	restaurant,map
4	0.09091	1	hotel,name	→	hotel,address
		2	hotel,address	→	restaurant,zipcode
		3	restaurant,zipcode	→	restaurant,address
		4	restaurant,address	→	restaurant,map
5	0.09091	1	hotel,name	→	hotel,address
		2	hotel,address	→	restaurant,address
		3	restaurant,address	→	restaurant,zipcode
		4	restaurant,zipcode	→	restaurant,map

For Task 2, totally 234 combination paths were generated by the system. The highest five paths ranked by equation (3) are shown in Table 3.

For Task 1, top the three paths can be considered to be appropriate to accomplish the task. For Task 2, top one path can be considered to be appropriate to accomplish the task. The result shows that our proposed service composition method can find out sufficient compositions from a large number of composition candidates.

3.3 Discussion

Technologies for (semi-)automated Compositions of Web services have been investigated in many different ways [12]. For example, Zeng et,al. proposed a method based on Service's quality Criteria[10], and Narayanan et,al. proposed a method based on DAML and Petri net[12]. In our system, we used task decompositions with

Table 3. Result of task2

rank	V	combination			
		turn	input		output
1	0.02632	1	menu,name	→	cooking,kind
		2	cooking,kind	→	restaurant,name
		3	restaurant,name	→	restaurant,address
		4	restaurant,address	→	station,name
2	0.02564	1	menu,name	→	cooking,kind
		2	cooking,kind	→	restaurant,name
		3	restaurant,name	→	restaurant,address
		4	*,address	→	restaurant,address
		5	restaurant,address	→	station,name
3	0.025	1	menu,name	→	cooking,kind
		2	cooking,kind	→	restaurant,name
		3	restaurant,name	→	restaurant,address
		4	restaurant,address	→	restaurant,phone
		5	restaurant,phone	→	restaurant,address
		6	restaurant,address	→	station,name
4	0.025	1	menu,name	→	cooking,kind
		2	cooking,kind	→	restaurant,name
		3	restaurant,name	→	restaurant,address
		4	restaurant,address	→	restaurant,zipcode
		5	restaurant,zipcode	→	restaurant,address
		6	restaurant,address	→	station,name
5	0.02439	1	menu,name	→	cooking,kind
		2	cooking,kind	→	restaurant,name
		3	restaurant,name	→	restaurant,address
		4	*,address	→	restaurant,address
		5	restaurant,address	→	restaurant,zipcode
		6	restaurant,zipcode	→	restaurant,address
		7	restaurant,address	→	station,name

task repository in combination with ad-hoc service compositions with semantic matching of service interfaces. We believe that the proposed path selection mechanism will improve the quality of such (semi-)automated compositions of integrated information services for better knowledge management since our approach takes into account existing domain knowledge (task repository) for better information service integration. In our case, over 50,000 combination paths were generated from only 50 services and our selection mechanism could find out appropriate combinations from them. We argue that the sophisticated selection mechanism plays important role to realize better integration of information services. Also, it is possible to use our method for integrating existing knowledge management tools and information services.

4 Conclusions

In this paper, we proposed semi-automated Web service composition mechanism that uses both referring to existing task repositories and automated generation by semantic annotations of services. We showed that our proposed service composition selection mechanism works effectively for realizing better integration of information services over divisions and organizations.

In this paper, we only used evaluation criteria that are independent from actual service instances. No evaluation criteria based on the actual service instances are used. We expect that the number of Web Services will increase. When there are sufficient services available for actual use it will be effective to add a mechanism that considers instance specific criteria in the evaluation mechanism that are extracted from user's operating log data. It is our future work to incorporate such feedback mechanisms that reflect the usages of actual services.

References

[1] Web Services: http://www.w3.org/2002/ws/
[2] S.McIlraith and T.Son and H. Zeng, "Semantic Web services", IEEE Intelligent Systems. Special Issue on the Semantic Web, vol16, num.2 pp.46-53, March/April,2001
[3] WSMO: http://www.wsmo.org/
[4] Yokoi, T.: The EDR Electronic Dictionary. Communications of ACM, Vol.38, No.11, pp.42-44, 1995.
[5] T.Ohsawa, N.Fukuta, T.Iijima and T.Yamaguchi: Ranking Web Application Composition Based on Ontologies, 5th International Conference of Practical Aspects of Knowledge Management, Workshop on Enterprise Modelling and Ontology: Ingredients for Interoperability Dec, 2004
[6] K.Terai, N.Izumi and T.Yamaguchi: "Coordinating Web Services based on Business Models", Proc. of the Fifth International Conference on Electronic Commerce(ICEC2003), Pittsburgh, PA, 2003.
[7] T.W.Malone, K.Crowston, and G.A.Herman :Organizing Business Knowledge :The MIT Process Handbook, The MIT Press, 2003
[8] G.Schreiber, H.Akkermans, A.Anjewierden, R.de hoog, N.Shadbolt, W.V.de Velde, and B.Wielinga :Knowledge Engineering and Management, The MIT Press, 1999.
[9] Kouji Kozaki, Yoshinobu Kitamura, Mitsuru Ikeda, and Riichiro Mizoguchi, Hozo: An Environment for Building/Using Ontologies Based on a Fundamental Consideration of "Role" and "Relationship", Proc. of the 13th International Conference Knowledge Engineering and Knowledge Management(EKAW2002), pp.213-218, 2002.
[10] Zeng, L. and Benatallah, B. and Dumas, M. and Kalangnanam, J. and Sheng, Q.Z.:Quality driven web services composition, Proceedings of the twelfth international conference onWorld Wide Web, pp.411-421, 2003
[11] Milanovice, N. and Malek, M.:Current solutions for Web service composition, Internet Computing, IEEE volume8 number6, pp.51-59, 2004
[12] Narayanan, S. and Mcllraith, S.A.:Simulation, Verification and Automated Composition of Web Services, Proceedings of the eleventh international conference on World Wide Web, pp.77-88, 2002

Towards an Ontology for Knowledge Management in Communities of Practice

Géraldine Vidou[1], Rose Dieng-Kuntz[2], Adil El Ghali[2], Christina Evangelou[3], Alain Giboin[2], Amira Tifous[2], and Stéphane Jacquemart[1]

[1] Centre de Recherche Public Henri Tudor, Centre d'Innovation par les Technologies de l'Information, Luxembourg
{geraldine.vidou, stephane.jacquemart}@tudor.lu
[2] INRIA, Acacia Project, Sophia Antipolis France
{rose.dieng, adil.elghali, alain.giboin, amira.tifous}@sophia.inria.fr
[3] Research Academic Computer Technology Institute, E-Learning Sector, Rio Patras, Greece
chriseva@cti.gr

Abstract. The work presented in this paper is about learning in Communities of Practices (CoP). It is situated in the context of Knowledge Management (KM) services that we are developing in the Palette project dedicated to learning in CoPs. The approach is based on several models detailed in this paper. These models constitute the theoretical grounding upon which the KM services will be based; they are organized in order to constitute a generic meta-ontology, from which a CoP-dependent ontology can be built, so as to annotate the CoP's knowledge resources.

Keywords: Knowledge Management, Communities of Practice, Learner Profile, Competency, Collaboration, Process/Activity, Lessons Learnt.

1 Introduction

According to Wenger [1], CoPs are groups of people who share a passion for something that they know how to do, and who interact regularly in order to learn how to do it better. CoPs can be found within businesses, across business units or across company boundaries [2], still they differ from business or functional units, from teams and networks: people belong to CoPs at the same time as they belong to other organizational structures. An effective organization comprises a constellation of interconnected CoPs, as these are privileged nodes for the exchange and interpretation of information. CoPs preserve the tacit aspects of knowledge that formal systems cannot capture. CoPs can be considered as a means by which knowledge is "owned" in practice. Indeed, such groups allow the functions of creation, accumulation and diffusion of knowledge in organizations.

Acknowledging CoPs emerging significance in KM, this paper presents a set of models enabling the formalization of core aspects related to CoP's every day work. More specifically, the work presented in this paper is carried out in the framework of the Palette IST project (http://palette.ercim.org/). Several CoPs on three domains

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 303 – 314, 2006.

(management, engineering and learning) are involved and studied in the Palette project. Our work is situated in the context of KM services, our aim being to facilitate the efficient and effective management of the CoP's knowledge resources. In order to reach this objective, we studied the theoretical grounding upon which the foreseen services will be based. This theoretical grounding is composed of models necessary for the service tools to exploit the knowledge properly. These models will be organized in order to constitute a generic meta-ontology, from which a CoP-dependent ontology can be built, so as to annotate the CoP's knowledge resources. The CoP-dependent ontology could then be instantiated for the different CoPs involved in the Palette project. The ontology plays two roles: enabling to model a group in general and a CoP in particular, and enabling to annotate the CoP's resources.

The paper is organized as follows: first, we present our approach based on several models. Then, we detail successively each of the proposed models (community, actor and learner profile, competency, collaboration, process / activity, and lessons learnt) and do a comparison with related works. An example illustrating the use of the models is also presented. Last, we conclude by a summary of our contributions and the further work planned.

2 The Palette Approach

Fig. 1 summarizes the models we identified as the most significant. They concern the following main concepts: community, actor, learner profile, competency, collaboration, process/activity and lessons learnt.

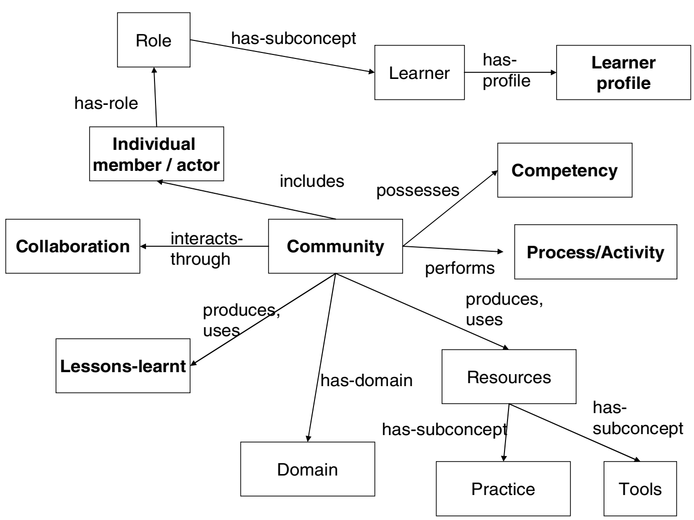

Fig. 1. Models linked to the concept of Community

Different actors can participate in a CoP as members: experts in a domain, students, or professionals. Actors can be characterized by their various roles in the CoP, and by their individual competencies, linked to the domain of the CoP. According to their competencies, actors can learn more or less about a practice and can participate more or less actively in an activity. Therefore, competency is one of the major concepts useful to define KM services appropriate to CoPs.

Collaboration is an important concept since the objective of a CoP is to deepen members' knowledge and expertise in the CoP's domain by interacting on an ongoing basis [3]. Participation is one of the two fundamental principles of negotiation of meaning in a CoP [4], the other one being reification.

Activities are central to the life of a CoP. They are the place and moment where and when interaction is made visible and fruitful. Specifically, activities are organized in order to exchange experience about a practice but also to enlarge knowledge of different members.

Learning is one of the key reasons why CoPs are being created and cultivated [3]. Every member of a CoP is at one moment or another involved in a learning process. Being able to define and characterize learners' profiles is an important aspect of KM within CoPs. Indeed, it is important to know how learners react, exposed to a piece of knowledge in order to provide services personalized to their cognitive profiles.

One key activity of a CoP is to share and exchange about the CoP's practice. This sharing of knowledge can lead to the definition of best practices, this is considered as lessons learnt. Lessons Learnt allow us to determine the behavior that is appropriate to a given situation. They lead to identification and qualification of best practices.

Let us detail each of these models.

3 Community and Actor Models

3.1 Presentation of the Palette Community and Actor Model

Wenger [2] distinguishes three dimensions along which a CoP defines itself. Firstly, its *joint enterprise* indicates what the CoP is about, as understood and continuously renegotiated by its members. The second dimension concerns the *mutual engagement* that indicates how the CoP functions and binds members together into a social entity, while the third, so called the *shared repertory of common resources* (routines, artifacts, vocabulary, styles…) indicates what capability the CoP has produced and is developed by the CoP's members over time.

As stressed in [1], a CoP can be characterized by its domain, meaning the area of knowledge that brings the community together, gives it its identity and defines the key issues that the CoP's members need to address. Furthermore, the community is another characteristic of CoPs. A CoP involves people who interact and who develop relationships that enable them to address problems and share knowledge. Community builds relationships that enable collective learning. Another aspect characterizing a

CoP is its practice. A CoP brings together practitioners who are involved in doing something. Practice anchors the learning in what people do.

The community is thus composed of members: these actors can play different roles, according to the activities of the CoP and to the CoP's stage of development. They interact, collaborate and learn by doing. They may also interact with the CoP's external environment. As far as the activity of knowledge sharing is concerned, we can distinguish the roles of knowledge provider and of knowledge recipient. On the other hand, for the social structure of the CoP, we can distinguish different roles of leaders, as suggested in [2]: inspirational leadership by thought leaders and recognized experts, day-to-day leadership by those who organize activities; classificatory leadership; interpersonal leadership; boundary leadership by those who connect the community to other communities; institutional leadership by those who maintain links with other organizational constituencies (in particular the official hierarchy); cutting-edge leadership.

Taking all the above into account, in our proposed model, a *Community* is characterized by: (1) its *Domain*; (2) its *Practice*; (3) its *Members*: these *Individual Actors* will be characterized by their individual competency, their *Social Relationships* in the CoP, their modes of participation in the CoP and of *Collaboration*, their *Roles*, their *Learning Profile*, their *Activities* inside and outside the CoP; (4) its *External Environment* that can be constituted by other actors (e.g. stakeholders in the organization that play a role of support to the CoPs, other CoPs, etc.); (5) its *Resources*: we can distinguish on the one hand the resources or outcomes developed by the CoP (artifacts, stories, routines, documents) and that constitute the *Practice* of the CoP, and on the other hand, the resources used by the CoP (e.g. the CoP's *Tools* that, according to [5], we classify into publishing tools, tools ensuring individual participation, tools ensuring community cultivation, tools for asynchronous interaction and tools for synchronous interactions); (6) its *History* and its *Life*: in particular, its life status corresponds to its current stage of development (potential, coalescing, active, disperse or memorable according to [2]).

3.2 Presentation of the Palette Learner Profile Model

Given the fact that learning is a major part of a CoP's activities, one of the most significant roles undertaken by almost all CoPs' members is the role of a learner. Acknowledging the importance of enhancing learning within an organization, in our approach we focus on learners, i.e. actors whose main objective is learning. More specifically, we present a generic Learner Profile model that aims at exposing the learners' cognitive characteristics when exposed to a piece of knowledge. The proposed model has derived after the common consideration of existing approaches on learners' profile models, learning activities and learners *per se* [6] [7]. The selection of the specific notions and relations used in the proposed model was driven by our aim to design a learner profile that could serve the ontologies development for both individual and group learners. Furthermore, in developing the proposed model, our aim was to provide a model for representing the static as well as the dynamic aspects of a learner's profile.

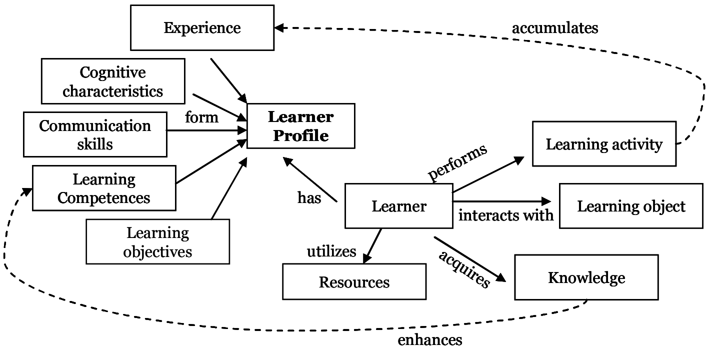

Fig. 2. The Palette Learner profile model

Fig. 2 presents the proposed *Palette Learner* Profile model. In this model, the notion of *Experience* refers to the knowledge of or skills in or observation of some thing or some event gained through involvement in or exposure to that thing or event. *Cognitive characteristics* comprise intelligence, perception, memory capabilities, creativity, organizing skills. *Communication skills* refer to the individuals' abilities in interacting with their environment. A *Learning objective* is a statement establishing a measurable behavioral outcome. The statement must include how the measurement is accomplished. *Learning competences* refer to academic background, education, training, working experience etc. *Learner* is the person who learns or takes up knowledge or beliefs. A learner is an actor's role that can be undertaken by an individual or a group of actors. *Learning activity* is every activity performed that intentionally or non-intentionally resides to knowledge acquisition. *Learning object* is every piece of knowledge. *Knowledge* refers to a fluid fix of verbal and/or manual skills brought about through training, instruction or practice that denote familiarity with facts, truths, concepts or principles. The *Resources* notion refers to every means a learner utilizes to perform a learning activity. All arrow connections appearing between the Palette Learner Profile model concepts express the relations occurring between them. For instance, the relation between learner and knowledge is the topic acquisition. It should be noted that the interactions among notions are not exhaustively defined, these are indicative and further relations or amendments to the proposed ones may occur according to findings of our future work.

Related research about learners' modeling proves that due to the complexity of human actors and the diversity regarding the learning context it is a thorny task to develop a commonly accepted learner profile [7]. For instance, in [8] a learner is depicted as a concept hierarchy that does not refer to issues such as the learning object, or the learners' interactions with their environment. The conceptual model in [9] comprises various types of information, yet it includes only the minimum information necessary to satisfy the functional requirements and it lacks information about dynamic aspects. The learner specification in [10] is a collection of information that addresses the interoperability of internet-based Learner Information Systems that support the Internet learning environment. Still, this like all the above cannot be employed for the representation of a community as a learning entity.

4 The Competency Model

The goal of our competency model is to represent the competency in the context of CoPs, specially the acquisition/exchange of competencies. We take it into account through the distinction of different roles that actors can play in their relation with competency. We also need to define the competency, and we choose to make the distinction between three types of resources that characterize the competency. The last aspect that this model allows us to represent is the link between a competency and its context of use that is represented by the environment in which it is involved.

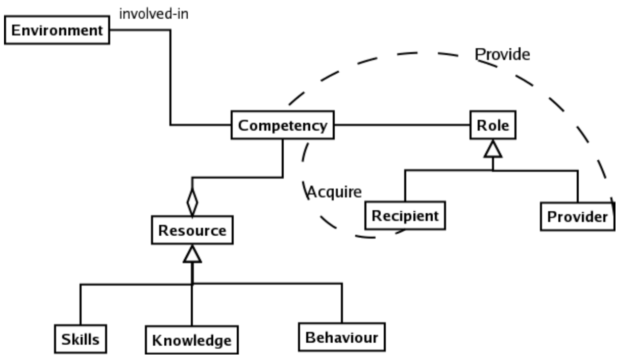

Fig. 3. The Palette Competency model

The competency model we propose involves the following concepts: *Environment*, that describes the situation in which the Competency is involved: solving a problem, achieving an objective or a task; *Competency* which is defined as a set of Resources provided or to be acquired by an Actor that plays a particular Role in the Environment to perform an Activity; *Role* that is used to link Competency to the actors. An actor can be *Provider* or *Recipient* of a Competency; *Resource* which is the set of items that compose a Competency. It can be of three types: *Knowledge* (theoretical knowledge (declarative or procedural)), *Skills* (capabilities of an actor to do something), *Behavior* (the way of behaving of the actor in a group or in a given situation).

Many models of competency were proposed in the literature, they give different points of view of competency. Our work can be compared to two main approaches: on one hand, an internal point of view that characterizes or defines the competency - thus [11] [12] make an interesting distinction between objective kinds of knowledge involved in a competency and subjective ones that provide important information on how people use their competencies; on the other hand, an external point of view that considers the competency in its context of use and acquisition [13] [14]. The KmP model [14] makes it possible to deal with both individual and collective competency and allows us to search the space of existing competencies. Since these two points of view are complementary and we need both of them to represent competency in the context of CoPs, our model tries to unify them.

5 The Collaboration Model

Collaboration is represented as a relation between four main concepts: the actors involved, the linked activities, the objectives of the collaboration and finally the resources it needs or produces.

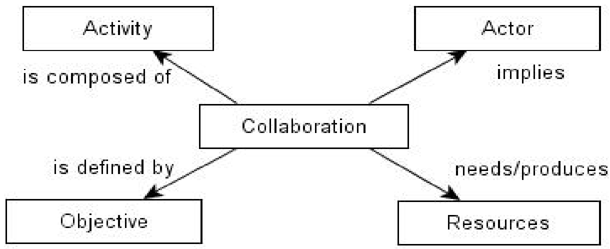

Fig. 4. The Palette collaboration model

Actor is a member participating in the collaboration, whatever his/her level of commitment or his/her knowledge. He may have several roles during collaboration, as he inherently possesses various competencies that allow a participation in several activities. *Activity* is the means to achieve the aim of collaboration. It can be planned (such as a meeting) or impromptu (such as chats or mail exchanges). Its observation and analysis can lead to best practices definitions, a decision, the creation of a document, etc. *Collaboration* arises from a goal that is common to each actor: the realization of a particular *Objective*. Each actor can have personal aims he wants to reach during collaboration. Finally, *Resources* represent anything that is used or produced by collaboration: documents, theories, software, instruments, etc.

In order to build our model of collaboration, we studied the theories of Engeström [15], Laferrière [16] and Montiel [17]. In his theory, Engeström presents the activity as a relation between the subject, the object and an artifact that could be an instrument, a tool, or a product from another activity. In the same way, Laferrière's model of collaborative learning shows that the objective of collaboration is important in order to have a precise vision of collaboration. In [17], several definitions of collaboration are presented. They all rely on the same main concepts, i.e. actor, activity, artifacts and objective, that is the reason why we proposed a unified model in Palette.

6 The Process/Activity Model

The Process model within the Palette context aims at describing sequences, roles, objectives, inputs and outputs of transformations, be they knowledge transformations within the CoP or transformations being part of the CoP's objective or core processes.

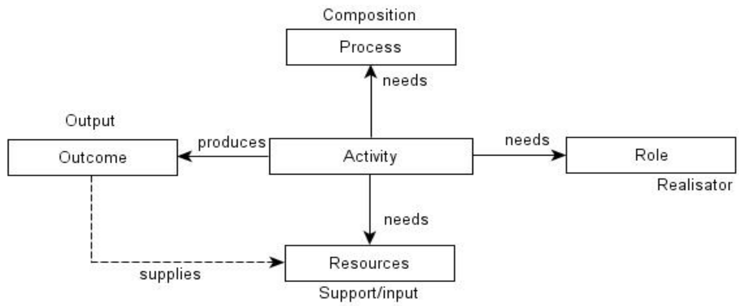

Fig. 5. The Palette process/activity model

A *Process* is a set of activities that need roles and resources in order to transform input objects into output objects, called outcome. An *Activity* is considered as a transformation of an input resource by a role during a process. An activity is seen as a ternary relation: a process, a role and resources. It needs these three elements in order to be performed. In a process, the activities are planned. A *Role* represents the responsibilities ensured by a function. It refers to a specific level of competency and to specialized skills. A *Resource* enables or helps the realization of an activity. A resource can be a tool or a product: software, a document, a competency, a practice, a method The *Outcome* is the output of the activity. It can be part of resources needed to perform another activity.

In their Coordination theory [18], Crowston and Osborn describe “processes as sequences of activities performed by organizational actors that produce and consume resources”. The Palette Process/Activity model is inspired from this theory [18]. After having defined the main elements describing a process and an activity, the terms have been adapted to be understandable and sufficiently explicit. The Activity System Model (ASM) of Engeström [15] refers to the activity theory, and allows to define activity in a context of community. In the ASM, an activity is a systemic whole. Each element has a relationship to others, each relation is also mediated. This model is complex and presents a lot of relations between the elements. It can be used in various contexts and enables to see the relations with the other models.

7 Lessons Learnt Model

Since one of the main objectives of a CoP is to enable and foster collective learning, this last model was a crucial one to build. In the model, a *Lesson Learnt* is considered as the result of a process, collectively performed by the CoP’s members; this process consists of analyzing ones’ practices in given situations, and of drawing useful recommendations from this analysis that the CoP’s members can refer to when encountering similar situations of practice.

The Lessons Learnt model that we propose, in the context of the Palette project, includes the following concepts: the *Environment* represents the context or situation in which Lessons Learnt are used or produced, it relates to the concepts of Competency and Collaboration; *Activity* relates to the individual objectives of the actors, that is to

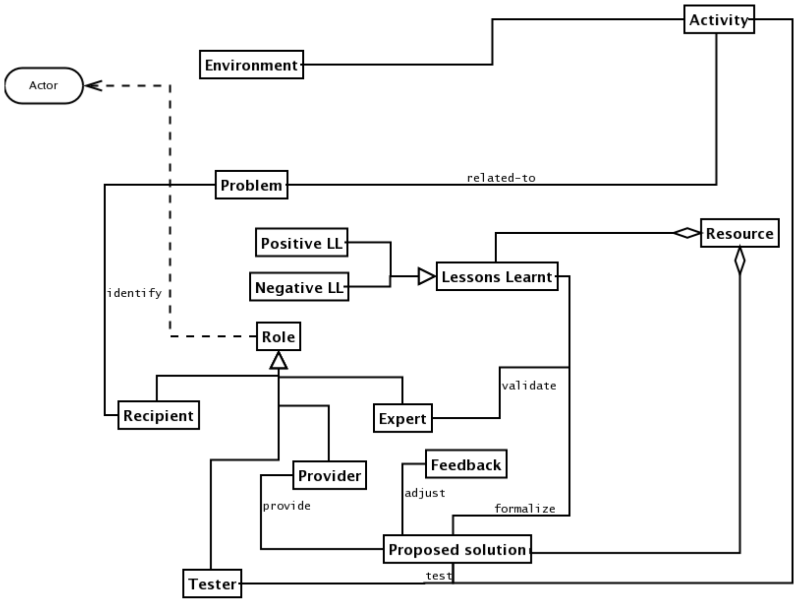

Fig. 6. The Palette Lessons Learnt model

say the tasks they have to accomplish in the organizations they belong to, the activity requires the use of Lessons Learnt in order to be performed. *Problem* is one of the main concepts linked to Lessons Learnt; it describes, in the context of an activity or practice, a point at issue whose related solutions are analyzed so as to determine the best way to figure it out; a *Proposed solution* represents the proposal of a solution to the Problem, or a clue to solve it; *Role* is the status of each Actor involved in the processes related to the Lessons Learnt, four main dynamic roles have been identified: the *Recipient* (who submits a Problem to be solved), the *Provider* (who offers a solution or a clue to the Problem), the *Tester* (who makes experimentation on the Proposed solutions and gives his/her feedback) and the *Expert* (who assesses the Proposed solutions, using his/her expertise on the domain and, at the same time, taking into account the feedback of the Testers); a *Resource* includes the different types of knowledge resources used to produce Lessons Learnt (knowledge, know-how, etc.) and which form a competency; and finally *Lesson Learnt* represents the knowledge gained and produced as a result to the Activities of sharing, exchanging and analyzing knowledge. It is the synthesis and formalization of the Proposed solutions to the Problem. A typology of Lessons Learnt can rely on their nature; for example, we distinguish: the *Positive Lessons Learnt* which consist of the activities recommended in the problem solving, they relate to the good practices of the CoP; and the *Negative Lessons Learnt* which describe the activities that are unadvised or to avoid, they relate to the bad practices of the CoP.

A survey of the works related to Lessons Learnt and experience capitalization modeling enabled us to deal with aspects specific to Lessons Learnt, such as the definition of the different operations to achieve (through the diverse roles we identify) and the description of the context in which lessons are learnt. Weber's model [19] describes the life-cycle of Planning Lessons Learnt within an organization. Weber's work was used by [20] as a basis for identification and representation of use cases in the framework of Lessons Learnt systems. Considering experience capitalization, the REX[1] method [21] [22] consists of constituting "experience cards" stemming from any activity, and containing information about the context, comments and recommendations. These knowledge elements are then stored in a corporate memory in order to be retrieved and reused by members of the company. MEREX[2] method [23] [24] also deals with experience capitalization, and aims to make explicit Good Practices to be stored in a project memory, through the use of "knowledge forms" containing the same kind of information as in REX method, but deals more explicitly with the actor's aspect.

8 Example of Use of the Models for a CoP

Let us consider the use case of a semantic portal, within Learn-Nett (Learning Network for Teachers and Trainers), a CoP involved in Palette: this CoP is focused on a shared course and aims at preparing future teachers or trainers for educative uses of Information and Communication Technologies. The models proposed will be useful for annotating the CoP itself, the CoP's members and the resources they produce or use through the portal. The model of the community enables to emphasize the practice of this CoP and the model of actors to describe the various possible roles of the actors involved: coordinator, teachers, animators and tutors, The learner profile can depict the way a new member of the CoPs learns throughout his/her interactions in the CoP. The competency model allows us to describe the competencies needed for the different roles and the concrete competencies of the CoP's members, as well as the potential competencies useful for the different CoP's activities. The collaboration model enables to describe how the members collaborate for exchanging about their activities (see below). The resources produced or used are documents (e.g the pedagogical guide and the technical guide for the course; the charter of the CoP to be transmitted to newcomers) and tools (e.g. the videoconferencing system and the virtual environment Moodle). The activity model allows us to describe the various activities of the teachers, of the tutors and of the members inside the CoP: (1) the administrative and pedagogical preparation of the course (with the pedagogical guide and the technical guide as outcome), (2) training of tutors (in this case, the outcome will be learning activities, shared views on the tutor's interventions profile), (3) regulation of the tasks of the tutors during the course, (4) evaluation and regulation of the course itself. The Lessons Learnt model will enable to represent the positive and negative lessons learnt for example from reflective analysis of the supervision methods of the tutors throughout their effective experiences of tutoring students.

9 Conclusions

This paper proposed several models useful for describing a CoP: community, actor, learner profile, competency, collaboration, process/activity, and lessons learnt. These

[1] REX: Retour d'Expérience.
[2] MEREX: Mise en Règle de l'Expérience.

models were built by adaptation of some existing models; they propose a unified view of some common models. We illustrated our models through several examples of CoPs studied in Palette[3] project: a semantic portal for Learn-Nett, a CoP of researchers and teachers in the field of educational technology; a meta-journal for Did@ctic, a CoP in training of faculty members in Higher Education and Educational Technology; an e-learning platform for UX11, a CoP of engineer students. The proposed models are structured in an ontology that will be later on extended and specialized according to the analysis of the other CoPs involved in Palette.

The link between CoPs and ontologies was studied in some recent work. In [25], the authors present a method based on analysis of the relationships between instances of a given ontology in order to identify potential CoPs in an organization. In [26], the authors develop an ontology aimed at enabling services among a civil servant CoP; [27] studies the design of situated ontologies for knowledge sharing in a CoP. But the role of all these ontologies is quite different from our ontology that aims at both modeling the notion of CoP, and at annotating CoP's resources.

As a further work, we will analyze other CoPs involved in the Palette project, in order to extend the ontology and develop several KM services based on this ontology.

Acknowledgements

This work is supported by the European Commission within the 6th Framework Program (IST-2004-2.4.10). We also thank very much Amaury Daele for his synthesis on Learn-Nett CoP and our Palette partners for fruitful discussions.

References

1. Wenger, E.: Knowledge management as a doughnut: Shaping your knowledge strategy through communities of practice. Ivey Business Journal, 68(3) (Jan-Feb 2004)
2. Wenger, E.: Communities of Practice: Learning as a Social System. Systems Thinker (Jun 1998)
3. Wenger, E., McDermott, R., Snyder, W.M.: Cultivating Communities of Practice: A guide in Managing Knowledge. Harvard Business School Press, (2002)
4. Wenger, E.: Communities of Practice, Learning, Meaning and Identity. Cambridge University press (1998)
5. Wenger, E., White, N., Smith, J. D., Rowe, K.: Technology for Communities. CEFRIO Book Chapter (Jan 2005)
6. Jameson, A.: Numerical Uncertainty Management in User and Student Modeling: An Overview of Systems and Issues. In: User Modeling and User-Adapted Interaction, 5(3-4) (1996) 193-251
7. Dolog, P., Schäfer, M.: Learner Modeling on the Semantic Web?. Workshop on Personalization on the Semantic Web (PerSWeb05), Edinburgh, UK (Jul 2005)
8. Chen, W., Mizoguchi, R.: Communication Content Ontology for Learner Model Agent in Multi-agent Architecture. In: Proc. AIED99 Workshop on Ontologies for Intelligent educational Systems, Le Mans, France (1999)
9. IEEE: Draft Standard for Learning Technology - Public and Private Information (PAPI) for Learners (PAPI Learner). IEEE P1484.2/D7 (11/28/2000)

[3] Some examples can be viewed at http://www-sop.inria.fr/acacia/project/palette/PAKM06

10. IMS Global Learning Consortium Inc.: IMS Learner Information Package Specification. Available on line: http://www.imsglobal.org/profiles/index.html (2001)
11. Paquette, G.: La modélisation des connaissances et des compétences, pour concevoir et apprendre. Presses de l'Université du Québec (2002)
12. I/T Competency Group: MIT I/T Competency Model. Working draft of the MIT, http://web.mit.edu/ist/competency/model.html (1996)
13. Harzallah, M., Berio, G., Vernadat, F.: Analysis and modeling of individual competencies: Toward better management of human resources. IEEE Transactions on Systems, Man and Cybernetics - Part A, 36(1) (Jan 2006) 187-207
14. Lazaric, N., Thomas, C.: The Coordination and Codification of Knowledge Inside a Network, or the Building of an Epistemic Community: The Telecom Valley Case Study. In Understanding the Dynamics of a Knowledge Economy. Edward Elgar Publishing, Cheltenham, UK (2006)
15. Engeström, Y.: Learning by expanding: An activity-theoretical approach to developmental research. Orienta-Konsultit Oy, Helsinki (1987)
16. Laferrière, T.: Preface of "Collaborer pour apprendre et faire apprendre - La place des outils technologiques". Deaudelin & Nault. Presses de l'Université du Québec (2003)
17. Montiel-Overall, P.: Toward a theory of collaboration for teachers and librarians. Research Journal of the School Library Media Research, 8 (Apr 2005)
18. Crowston, K., Osborn, C.: A coordination theory approach to process description and redesign. Working paper of the MIT, Cambridge, USA (1998)
19. Weber, R., Aha, D.W., Becerra-Fernandez, I.: Categorizing Intelligent Lessons Learned Systems. In: Aha and Weber (eds): Intelligent Lessons Learned Systems. Technical Report WS-00-03. AAAI, Menlo Park, California, USA (2000), p. 94-100.
20. Ahmad Sharif, M.N., Mohamad, K.M., Alias, R.A., Shahibudin, S., Zakaria, N.H.: Knowledge Management Framework For Representing Lesson Learned System For Communities of Practice In Institutions of Higher Learning. Malaysian Journal of Computer Science, 17(1) (Jun 2004) 1-12
21. Malvache, P., Prieur, P.: Mastering corporate experience with the Rex method. In: J.-P. Barthès (ed), Proc. of ISMICK'93, Compiègne, France (Oct 1993) 33-41
22. Eichenbaum, Ch., Malvache, P., Prieur, P.: La maîtrise du Retour d'Expérience avec la méthode REX. Performances Humaines et Techniques, 69 (Mar-Apr 1994) 6-20
23. Corbel, J.-C.: Méthodologie de retour d'expérience : démarche MEREX de Renault. In : Fouet (ed) : Connaissances et Savoir-faire en entreprise, Hermès (1997) 93-110
24. Golebiowska, J.: Exploitation des ontologies pour la mémoire d'un projet-véhicule, Méthode et outil Samovar. Ph.D. in Computer Science, UNSA, France (Feb 2002)
25. O'Hara, K., Alani, H., Shadbolt, N.: Identifying Communities of Practice: Analysing Ontologies as Networks to Support Community Recognition. In: Proc. of the IFIP World Computer Congress (IFIP 02), Montreal, Canada (2002)
26. Bettahar, F., Moulin, C., Barthès, J.P: Ontologies Support for Knowledge Management in E-Government Environment. In Proc. of ECAI'06 Workshop on Knowledge Management and Organizational Memories, Riva del Garda, Italy (2006)
27. Floyd, C., Ulena, S.: On Designing Situated Ontologies for Knowledge Sharing in Communities of Practice. Proc. of the CAiSE'05 Int. Workshop on Philosophical Foundations of Information Systems, Porto (2005)

Designing a Knowledge Management Approach for the CAMRA Community of Science

Rosina O. Weber, Marcia L. Morelli, Michael E. Atwood, and Jason M. Proctor

College of Information Science & Technology, Drexel University
3141 Chestnut Street, Philadelphia, PA, USA 19104
{rw37, mlm742, atwood, jp338}@drexel.edu

Abstract. CAMRA (Center for Advancing Microbial Risk Assessment) gathers a community of scientists that investigate several stages in the life cycle of biological agents of concern. This paper describes the knowledge management (KM) approach adopted for CAMRA's community of scientists. The approach includes knowledge facilitators, a web- and repository-based KM system, and use-centered design. The approach relies on a KM methodology that addresses the most common causes of failures in KM approaches that was complemented with a use-centered design methodology. The resulting combined methodology represents a unique way of implementing KM to promote knowledge sharing and collaboration. We describe the principles in our design and the initial steps undertaken to implement it for CAMRA. We conclude by laying out our future steps.

Keywords: Capturing and securing knowledge, Case-based reasoning, Collaboration, Human-computer interaction, Knowledge repository, Knowledge sharing, Web-based knowledge applications.

1 Introduction

Knowledge management (KM) refers to the manipulation of knowledge assets as a means to improve organizational processes. KM approaches include the resources, methods, and instruments to deliver KM goals. CAMRA is a consortium of seven universities, including several investigators from each university. This consortium is an example of the model that government agencies are adopting for grant funding. This model makes knowledge sharing even more crucial than ever to guarantee meaningful results from the funding. One reason is that geographically dispersed investigators pose the need of support for remote collaboration. Furthermore, strategies to promote sharing and collaboration are the only guarantee that funds will yield the intended results, i.e., that the overall result is greater than the sum of its parts. Hence, the KM goals for CAMRA are knowledge sharing and collaboration.

The approach for CAMRA is based on a repository-based KM system, one of the main categories of KM initiatives [10]. Although this type of system has been around for decades, such as best practices and lessons-learned repositories, many of its implementations have failed to demonstrate success [28]. For this reason, several authors (e.g., [5], [11], [13], [21]) have investigated causes for those failures,

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 315 – 325, 2006.

suggesting failure factors for KM approaches. Based on these factors, a methodology that attempts to overturn the effect of those failure factors was proposed by Weber [26]. This methodology focuses on knowledge sharing with strategies being implemented both computationally and by knowledge facilitators. In pursuing a successful KM approach for CAMRA, we complemented these strategies with use-centered design. Use-centered design is a method from human-computer interaction (HCI), a field that focuses on the user perspective.

The next section describes common failure factors in KM systems, followed by the resulting principles we adopted in designing the KM approach for the CAMRA community of scientists. In Section 3, we describe the use-centered cycle and, in Section 4, we describe how we implemented it. In Section 5, we conclude and present plans for future work.

2 Overcoming Failure Factors for KM Systems

In this section, we indicate the areas where failure factors most commonly occur and then describe the principles in our KM approach. There is not a perfect correspondence between failure factors and principles because factors originating in technology may be addressed with human participation and vice-versa.

2.1 Failure Factors

A general cause of failure in KM approaches has been proposed by Abecker, Decker, and Maurer [1]. Approaches that do not integrate humans, technology, and processes are likely to fail [1]. Another general cause of failure was suggested by the difficulty in measuring knowledge [3], which makes it difficult to measure knowledge sharing. The failure factor stems from the lack of trust in endeavors that cannot demonstrate their effectiveness.

Failure and success of organizational efforts are strongly influenced by management actions. Marshall, Prusak, and Shpilberg [18] introduced a series of responsibilities community leaders must exercise with respect to knowledge creation. Holsapple and Singh [17] group those responsibilities into four categories: leadership, control, coordination, and measurement. These relate to the key elements (humans, technology, processes, and evaluation) described above. The key leadership practice when implementing a KM approach is to support it [11]. Skepticism is easily spread to all community members [19], particularly if it comes from the leaders.

Repository-based KM systems were originally adopted after the purchase of text database systems [10]. However, long texts are time consuming to read and may be difficult to interpret [4][27]. Moreover, free text typically lacks (or hides) essential elements that constitute a knowledge artifact, thus only information can be shared. The field of knowledge engineering has several decades studying methods to represent knowledge. Their absence is likely to prevent knowledge sharing.

The design and integration of KM systems is another determining element in knowledge sharing. KM systems that are not integrated into the community processes pose several problems [6][24][27]. Standalone systems that rely solely on pull methods are generally less effective since they rely on users to initiate search behavior, what may be prevented due to several reasons [27].

It is through collaboration that humans typically share. A KM approach has collaboration as an implied goal because it is a condition that promotes sharing while sharing also encourages collaboration. Transparency is an essential element for collaboration [23], thus lack of transparency tends to hinder knowledge sharing.

A well-known failure factor in repository-based KM systems is the difficulty to motivate members to contribute artifacts. For many workers, contributing to a KM system adds an additional task to already tight workloads. Community members do not typically find they are properly compensated for the time allocated to submitting knowledge. Incentives from leaders and the participation level of other community members both influence users' perceived payoff [9]. Furthermore, when members contribute artifacts, they are exposing themselves to the community. Their contributions may be perceived as extreme in some dimension [11]. This suggests that communities that are not culturally tolerant are more likely to discourage sharing thus favoring failure.

Recognizing the community's culture is crucial. Therefore, KM approaches that ignore the targeted community culture are very likely to fail. All categories of stakeholders have something to contribute and so do all community's processes. However, simply targeting a community by building a monolithic organizational memory is also prone to failure [2].

Among studies on impediments to knowledge sharing, Szulanski [25] presented four important impediments that constitute failure factors. Two impediments relate to incompleteness. The first refers to the limitation of an artifact that is created from one originating event only. The second is the lack of facts that validate the quality of an artifact. The two other impediments are caused by the user who accesses an artifact and with whom knowledge is supposed to be shared. The third and fourth impediments occur when the user is not sufficiently knowledgeable of the subject matter. This user will have limited conditions to absorb and then to retain shared knowledge. Szulanski [25] refers to these two last impediments as lack of *absorptive* and *retentive* capacities.

2.2 Principles in Designing the KM Approach

The design principles in our KM approach are based on the general strategies described by Weber [26], implemented both computationally and by human facilitators. One distinguishing element of the approach is its targeted community.

2.2.1 The Targeted Community of Science

We propose the term *community of science* (CoS) to describe a group of scientists that is joined by a research project or department, who share the same goals as defined by their common project or department. Because junior scientists are supervised by a senior member, addressing their needs is not contemplated by the approach but by their supervisors. Examples of a CoS are investigators that are funded by the same grant and faculty in the same department. Although their background interests may be multidisciplinary, their goals are intrinsic to the grant or department they belong.

A CoS shares similarities and differences with *communities of practice* [29]. Communities of practice (CoP) unite members because of a common topic or practice [15], tending to be permanent. A CoS unites a group of members who are research scientists and are united by the goals of a grant or employment in a department for the duration of the relationship.

A CoS also shares similarities and differences with *communities of interest* [5]. When defined in the context of design tasks [5], communities of interest (CoI) are interdisciplinary groups that are united for the duration of a specific project or task [15]. Nevertheless, the web has adopted a looser connotation for CoI as a group that is united by a common interest. This is observed by the solicitation of members to join communities of interest and the definition in Wikipedia: "*A Community of interest is a community of people who share a common interest or passion*[1]".

Although sharing the temporal characteristic of the first definition of CoI, a CoS has exclusive characteristics that make it unique when designing a KM approach to support it. They have innovation as their role in science, becoming culturally tolerant to criticism when members expose themselves trying to share knowledge. The cultural practice among members of CoS is to be reviewed by their peers, which organizes them in flat hierarchies, making them culturally adequate to accept and support KM activities [26]. Members of a CoS are motivated to share their findings, making it easier to encourage them to do it in a systematic way. Further studies on the associations between CoI, CoP, and CoS are needed.

2.2.2 Technological Principles

Technology is utilized in the development of a repository- and web-based KM system. The most important concept in our design is the knowledge artifact we tailored for the CoS. We wanted to ensure that the repository retained knowledge artifacts and not information artifacts. A knowledge artifact must include the minimal elements for a user to make a decision to solve a problem, and be easily interpretable [4]. Otherwise, an information artifact distributes information and the user has to rely on his or her own knowledge to make a decision, producing inconsistent results for the community. Furthermore, knowledge artifacts should adopt a knowledge-based representation formalism. Weber and Aha [27], motivated by the definition of lessons-learned, proposed a knowledge-based representation for knowledge artifacts. Such representation allows the adoption of the case-based reasoning methodology for retrieval and reasoning.

This representation consists of two indexing elements that allow the assessment of when an artifact is applicable; and two reusable elements. Indexing elements are tasks or processes, and state variables describing the context of applicability of the artifact. The reusable elements are the core of the knowledge artifact, i.e., the strategy that will enable a decision to be made to solve a problem; and the fact that substantiates its validity.

For a CoS, we tailored this knowledge artifact and call it a *learning unit* (LU). In its origin, it is similar to a lessons-learned, but its elements are associated with processes typical of a CoS, e.g., research contributions. LUs have four core elements: 1) *Research Activity* is the activity a user is to be performing for the artifact to be

[1] Retrieved on 2 October, 2006, from http://en.wikipedia.org/wiki/Community_of_interest

useful. 2) *Contexts* are the conditions that determine that the artifact is applicable for the research activity. 3) *Contribution* is what was learned from performing the research activity. 4) *Rationale* refers to a supporting evidence for the contribution.

LUs are objective, explicit, and easy to interpret. They make the knowledge of a member easy to understand and interpret. This transparency promotes collaboration [23]. LUs are meant to be exchanged among scientists that are members of a CoS. They are not meant to be used to communicate scientific advances to the public in general.

LUs allow the system to manipulate research contributions for the creation of performance reports. Thus, we are currently testing the design of reporting tools in order to offer an explicit compensation as an incentive to users to allocate time to contribute LUs. Reporting tools for a CoS are particularly attractive because scientists are asked to report about their work on different formats, usually with the same contents. The commitment to the community members is they only need to communicate things to the system once.

The benefit of using this representation for knowledge artifacts is that LUs can be retrieved under the case-based reasoning (CBR) paradigm, in a fashion that resembles the way humans find relevant artifacts. In CBR, cases represent problem-solution pairs, thus case retrieval aims at finding a similar problem with the purpose of retrieving a useful solution. In a LU, a problem is described by a research activity plus its context. Thus, searching for a relevant contribution is about finding units that have the same or similar research activity, having contexts as a secondary attribute. For example, in the context of scientists working with microbes from different perspectives, finding all contributions that relate to the same microbe (e.g., *E. coli*) are likely to produce less relevant results than finding contributions describing the same research activity the scientist is current involved (e.g., modeling microbial fate). Typical keyword search methods, for example, would include this paper in the results for *E. coli*.

2.2.3 Human Principles

In designing our approach, human facilitators are responsible for complementing and compensating for the limitations of technology. They also play the role of mediators to help enforce managerial responsibilities. They educate users about the need for KM, the goals of the approach, and the technological principles. Knowledge facilitators help community members understand what knowledge will be useful and at which level of specificity serving as a mediator for integration. Knowledge facilitators review submitted LUs to verify compliance with the four core fields, so that they will keep the characteristics that result in benefits to the community. When interacting with community members, they discuss the KM system collecting feedback and suggestions.

3 Designing a Usable KM System

The principles in the last section are still short in fully addressing the human failure factors. Since our KM system is designed for humans to enter and retrieve artifacts, we must consider how it can be designed to be easy to use while supporting the goals and tasks of its users. Several authors (e.g. [22], [24], [30]) have suggested that knowledge management efforts are more successful when the system is initially designed with the users in mind. We also need a method to guarantee that the design process will include all the CAMRA stakeholders. Since we needed to strengthen our focus on the users and stakeholders, we included user- and use-centered methods from HCI targeted at helping us reduce difficulties that may be encountered by the users of our KM system.

3.1 User-Centered Design

As early as 1986, Norman and Draper [20] defined the user-centered design process for software applications. They suggested that users can be more successful with software if their needs are placed at the center of the design. Other practitioners and researchers have continued to define this concept and some key principles have emerged. Hix and Hartson [16] summarize two of these: understand the characteristics of different types of users and their tasks, and get actual future users of the system involved as early in the design as possible. Prototypes are commonly used in user-centered efforts to provide a common ground through which designers and users can communicate.

3.2 Use-Centered Design

We argue that a user-centered approach is necessary to the deployment of successful KM systems, but that it is not sufficient. Deploying a successful system requires a deep understanding not only of the application domain, but also of the practice of the people who will use the system [8]. In order for useful systems to be built in these conditions, system design must be based on a process of mutual education [12] between system builders and users. In other words, we must center not on *users,* but on *use* [7].

The goal of mutual education is to allow all members of the design team to contribute their expertise to the system building process. The most important resource in the system building process is an artifact that all stakeholders can understand and contribute to, and which serves to build a sense of community among the stakeholders.

To do this, we propose an evolutionary model of software development in which the expected utility of a system is improved by active, computer-based support for 1) feedback from users of prototype systems to developers, 2) communication and collaboration among users and developers, and 3) mutual education among user and developer communities. An overview of this “seeding, evolutionary growth, reseeding” cycle [14] is shown in Figure 1. Working with users, the design team (which include knowledge facilitators) develops an initial prototype for the system, which is then seeded into the use environment. Over time, as this prototype is used, new uses are noted and new ideas emerge. These are incorporated into a subsequent prototype and this seeding, evolutionary growth, reseeding cycle continues. The key to making this process succeed is recording the intent of design decisions and then observing how well actual use matches that intent. We call this technique *design intent* [7].

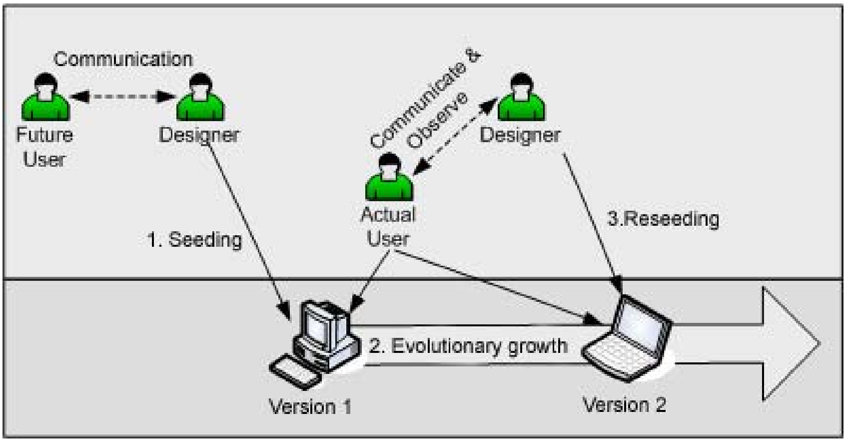

Fig. 1. Use-centered cycle

4 Implementing Use-Centered Design for CAMRA's KM System

The sections below describe the use-centered steps taken in the design and refinement of the KM system, being developed for the CAMRA community.

4.1 Seeding: The Initial Design Process

Before any actual user interface design began, we met with CAMRA scientists and collected user requirements for the system. The information from these early visits also provided us with a basic understanding of the user community that helped guide our early design efforts. We then used a paper prototyping process to develop the initial user interface screens. The team was able to meet with many members of the community during a project kickoff meeting, which provided the opportunity to demonstrate the user interface and to collect preliminary usability data based on common usage scenarios.

4.2 Evolutionary Growth: Findings from Reviewing the Paper Prototype with the CAMRA Community

At the kickoff meeting for the CAMRA project, all users were presented with an overview of the purposes and high-level design concepts for the repository. However, as we began walking the users through the paper prototype, the system became *real* to the users, since they were shown the system in the context of actual tasks they would be expected to complete. This process revealed a number of design challenges and user concerns about the use and purpose of the KM system that might have otherwise gone undetected until after the system was deployed, and we have already implemented changes to overcome the issues that were raised. We were also able to gather some information about the work practices and mental models of the future users of the system.

4.2.1 Entering Learning Units

The main function of the system is the ability to document scientific advances through LUs, which represent single pieces of knowledge that can be shared. We realized early in the design process that it would be critical to make the LU entry process as easy and efficient as possible.

According to our original concept, there are three different types of LU. *Findings* units were meant to store the knowledge obtained by experimental work conducted as part of CAMRA. *Ongoing* units were to be submitted during the research activity. Units labeled *Literature Review* were designed to collect contributions from published literature.

As we presented the original design for the LU screens, it became immediately apparent that while users grasped the general idea and purpose of a LU, they struggled to understand the distinctions we were making between the different types of units as well as between the four core fields that make up the key information in each unit. In some cases, the fields in the types Ongoing and Findings strongly reminded users of the sections of a research paper (i.e. task, results, conclusions, etc.), and they kept trying to match the required fields with that model. The Literature Review label also created some confusion, since it reminded some users of the research activity of analyzing broad sets of published literature, rather than the collection of specific background and prior research.

We addressed this problem by adopting a plain English approach to the types of LU and to the labels displayed in the user interface. For example, *Research Activity* became "What is the general research activity?"; *Contexts* became "In what contexts does this activity occur?"; and *Contribution* became "What is the contribution you learned?". Another confusion we addressed was the field *Rationale*, which seemed to users to support the research activity, rather than the contribution of the unit. Therefore, we adopted "Summarize your results" to ask for the results that validate the contribution. The three types of LUs were re-titled and became "Things I have completed," "Things that are in progress," and "Things I have read."

In our original design, we asked for the contribution and then for its justifying results. During the prototype review, the CAMRA scientists commented that they expected to enter results first and then the contribution since this is how they are used to describing their work. We revised the design accordingly.

4.2.2 Sharing Learning Units of Interest

Our approach includes two distribution methods: push and pull. We implement push via active casting [28] with our recommendation system. The system asks the contributor of a LU to identify users who might be interested in receiving a notification about the new LU. These users are then notified by email about the new LU.

Initial user response to this feature has been positive, although the success of these notifications depends on the contributor's ability to know what types of interests other community members possess. For this reason, we incorporated research activities of interest to users in the list of users shown when the link *Find Users* (Figure 2) is clicked. Furthermore, users also suggested the inclusion of a customization option to enter research activities and contexts of interest so that units containing those would be cast to them. This way they would not have to depend solely on the choice of the unit contributor.

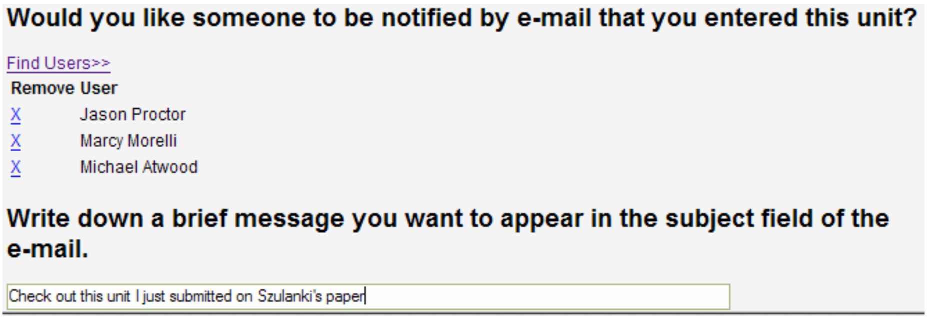

Fig. 2. Users can type a subject line when recommending learning units

Users also suggested having the option to type a personal subject line to appear as the subject line of the email (Figure 2). This feature will personalize the message, helping the user receiving the notification distinguish it from other messages.

The pull method of distributing LU takes advantage of the representation that is amenable to retrieve units using the case-based reasoning paradigm, as previously explained. Although mostly only the research activity and the contexts are indexing components, aspects from the results or contribution may also be used to assess the similarity between an activity being searched and stored units. In fact, users requested for a comprehensive keyword search capability as well. For example, they would like to be able to find all units that reference a given author under *Things I have read*.

4.3 Reseeding: Matching the Design to the Practice

The CAMRA community is already using the first release of the system, but our design efforts do not end. As the system is used, we expect the work practices of the users to change. While most design efforts stop once the system is delivered, we argue that the most important design decisions are made after the initial delivery. For this reason, we are using design intent to monitor any changes in practice that should result in redesigns. Human facilitators are meeting with users from the community to guide them on how to use the system and to collect relevant feedback.

5 Concluding Remarks and Future Work

The experience in designing and developing a KM approach as we describe in this paper is unusual in many aspects. We refer to communities of science that have specific characteristics that are not shared by CoP or CoI. The CAMRA community is an example of a CoS, which has advantages of being supportive of innovation, being flatly organized, and for its tolerance to positive criticism.

One of the ways we will evaluate the CAMRA knowledge repository is by observing associations. After a member associates a new unit with an existing unit authored by a different contributor, he or she is asked to explain the nature of the association. Members are oriented by knowledge facilitators to seek for associations that require their expertise to be recognized and explained. If the member who describes the association did not know about the existing unit before that session, then the assumption is that knowledge sharing took place, i.e., knowledge from the existing unit was transferred to the contributor of the new unit. However, it is possible that this member has heard about the work on the existing unit by other means, preventing us from concluding knowledge sharing from the association. Thus, we plan to ask members whether the association step caused knowledge to be shared.

Another situation where we can ask members whether knowledge was shared is when a contributor notifies other users of a new unit. In the first stage, the member who has been notified has to actually click to read the new unit. Secondly, after some time, a pop up window can ask this member whether knowledge sharing has occurred. This alternative will require us to determine an amount of time that is acceptable for one to understand a unit. Additional evaluations of this system confirming its ability to overcome failure factors will include formal usability tests, analysis of usage logs, and interviews and surveys of system users.

We are currently undergoing another stage of reseeding that will produce another version of the tool. This next version will incorporate the two capabilities to find units and reporting.

Acknowledgements

This work is supported in part by the U.S. EPA-Science to Achieve Results (STAR) Program and the U.S. Department of Homeland Security Programs, Grant # R83236201. The authors would like to thank CAMRA community's members, who have provided constructive feedback as part of the system design process, PAKM 2004 reviewers, and Mr. Nicholas M. Sillik, our programmer.

References

[1] Abecker, A., Decker, S., Maurer, F.: Guest Editorial: Organizational Memory and Knowledge Management. *Information Systems Frontiers* 2 (2000) 251–252

[2] Ackerman, M. S., Hadverson, C. A.: Reexamining Organizational Memory. *Communications of the ACM* 43 (2000) 58–64

[3] Ahn, J.-H., Chang, S.-G.: Valuation of Knowledge: A Business Performance-Oriented Methodology. In: *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE Press, Los Alamitos CA (2002) 2619–2626

[4] Alavi, M., Leidner, D. E.: Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues. *MIS Quarterly* 25 (2001) 107–136

[5] Arias, E., Eden, H., Fischer, G., Gorman, A., Scharff, E.: Transcending the Individual Human Mind—Creating Shared Understanding through Collaborative Design. *ACM Transactions on Computer-Human Interaction* 7 (2000) 84–113

[6] Atwood, M. E.: Organizational Memory Systems: Challenges for Information Technology. In: *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE Press, Los Alamitos CA (2002) 919–927

[7] Atwood, M. E., Burns, B., Gairing, D., Girgensohn, A., Lee, A., Turner, T., Alteras-Webb, S., Zimmermann, B.: Facilitating Communication in Software Development. In: Olson, G. M., Schuon, S. (eds.): *Proceedings of the Conference on Designing Interactive Systems: Processes, Practices, Methods, & Techniques*. ACM Press, New York (1995) 65–73

[8] Bjerknes, G., Bratteteig, T., Kaasboll, J., Nygaard, K., Sannes, I., Sinding-Larsen, H., Thingelstad, G.: Gjensidig Laerning: Florence Rapport Fra Fase 1 [Mutual Learning: Florence Report No. 1]. Department of Informatics, University of Oslo, Norway (1985)

[9] Chua, A.: Knowledge Sharing: A Game People Play. *Aslib Proceedings* 55 (2003) 117–129
[10] Davenport, T. H., Prusak, L.: *Working Knowledge: How Organizations Manage What They Know*. Harvard Business School Press, Boston (1998)
[11] Disterer, G.: Individual and Social Barriers to Knowledge Transfer. In: *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, Vol. IEEE Press. Los Alamitos CA (2001)
[12] Ehn, P.: *Work-Oriented Design of Computer Artifacts*. Arbetslivscentrum, Falköping, Sweden (1988)
[13] Fahey, L., Prusak, L.: The Eleven Deadliest Sins of Knowledge Management. *California Management Review* 40 (1998) 265–276
[14] Fischer, G.: Seeding, Evolutionary Growth and Reseeding: Constructing, Capturing, and Evolving Knowledge in Domain-Oriented Design Environments. *Automated Software Engineering* 5 (1998) 447–464
[15] Fischer, G., Ostwald, J.: Knowledge Management: Problems, Promises, Realities, and Challenges. *IEEE Intelligent Systems* 16 (2001) 60–72
[16] Hix, D., Hartson, H.: *Developing User Interfaces: Ensuring Usability through Product & Process.* John Wiley & Sons, Inc., New York (1993)
[17] Holsapple, C. W., Singh, M.: The Knowledge Chain Model: Activities for Competitiveness. *Expert Systems with Applications* 20 (2001) 77–98
[18] Marshall, C., Prusak, L., Shpilberg, D.: Financial Risk and the Need for Superior Knowledge Management. *California Management Review* 38 (1996) 77–101
[19] Nonaka, I., Konno, N.: The Concept of "Ba": Building a Foundation for Knowledge Creation. *California Management Review* 40 (1998) 40–54
[20] Norman, D. A., Draper, S. W.: *User Centered System Design: New Perspectives on Human-Computer Interaction*. Lawrence Erlbaum Associates, Hillsdale NJ (1986)
[21] Quinn, J. B., Anderson, P., Finkelstein, S.: Managing Professional Intellect: Making the Most of the Best. *Harvard Business Review* 74 (1996) 71–80
[22] Smalley, P., Herman, J.: Creating a System to Share User Experience Best Practices at eBay. In: *CHI '05 Extended Abstracts on Human Factors in Computing Systems*. ACM Press, New York (2005) 1797–1800
[23] Stahl, G.: *Group Cognition: Computer Support for Building Collaborative Knowledge.* MIT Press, Cambridge MA (2006)
[24] Stenmark, D., Lindgren, R.: Integrating Knowledge Management Systems with Everyday Work: Design Principles Leveraging User Practice. In: *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*. IEEE Press, Los Alamitos CA (2004)
[25] Szulanski, G.: Exploring Internal Stickiness: Impediments to the Transfer of Best Practice Within the Firm. *Strategic Management Journal* 17 (1996) 27–43
[26] Weber, R. O.: Addressing Failure Factors in Knowledge Management. *Electronic Journal of Knowledge Management* (under review)
[27] Weber, R., Aha, D. W.: Intelligent Delivery of Military Lessons Learned. *Decision Support Systems* 34 (2003) 287–304
[28] Weber, R., Aha, D. W., Becerra-Fernandez, I.: Intelligent Lessons Learned Systems. *Expert Systems with Applications* 20 (2001) 17–34
[29] Wenger, E.: *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, Cambridge, UK (1998)
[30] Zimmermann, B., Atwood, M. E., Webb, S., Kantor, M.: The Knowledge Depot: Building and Evaluating a Knowledge Management System. *Educational Technology & Society* 3 (2000) 137–149

Knowledge Management for a Large Service-Oriented Corporation

Sylvia C. Wong, Richard M. Crowder, Nigel R. Shadbolt, and Gary B. Wills

School of Electronics and Computer Science,
University of Southampton, UK
{sw2, rmc, nrs, gbw}@ecs.soton.ac.uk

Abstract. The design and maintenance of complex engineering systems such as a jet engine generates a significant amount of documentation. Increasingly, aerospace manufacturers are shifting their focus from selling products to providing services. As a result, when designing new products, engineers must increasingly consider the engine's complete life-cycle as part of the design process. To identify possible areas of concern, engineers must obtain knowledge gained from the entire life of similar engines. However, because of the size and distributed nature of the company's operation, engineers often do not have access to front-line maintenance data. In addition, the large number of documents accrued makes it impossible for them to be examined thoroughly. This paper presents a prototype knowledge-based document repository for such an application. It searches and analyzes distributed document resources, and provides engineers with a summary view of the underlying knowledge. The aim is to allow engineers to incorporate maintenance issues into the initial design. Unlike existing document repositories and digital libraries, our approach is knowledge-based, where users browse summary reports instead of following suggested links. To test the validity of our proposed architecture, we have developed and deployed a working prototype.

1 Introduction

The design and maintenance of large and complex engineering systems requires a significant amount of documentation, particularly if the system being considered is a turbofan jet engine used on the current generation of aircraft. The jet engine is amongst the most complex machine ever designed, incorporating a wide range technologies of including high temperature materials, complex fluid dynamics and high speed rotating components.

A fundamental shift is currently occurring in the aerospace industry away from selling products to providing services. Companies such as Rolls-Royce aims to make half its engine fleet subject to long-term maintenance service agreements by 2010 [1]. Essential to the success of this market shift is the significant cultural change from *offering a service to support a product* to *designing a service and the product to support it* [1]. In other words, new products must be designed to provide lower and more predictable maintenance costs. To minimize maintenance costs through out the engine's life cycle, engineers must obtain knowledge

U. Reimer and D. Karagiannis (Eds.): PAKM 2006, LNAI 4333, pp. 326–337, 2006.

gained from maintenance histories of similar engines during the design phase of a new engine. This will help engineers identify parts most likely to be problematic throughout the new engine's entire life cycle, possibly eliminating the problem during the design phase. It should be noted that engine design is typically undertaken by a number of teams who are responsible for individual engine modules, e.g compressor, turbine. Therefore it is impossible for any single member of a design team to access more than a fraction of available documentation. As is widely recognized, information systems usually develop over time into a set of heterogeneous resources. As a result, it becomes difficult for engineers to follow a trail through these resources [2]. The challenge for organizations is therefore to develop an information system that is both comprehensive and will satisfy the increasing demands from industry for up-to-date and easily accessible information.

In response to these challenges, we are implementing an intelligent, knowledge repository to support engineers to design for better performance and lower maintenance costs throughout the engine life-cycle. The knowledge repository searches and analyses relevant maintenance records and design guidelines, and provides design engineers easy access to such information. It is hoped that the summary reports provided by the intelligent document repository will help engineers in creating design documents that incorporate these issues into the design requirements.

This paper is organized as follows. Section 2 explains how knowledge can be reused within an aeroengine manufacturing company. It also introduces a scenario explaining how maintenance records can be used to help to improve the reliability of both existing and new products. Section 3 gives a brief overview of knowledge management. Section 4 then discusses other works that aims to help user to discover knowledge by integrating heterogeneous documents sources. Section 5 introduces our proposed architecture for an intelligent document repository. This is followed by Sections 5.1 to 5.3, which describes the underlying components of the proposed document repository. A discussion is presented in Section 6 with conclusions in Section 7.

2 Scenario

As is well recognized in engineering design, the use of past experiences and previously acquired knowledge, either from the designer's own experiences or from resources within their organization forms an important part of the design process. It has been estimated that 90% of industrial design activity is based on variant design [3], while during a redesign activity up to 70% of the information is taken from previous solutions [4]. A cursory consideration of these figures identified two immediate challenges — how to capture knowledge, and how to retrieve it. The purpose of the work reported in this paper is to develop an information system that can be used within a manufacturing organization for the retrieval of knowledge from across the organization to support the design activities.

The following scenario illustrates the potential use and benefit from such an information system. The scenario involves three separate and different groups of users that are involved in the life of a jet engine. Front-line maintenance engineers are involved in the day to day servicing of the engines, and thus responsible for populating the document repository with maintenance reports and other similar documents.

> *During the regular pre-flight checks, a flight crew reported a problem caused by a leak from an engine's bleed air system. Subsequent inspection which required the removal of the engine revealed that a duct had failed at a joint due to vibration. After repair, the engine was returned to service, and a full maintenance event report submitted to a document repository.*

Traditionally, after production and sales, responsibility of maintenance of the engine passed from the manufacturer, to the airlines, who own the engines. This maintenance activity is supported by the manufacturers technical support and operations team. Technical support and operation engineers are responsible for improving the performance of existing engines. They can use information collected in the repository to monitor trends that develop over a fleet of engines. Modification can then be designed to mitigate any problems found:

> *Following a review of the maintenance events relating to a specific engine fleet, a trend was noticed in the high than expected number of failure of a bleed air duct joint due to vibration. To maintain the reliability of the engine fleet, a modification was developed and implemented.*

Design engineers are currently remote from the problems experienced in the field by operations. Due to the importance of increasing operational reliability and minimizing maintenance costs in the new market paradigm of product support, information gained in the operation of a fleet of engines needs to be fed back to the designers of subsequent engines. However, the current information infrastructure makes this difficult as design engineers do not have access to maintenance knowledge. Our work aims to strengthen and help formalized the information flow between aftermarket operations and the design teams. It allows knowledge gained during the design, production and operation of an engine to advice the design of the next variant:

> *The design team for the next variant of this engine reviews the performance of the air bleed system across the fleet to learn from previous design rationale and operational history. The designers noticed a higher than predicted failure rate on a particular joint. Subsequent finite element analysis showed that a joint failure could occur due to vibration if certain operational conditions were met. It was therefore decided that the future variant of the engine would both eliminate the joint and reroute the duct work. The revised design costs 50% more than the original. However, the saving over the life of the engine will be substantial due to lower likelihood of in-service failure.*

The goal of our work can thus be summarized as follows: To feedback and harvest knowledge gained from the aftermarket operations documents to help (a) operations engineers in designing modifications to existing engines, and (b) design engineers in designing the next variant engine for the aftermarket.

3 Knowledge Engineering

Organizations have become increasingly concerned with knowledge management [5], amassing large amount of information into their corporate memory [6]. The aim is to use this repository of information to inform future discussions, decisions and activities. Figure 1 shows the four key activities in knowledge management – knowledge creation, knowledge mapping, knowledge retrieval and knowledge use.

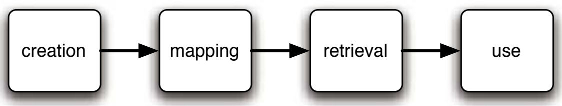

Fig. 1. The four key activities in Knowledge Management

The first step in creating a knowledge system is knowledge creation. In our scenario, this is the documentation written by engineers throughout the life time of an engine, from design to maintenance. In practice it can take many forms, including formal design reports, e-mails submitted by airlines and field service personnel, detailed inspection reports and the engine performance over individual flights recorded by engine monitoring systems.

Knowledge mapping involves creating an *ontology*, which is *a specification of a conceptualization* [7]. Gruber explains that a common ontology defines the vocabulary with which queries and assertions are exchanged among agents (people or software). The ontology sets out all the entities (objects or concepts) that we are interested in and the relationships that connect these entities together. This is intended to be a *pragmatic* definition, i.e. it defines the vocabulary that is actually *in use*, and the concepts that are *useful* in problem-solving. It does not give the deep underlying philosophical vision of the fundamental entities in the field. Hence, in knowledge management, an ontology is a tool, whose quality is entirely dependent on its usefulness.

Retrieval is the step that transform mere *information* into *knowledge*. We believe that knowledge is relevant information delivered at the right time and context [8]. To deliver this knowledge, information needs to be semantically enriched so that it can be better reused. When a knowledge system has a shared ontology for its disparate information resources, software agents can handle the semantically enriched resources consistently. Thus semantics and ontology together help deliver the right information at the right time, hence generating knowledge.

In our scenario, knowledge use occurs when engineers apply the knowledge gained from the repository in their work. For example, a design engineer applies the knowledge gained to create a better engine for the aftermarket. Or an operations engineer discovers a recurring, but minor problem with an existing engine, which would indicate a larger problem then each individual incident suggests.

4 Related Work

The work described in this paper is an extension on our previous work with Rolls-Royce. In [9], we presented a future vision for the working practices of designers within a manufacturing organization. We have found that engineering design environments are highly distributed in nature and are characterized by a large number of information sources. It is concluded that a range of knowledge management tools would be required to support this future vision of engineering design environments [10]. Therefore one of the objectives of our current work is to define a future engineering design environment, with particular emphasis on the social and technical systems that will support designers in their day-to-day activities.

In [2], we created a document repository from distributed and heterogeneous engineering document resources. When an engineer searches for documents within the repository, the system generates a list of documents ordered according to the engineer's role and its related concepts. Thus, the document retrieval process is *intelligent* and adapts to the user's role within the company. However, we found that engineers actually want the knowledge that is buried within the documents, instead of the actual documents themselves. This is due to the large volume of documents available within the repository which makes it very time consuming to peruse thoroughly. In other words, engineers prefers to see *summary reports* of documents archived. For example, when searching for engine part deterioration, engineers want to see how many times the deterioration of a particular part leads to engine removal, but not the list of original maintenance documents. Thus, for our new information system, we aim to include the ability to provide analysis of information stored, in addition to simple document search.

Our work in [2] can be seen as a digital library, with the extension where information presented is adapted to the role of the user. Digital libraries concentrate on the problem of searching for documents distributed over multiple repositories. For example, Priebe and Pernul [11] developed a portal over multiple document repositories by using an integrated metadata store. In contrast, search functionality does not form part of our proposed infrastructure. However, global document searches can be provided to our knowledge repository as *services* that implement document indexes and metadata indexes.

Another area digital libraries concentrate on is dynamic links generation to relevant document resources [12,13]. Dynamic links are injected into documents automatically during presentation time, and does not alter the original documents. These dynamic links can point to related documents, or even services such as searching annotation and peer reviews [13]. In comparison, our proposed system does not perform dynamic link injection on existing documents. However, it can provide a list of suggested documents as a *service*.

Finally, there are also projects working on creating new semantically marked up data for large knowledge repositories, such as [14] and [15]. Creating new documents is outside the scope of our project, as we concentrate on the problem of delivering knowledge from existing documents. However, employing techniques to generate semantic information automatically for new documents to be deposited into the knowledge repository will improve document analysis and searching inside our framework.

5 Architecture

Two key technologies are used in integrating the distributed and heterogeneous data sources available from the document repository – Web Services and the Semantic Web. Web Services are loosely coupled standards-based components that perform specific tasks. Using Web Services, applications can be developed and deployed incrementally. In addition, new features can be easily added after the system is deployed. This modularity and extensibility make it especially suitable as a platform for an integrated knowledge repository within large engineering organizations. The Semantic Web is an application of the World Wide Web aimed at computational agents, so that *programs*, and not just humans, can interpret the meaning of documents on the Web (or an intranet). This allows the Web to be used for more than a human-browseable repository of information. The basis of this interpretation is an ontology, a structure which forms the backbone of the knowledge interpretation for an application.

The system architecture diagram of our knowledge repository is shown in Figure 2. Existing web standards are used wherever possible, to maximize tool reuse, compatibility and portability. A web portal provides an integrated user interface, and handles authentication and workflows. The user interface is formed by a series of portlets — web applications that run within a portal. In general, portlets appear to end users as rectangular sections of a web page offering a limited set of information. The portlets access one or more web services. We envisaged that these web services are provided by different departments within the company, and can be distributed across multiple sites. These web services will perform more than document searches that are already available in all document repositories. For example, it can provide an interface to existing manufacturing systems such as PDM (Product Data Management), or allow access to analysis tools such as Life-Cycle Cost Models and Finite Element Analysis. Documents in the repository use the Semantic Web data format RDF [16], with its schema in OWL [17].

The remainder of this section discusses each component within the architecture in greater details.

5.1 Documents and Ontology

For each maintenance event, maintenance engineers document information surrounding the event for later references. A maintenance event occurs whenever actions are performed on an engine. Usually, each maintenance event involves multiple actions. Information documented includes the circumstances of the event,

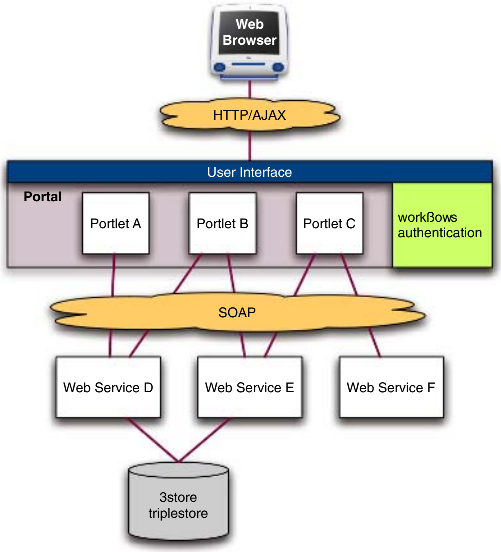

Fig. 2. System Architecture of the proposed knowledge system

actions taken, parts installed and removed and any other findings observed. The engineers who created these documents are located in numerous sites around the world, in the manufacture's or third party repair bases or at airports. As a result, the maintenance documents are in a large variety of formats. Some of these reports are in the form of unstructured Word documents, with different sites using different templates. Other engineers record maintenance information using a centralized Service Data Manager (SDM).

To enable machines to interpret meanings stored within the documents, we need an ontology that captures all the terms and concepts used. Moreover, since the document repository is to be used by both design and service engineers, the ontology should capture concepts from engineers working in both areas. The ontology is created by analyzing existing documents and conducting knowledge acquisition interviews with engineers [2]. The engineers interviewed are carefully selected and are domain experts from several different specialization. During these interviews, the *card sort* technique is used to help the engineer show how they used different document types and the relationships between these documents. The result of these interviews enabled us to identify, by specialism, the main concepts and the associated keyword for these concepts used by the particular type of engineer when searching for information.

The resulting ontology contains concepts ranging from engine deterioration mechanisms, engine models and parts to airport locations. Using this ontology, we have populated the triplestore with maintenance records from both the SDM database and reports provided as Word documents. Currently, our triplestore contains approximately 3250 maintenance events, with around 31,000 actions.

This information equates to 389,000 rows in a SQL database table. The space usage in the table is 12MB in data and 25MB in index (ie total of 37MB).

5.2 Web Portal

Users access our knowledge framework via the web portal. The portal uses username/password authentication, and role-based access control. Using role-based access control, we can customize the content of the portal according to the engineer's specialization. The roles defined in the system can reflect job functions of the engineers. Thus, engineers with different specialization can be served a different set of portal pages. For example, when studying deterioration mechanisms, maintenance engineers can be presented with information from individual engine parts, while design engineers will be presented with overall information for the entire fleet. This difference arises from the different tasks the engineers have to perform. Maintenance engineers are interested in locating and fixing problems of existing engines in service. On the other hand, design engineers are interested in overall performance to aid the design of future variants. Another possible customization can be used for navigation/drill down over the engine taxonomy. Since an engine contains a large number of parts, a simplified taxonomy for navigation will greatly speed up information browsing. The navigation taxonomy can be a subset of the entire engine taxonomy, based on the engineer's specialization.

A questionnaire based study was carried out to better understand what kind of knowledge engineers would like to discover from the knowledge web [18]. In the questionnaire, engineers are presented a list of questions relating to maintenance experience with a product. They are asked how often they might ask them when designing a new product. They are also asked what other questions they might want to ask. The result of this questionnaire tells us what are the most important and most common life cycle information design engineers seek from maintenance documents.

Based on the results of the user study, we developed two portlets that help engineers harvest knowledge from the data sources. The first portlet is a maintenance event browser that allows engineers to follow an engine taxonomy to retrieve lists of maintenance events, view timelines of events and obtain detailed information about a particular event. The second portlet allows engineers to browse a list of parts that dominate maintenance statistics. Figure 3 shows screenshots of the two portlets.

5.3 Web Services

Web services provide processing functionalities that can be access from anywhere on the network. The web services are decoupled from the portlets. In other words, multiple portlets can execute the same web service, and a single portlet can execute multiple web services. We have developed several web services that perform common requests on the maintenance documents. For example, obtaining a list of parts that are involved in the highest number of maintenance events, tracing

(a) (b)

Fig. 3. (a) Maintenance event browser portlet, (b) Most serviced part portlet

the maintenance record of an engine or a part, or retrieving details of a maintenance event. We have also developed a graphing service that returns a histogram or a bar chart for a set of given data. We have also created a document index for the maintenance reports that are in Word format.

6 Discussion

In this paper, we have presented an intelligent document repository for an aero-engine manufacturer. However, the system architecture is generic and can be applied to organizations outside of aero-engineering. Most large organizations generate a substantial amount of documents everyday. This is regardless of the sector the organization operates in, be it healthcare or finance. Our proposed architecture are applicable to document repositories in any industry, where the number of documents are too large to peruse, and that users are interested in knowledge extracted from documents deposited.

Heterogeneous document sources are integrated by a shared vocabulary – an ontology. To answer some of the questions design engineers want to know about maintenance, we need to combine information from multiple sources. For example, to know which parts dominate the cost drivers in the engine, we need to combine reliability data from maintenance event reports with the engine taxonomy and costing information. These data are contained in heterogeneous data sources with different schemas within the company. Integration is made possible with the use of the ontology, which allows software agents to reason over the different resources.

Users access the document repository via a standard web browser. System components are hosted on distributed servers. As a result, the software can be deployed and updated centrally, without changing the configurations of thousands of desktop computers. Also, users can access the document repository without special software installed on their computers.

The two portlets developed demonstrate the concept of 'summary reports' within a document repository. These summary reports correlate information that are present in the underlying heterogeneous documents, but cannot be easily discovered through simple document browsing. For example, the event browser portlet allows operations engineers to follow through the sequence of events that leads to a part's failure. The serviced parts frequency portlet allows design engineers to discover which parts are dominating maintenance statistics and thus require further attention when designing the next variant engine.

7 Conclusions

In this paper, we discussed the development of a knowledge-based document repository. The system allows users from across a company to access knowledge that they need to undertake their activities. In particular, we have developed a prototype for a large aero-engine manufacturer. A fundamental shift is occurring in the aerospace industry away from selling products to providing services. This shift in market focus means that new products must be designed to provide lower and more predictable life cycle costs. To achieve this, engineers must obtain knowledge gained from the entire life of an engine.

However, because of the large size and distributed nature of aerospace manufacturers, engineers often do not have access to front-line maintenance data. In addition, the large number of documents accrued during the life of an engine makes it impossible to examine all information thoroughly. Therefore, unlike existing document repository or digital libraries, our system is knowledge, and not information, based. Our proposed system searches and analyzes relevant maintenance records and design guidelines, and provides engineers with a 'summary report' to such information. It is hoped that the summary reports provided will help engineers in creating design documents that incorporate aftermarket issues into the design requirements.

Two key technologies are used in integrating the distributed and heterogeneous data sources — the Semantic Web and Service-Oriented Architecture. The Semantic Web provides the framework allowing computer programs to interpret and reason over the heterogeneous document sources. The documents are integrated using an ontology, which captures the terms and concepts used in aerospace engineering. Service-Oriented Architecture allows knowledge extraction and analysis functionalities to be added to the system as modules called web services.

To test our proposed architecture, we have developed a working prototype. The prototype consists of a web portal, several web services and a RDF triplestore. Users access the document repository via a web portal, which is customized according to users' roles. The prototype has been demonstrated to Rolls-Royce engineers and has received positive reviews.

Acknowledgments

This research was undertaken as part of the IPAS project (DTI Grant `TP/2/IC/6/I/10292`). The authors would also like to thank the project partners for providing us with data and ontologies. Specifically, we would like to thank Derek Sleeman and David Fowler from Aberdeen AKT for the ontology, and Alymer Johnson and Santosh Jagtap from Cambridge EDC for the user requirement analysis.

References

1. Harrison, A.: Design for service – harmonising product design with a services strategy. In: Proceedings of GT2006, ASME Turbo Expo 2006: Power for Land, Sea and Air, Barcelona, Spain (2006)
2. Wills, G., Fowler, D., Sleeman, D., Crowder, R., Kampa, S., Carr, L., Knott, D.: Issues in moving to a semantic web for a large corporation. In: Proceedings of 5th International Conference on Practical Aspects of Knowledge Management (PAKM). Volume 3336 of Lecture Notes in Artificial Intelligence., Springer (2004) 378–388
3. Gao, Y., Zeid, I., Bardasz, T.: Characteristics of an effective design plan system to support reuse in case-based mechanical design. Knowledge-Based Systems Knowledge-Based Systems Knowledge-Based Systems **10** (1998) 337–350
4. Khadilkar, D.V., Stauffer, L.A.: An experimental evaluation of design information reuse during conceptual design. Journal of Engineering Design **7** (1996) 331–339
5. Shadbolt, N., Milton, N.: From knowledge engineering to knowledge management. British Journal of Management **10** (1999) 309–322
6. Heath, I., Wills, G., Crowder, R., , Hall, W., Ballantyne, J.: Towards a new authoring methodology for large-scale hypermedia applications. Multimedia Tools and Applications **12** (2000) 129–144
7. Gruber, T.R.: A translation approach to portable ontology specifications. Knowledge Acquisition **5** (1993) 199–220
8. Millard, D.E., Tao, F., Doody, K., Woukeu, A., Davis, H.C.: The knowledge life cycle for e-learning. International Journal of Continuing Engineering Education and Lifelong Learning **16** (2006) 110–121
9. Crowder, R., Bracewell, R., Hughes, G., Kerr, M., Knott, D., Moss, M., Clegg, C., Hall, W., Wallace, K., Waterson, P.: A future vision for the engineering design environment: A future sociotechnical scenario. In Folkeson, A., Gralen, K., Norell, M., Sellgren, U., eds.: Proceedings of 14th International Conference on Engineering Design, Stockholm (2003) 249–250
10. Wallace, K.M., Clegg, C., Keane, A.: Visions for engineering design: a multidisciplinary perspective. In: Proceedings of 13th International Conference on Engineering Design, Glasgow, Scotland (2001) 107–114
11. Priebe, T., Pernul, G.: Towards integrative enterprise knowledge portals. In: CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management, New York, NY, USA, ACM Press (2003) 216–223
12. Carr, L., Hall, W., Bechhofer, S., Goble, C.: Conceptual linking: Ontology-based open hypermedia. In: Proceedings of 10th International World Wide Web Conference (WWW), Hong Kong (2001) 334–342
13. Nnadi, N., Bieber, M.: Lightweight integration of documents and services. In: DocEng '04: Proceedings of the 2004 ACM symposium on Document engineering, New York, NY, USA, ACM Press (2004) 51–53
14. Carr, L., Miles-Board, T., Woukeu, A., Wills, G., Hall, W.: The case for explicit knowledge in documents. In: Proceedings of ACM Symposium on Document Engineering, Milwaukee, Wisconsin (2004) 90–98

15. Ciravegna, F., Wilks, Y.: Designing adaptive information extraction for the semantic web in amilcare. In Handschuh, S., Staab, S., eds.: Annotation for the Semantic Web, IOS Press (2003)
16. Manola, F., Miller, E.: RDF primer. Technical report, W3C Recommendation, http://www.w3.org/TR/rdf-primer (2004)
17. McGuinness, D.L., van Harmelen, F.: OWL web ontology language overview. Technical report, W3C Recommendation, http://www.w3.org/TR/owl-features (2004)
18. Jagtap, S., Johnson, A., Aurisicchio, M., Wallace, K.: Pilot empirical study: Interviews with product designers and service engineers. Technical Report 140 CUED/C-EDC/TR140- March 2006, Engineering Design Centre, University of Cambridge (2006)

Author Index