



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Document

D-97-03

Knowledge-Based Systems for Knowledge Management in Enterprises

Workshop held at the
21st Annual German Conference on AI (KI-97)

A. Abecker, S. Decker, K. Hinkelmann, U. Reimer

September 1997

**Deutsches Forschungszentrum für Künstliche Intelligenz
GmbH**

Postfach 20 80
67608 Kaiserslautern, FRG
Tel.: + 49 (631) 205-3211
Fax: + 49 (631) 205-3210

Stuhlsatzenhausweg 3
66123 Saarbrücken, FRG
Tel.: + 49 (681) 302-5252
Fax: + 49 (681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler-Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, Sema Group, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry of Education, Science, Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct systems with technical knowledge and common sense which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ☐ Intelligent Engineering Systems
- ☐ Intelligent User Interfaces
- ☐ Computer Linguistics
- ☐ Programming Systems
- ☐ Deduction and Multiagent Systems
- ☐ Document Analysis and Office Automation.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Dr. Dr. D. Ruland
Director

Knowledge-Based Systems for Knowledge Management in Enterprises

– Workshop held at the 21st Annual German Conference on AI (KI-97) –

A. Abecker, S. Decker, K. Hinkelmann, U. Reimer

DFKI-D-97-03

This work has been supported by a grant from The Federal Ministry of Education, Science, Research, and Technology (FKZ ITWM-9705 C4).

© Deutsches Forschungszentrum für Künstliche Intelligenz 1997

This work may not be copied or reproduced in whole or part for any commercial purpose. Permission to copy in whole or part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of the Deutsche Forschungszentrum für Künstliche Intelligenz, Kaiserslautern, Federal Republic of Germany; an acknowledgement of the authors and individual contributors to the work; all applicable portions of this copyright notice. Copying, reproducing, or republishing for any other purpose shall require a licence with payment of fee to Deutsches Forschungszentrum für Künstliche Intelligenz.
ISSN 0946-0098

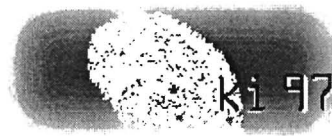
Knowledge-Based Systems for Knowledge Management in Enterprises

– Workshop held at the 21st Annual German Conference on AI (KI-97) –

Andreas Abecker, Stefan Decker, Knut Hinkelmann, and Ulrich Reimer

September 1997

Notes of the Workshop on "Knowledge-Based Systems for Knowledge Management in Enterprises"



held as a part of the:

**21st Annual German Conference on AI '97
(KI-Jahrestagung '97)**

**September 9th - 12th
Freiburg, Germany**

Editors:

■ **Andreas Abecker**

DFKI GmbH, Postfach 2080, D-67608 Kaiserslautern
Tel: +49 631 205-3456; Fax: +49 631 205-3210
e-mail: aabecker@dfki.uni-kl.de
WWW: <http://www.dfki.uni-kl.de/~aabecker/>

■ **Stefan Decker**

University of Karlsruhe, Institut AIFB, D-76128 Karlsruhe
Tel.: +49-(0)721-608 4062, Fax.: +49-(0)721-693 717
e-mail: sdecker@aifb.uni-karlsruhe.de
WWW: <http://www.aifb.uni-karlsruhe.de/Staff/sde.html>

■ **Dr. Knut Hinkelmann**

DFKI GmbH, Postfach 2080, D-67608 Kaiserslautern
Tel: +49 631 205-3467; Fax: +49 631 205-3210
e-mail: hinkelma@dfki.uni-kl.de
WWW: <http://www.dfki.uni-kl.de/~hinkelma/>

■ **Dr. Ulrich Reimer**

Rentenanstalt / Swiss Life, Informatik-Forschungsgruppe Postfach, CH-8022 Zuerich
Tel.: +41-1-7114061; Fax: +41-1-7115007
e-mail: reimer@swisslife.ch

Preface

The workshop on “Knowledge-Based Systems for Knowledge Management in Enterprises” took place as part of the 21st Annual German Conference on Artificial Intelligence KI’97, being held on September 9-12, 1997 in Freiburg.

The idea to this workshop arose at a time when more and more enterprises recognized that knowledge has become one of their most important assets. An active and systematically accomplished knowledge management is essential for today’s enterprises’ effectiveness and competitiveness. Since dealing with knowledge has also been a main topic of Artificial Intelligence, the question comes into mind whether AI can make any considerable contributions to the objectives of knowledge management.

Knowledge Management seems to be an issue of enterprise organization and enterprise management beyond any specific technology questions. But there are central and important aspects which can be supported or even enabled by state-of-the-art information systems. The core of successful knowledge management is a comprehensive corporate memory, which influences all kinds of business activities. A corporate memory integrates the dispersed knowledge of an enterprise that relates to problem solving expertise, project experiences, design issues, lessons learned etc. The intention of this workshop was to create a vision for an intelligent corporate information infrastructure with an IT-based corporate memory as its core.

In the following we give a short overview of the workshop contributions. The papers illustrate methodologies, knowledge management tools with more or less strong relations to AI, enabling technologies, and first encouraging knowledge management applications.

Methodologies

Knowledge management can be seen from a number of different perspectives, in particular the social, organizational, and technological perspectives. The paper of Romhardt mainly deals with the human side of knowledge management. It presents ideas on how to preserve the know-how of key knowledge workers even after they left an enterprise. Templeton and Snyder deal with knowledge management from a more technical viewpoint. They describe a methodology for building a database of ‘best practices’ by using an ontological specification process. Their objective is to make knowledge more explicit and transferable.

Knowledge Management Tools

There already are a number of tools that support different basic operations of knowledge management, e.g. identifying, capturing, distributing, utilizing and maintaining knowledge. Frank presents practical experiences with the use of images and spatial representations for building, maintaining and utilizing a common corporate memory. Snyder and Wilson describe a software product that supports the process of incorporating the tacit knowledge of experts in computer-based learning resources. Schmalhofer and Aitken present the EKI system that supports a knowledge manager to analyze a company’s knowledge assets at the knowledge level. Their approach is based on models of human comprehension instead of knowledge-based technologies.

AI Approaches

Knowledge representation and reasoning has been a core task of Artificial Intelligence from its beginning. Pun *et al.* propose a knowledge management framework in the domain of irrigation of grape vines. Information contained in scientific documents is formalised as conceptual graphs such that each document corresponds to one knowledge base. These knowledge bases can be used in multiple ways, e.g. expert systems, decision support, computer-aided instruction. Reimer presents a solution to two kinds of knowledge integration problems: first, integrating various knowledge bases into a (virtually) single one and second, integrating knowledge represented in different degrees of formalization, from formal knowledge representation via semi-structured text to plain text. Daniel *et al.* present a knowledge management framework supporting the technical design process by different kinds of knowledge.

Enabling Technologies

Successful knowledge management is a very complex problem combining a number of different tasks. There are various technologies that can support or enable single tasks. For example, important knowledge is still held in large volumes of books, manuals, and paper archives. Also paper still is a common medium for exchanging information between enterprises and inside an enterprise. Baumann *et al.* demonstrate how document analysis and understanding can be used in order to integrate paper-based information into a corporate memory. Bäsius *et al.* present an alternative approach for extracting information from business letters. Databases and data warehouses are a second wide-spread medium for storing important information. Erdmann shows how data warehouses can be used as a means for storing an important part of a corporate memory that can be represented as formalized data. Efficient execution of processes is a very important aspect of knowledge utilization. Workflow management systems control the execution of business processes. Wargitsch *et al.* present a learning WMS which itself is an organizational memory information system containing a workflow case base and general domain knowledge of an enterprise. Dellen *et al.* present the system CoMo-Kit that supports the execution of unstructured processes by tracing the information flow.

Applications

Applications of knowledge management involve both the utilization of knowledge for improving enterprise-internal collaboration and decision making and for improving the cooperation with suppliers and customers. As described by Wess (there will be no talk by Wess, but the paper is included in this volume), case-based reasoning is a suitable technology for different areas in customer support: help desks, call centers, and self-help applications. Stolze describes important issues involved in creating knowledge structures in an open electronic catalog. Greiner and Rose describe an application of knowledge management in a large organization, the World Health Organization WHO.

Andreas Abecker
Stefan Decker
Knut Hinkelmann
Ulrich Reimer

Preliminary Workshop Schedule

Tuesday, Sept. 9th '97

14.00 - 15.30 h Session: GENERAL ASPECTS

Welcome

A. Abecker, St. Decker, K. Hinkelmann, and U. Reimer

Processes of Knowledge Preservation: Away from a Technology Dominated Approach
K. Romhardt

Toward a Method for Providing Database Structures Derived from an Ontological Specification Process:
the Example of Knowledge Management
G. Templeton and Ch. Snyder

15.30 - 16.00 h Coffee Break

16.00 - 17.30 h Session: TOOLS & TECHNIQUES

Making the Tacit Explicit: The Intangible Assets Monitor in Software
Ch. Snyder and L. Wilson

Multimediale Wissensräume - Werkzeuge für Aufbau, Wartung und Nutzung eines gemeinschaftlichen
Unternehmens-Gedächtnisses
H.-J. Frank

The Acquisition of Novel Knowledge by Creative Re-Organizations
F. Schmalhofer and J. S. Aitken



Wednesday, Sept. 10th '97

14.00 - 15.30 h Session: AI APPROACHES

A Framework for Knowledge Management Systems: A Proposal
D. Pun, C. McDonald, and J. Weckert

Knowledge Integration for Building Organisational Memories
U. Reimer

ERBUS - Towards a Knowledge Management System for Designers
M. Daniel, St. Decker, A. Domanetzki, Ch. Günther, E. Heimbrodt-Habermann, F. Höhn, A. Hoffmann,

H. Röstel, R. Smit, R. Studer, R. Wegner

15.30 - 16.00 h Coffee Break

16.00 - 17.30 h Session: APPLICATIONS

To be announced
N.N.

Smart Product Catalogs at IBM
M. Stolze

Knowledge Management in Global Health Research Planning
C. Greiner and T. Rose



Thursday, Sept. 11th '97

14.00 - 15.30 h Session: ENABLING TECHNOLOGIES I

The Data Warehouse as a Means to Support Knowledge Management
M. Erdmann

From Paper to a Corporate Memory - A First Step
St. Baumann, M. Malburg, H. Meyer auf'm Hofe, and C. Wenzel

Knowledge-Based Interpretation of Business Letters
K.-H. Bläsius, B. Grawemeyer, I. John, and N Kuhn

15.30 - 16.00 h Coffee Break

16.00 - 17.30 h Session: ENABLING TECHNOLOGIES II

WorkBrain: Merging Organizational Memory and Workflow Management Systems
Ch. Wargitsch

Knowledge Management mit CoMo-Kit
B. Dellen, H. Holz, G. Pews

Final Discussion



Contents

<i>St. Baumann, M. Malburg, H. Meyer auf'm Hofe, and C. Wenzel:</i> From Paper to a Corporate Memory - A First Step	1
<i>K.-H. Bläsius, B. Grawemeyer, I. John, and N. Kuhn</i> Knowledge-Based Interpretation of Business Letters	17
<i>M. Daniel, St. Decker, A. Domanetzki, Ch. Günther, E. Heimbrodt-Habermann, F. Höhn, A. Hoffmann, H. Röstel, R. Smit, R. Studer, R. Wegner:</i> ERBUS - Towards a Knowledge Management System for Designers	31
<i>B. Dellen, H. Holz, G. Pews:</i> Knowledge Management mit CoMo-Kit (in German)	41
<i>M. Erdmann:</i> The Data Warehouse as a Means to Support Knowledge Management	53
<i>H.-J. Frank:</i> Multimediale Wissensräume - Werkzeuge für Aufbau, Wartung und Nutzung eines gemeinschaftlichen Unternehmens-Gedächtnisses (outline of a talk - in German)	61
<i>C. Greiner and T. Rose:</i> Knowledge Management in Global Health Research Planning	63
<i>D. Pun, C. McDonald, and J. Weckert:</i> A Framework for Knowledge Management Systems: A Proposal	73
<i>U. Reimer:</i> Knowledge Integration for Building Organisational Memories	85
<i>K. Romhardt:</i> Processes of Knowledge Preservation: Away from a Technology Dominated Approach	97
<i>F. Schmalhofer and J. S. Aitken:</i> The Acquisition of Novel Knowledge by Creative Re-Organizations (abstract of a talk)	109
<i>Ch. Snyder and L. Wilson:</i> Making the Tacit Explicit: The Intangible Assets Monitor in Software	113
<i>M. Stolze:</i> Knowledge Management for Electronic Product Catalogs at IBM (abstract of a talk)	119

<i>G. Templeton and Ch. Snyder:</i>	121
Toward a Method for Providing Database Structures Derived from an Ontological Specification Process: the Example of Knowledge Management	
<i>Ch. Wargitsch:</i>	133
WorkBrain: Merging Organizational Memory and Workflow Management Systems	
<i>St. Wess:</i>	147
Intelligente Systeme für den Customer Support	
(in German - this paper will not be presented at the workshop)	

From Paper to a Corporate Memory

—

A First Step

Stephan Baumann, Michael Malburg, Harald Meyer auf'm Hofe, Claudia Wenzel

German Research Center for Artificial Intelligence DFKI GmbH

P. O. Box 2080, D-67608 Kaiserslautern

Germany

E-mail: {Stephan.Baumann, Michael.Malburg, Harald.Meyer, Claudia.Wenzel}@dfki.de

Abstract

Computer-based corporate memories aim to enable an efficient use of corporate knowledge. Therefore, such systems demand a formal representation of the knowledge itself and of its meta-aspects, i. e., how to use corporate knowledge effectively. Since typical information sources are continuous text and paper-based documents they initially need to be translated into a formal representation.

We argue that techniques of document analysis and understanding can be applied for this transformation. Furthermore, an integration of document and document analysis knowledge into the corporate knowledge leads to a significant improvement of analysis results.

Keywords: Knowledge Management, Knowledge-based Systems, Document Analysis, Document Understanding, Workflow Management

1 Introduction

Computer-based corporate memories aim to enable an efficient use of corporate knowledge. Knowledge-based systems can play a main role in this application scenario [1] because they contribute main ideas to realize *organizational memory information systems* (OMIS). Such systems obviously demand access to application and company specific knowledge concerning two aspects:

- The system uses the available knowledge in a goal-directed way, e. g., when answering queries. Therefore, corporate knowledge is either completely formalized in some knowledge representation language or at least main issues of unstructured knowledge are described formally.
- The system acts intelligently with respect to currently open tasks. A representation of tasks and processes occurring in the company is necessary to achieve such behaviour.

Disadvantageously, it is necessary to spend a lot of effort for the acquisition and the maintainance of such knowledge. Acquisition can be improved by re-use of previously formalized knowledge, e. g., workflow models. In the past, these models have often been specified to deploy *workflow management systems* (WFMS). Additionally, acquisition and use of corporate knowledge can be improved by extracting significant information from informal knowledge sources such as paper documents and electronic texts. This point is of special interest since the paperless office is still far away. Looking at today's situation in even modern enterprises only 20 % of relevant information is electronically available while 80 % is paper-based.

This paper claims that an integration of document analysis and understanding (DAU) techniques with workflow management systems can provide main contributions to realize such information extraction tasks [2]. DAU aims at extracting information from paper sources and translating it into a structured representation. Hence, this integration can be considered as a first step towards a corporate memory.

The analysis process is conducted in five steps [3]: First image analysis algorithms normalize the bitmap of the scanned document by deskewing, noise reduction, etc. Segmentation performs the isolation of textual and non-textual document portions and generates an initial hierarchy of blocks, lines, words, characters and connected components. The attachment of logical labels to these segments is part of the structure analysis. Subsequent text recognition converts the character and word segments to ASCII code. For the processing of electronic documents (e. g., txt, e-mail, HTML) text recognition is superfluous but techniques for the attachment of logical labels may also be applied. At this stage the final information extraction takes place independent of the previously processed media-type (paper or electronic). Domain-dependent approaches are applied in order to classify the entire document and to extract desired information portions. Considering a business letter, e. g., in a purchasing domain, document classification typically distinguishes different document types such as offer, delivery note, invoice, etc. Furthermore the recognition of sender and recipient as well as additional informations about articles, prices and discounts may be extracted.

Classification and information extraction results are appropriate to assign the analysed document to the corresponding open workflow instance of the company's WFMS. Such assignments can be considered as representing the meta aspects of the documents' contents - because recognizing the most relevant open workflow instance determines the way, the document's contents are further processed and stored.

Additionally, this procedure determines the semantics of the previously extracted information and, thus, turns it into a knowledge representation which formal methods can be applied to. For instance, recognizing an extracted digit sequence as an (urgent) invoice date probably leads to immediate money transactions. Contrarily, the digit sequence possibly denotes a date referring to previous letters that are of interest in searching for helpful information relevant to the current workflow instance. This is a completely different interpretation of the digit sequence. Consequently, the result of document analysis can be considered as partial translations into formal representation providing information about title, author, sender, recipient etc. Informal knowledge is projected into formal representations that are useful to implement intelligent archives, to identify relevant information concerning the current task, and to assist finding the next steps to go, hence, to implement an *organizational memory*.

However, providing these services reliably requires availability of knowledge on open processes (context information), expected document types, and several data concerning address data bases etc. from the corporate memory. Hence, we have a bidirectional producer-consumer-relation of document analysis and a corporate memory.

This paper is structured as follows. The first main section introduces into the services of document analysis and their implementation. In a following section, questions arisen in the integration of document analysis and workflow management into an OMIS are answered. Referring to the first section the second one points out that optimal performance of document analysis can only be achieved by exploiting previously archived knowledge coming from the organizational memory.

2 Document Analysis Tasks

In this chapter, we explain how to get from paper documents to an electronic representation of the document's contents.

Document analysis begins as soon as an incoming document is scanned: *Layout extraction* comprises all low-level processing routines like skew angle adjustment and segmentation to compute a document's layout structure, i. e., the detection of boundaries for characters, words, lines, and blocks. Afterwards, *logical labeling* assigns so-called logical labels like sender, recipient, body, signature, etc., to layout regions of the document. *Text recognition* explores the captured text of logical objects. This is done by performing OCR for each character segment and generating word hypotheses afterwards.

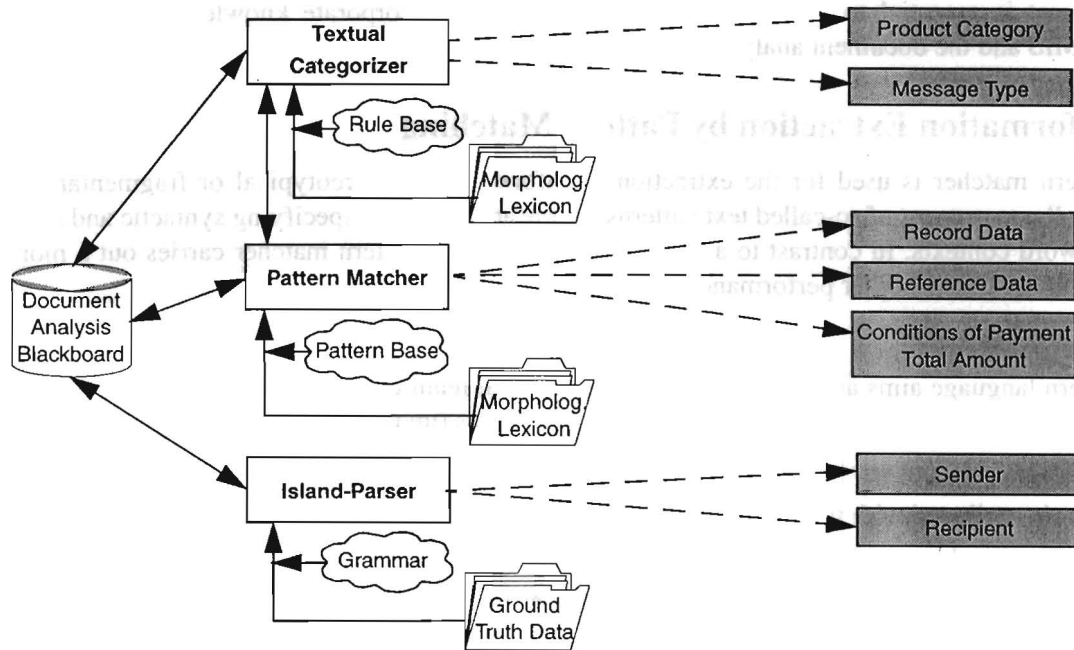


Figure 1: Information Extraction within OfficeMAID

The subsequent components for *document understanding* are shown in Figure 1. Each component gets three kinds of input: The first input source is our blackboard structure where actual analysis results such as OCR results or segmented blocks are stored. The second input source is used by each component exclusively and contains the knowledge specific for the domain and the task in a declarative form (drawn as clouds). The last kind of input is lexical data (drawn as folders). Note that the results of these components – shown at the right hand side – are also stored in the document analysis blackboard. The three document understanding components are used for different tasks as follows.

Firstly, the document is classified according to pre-defined message types. This information is used to start a more in-depth analysis. We apply a rule-based component [4] to determine the appropriate message type. Therefore, it propagates hit scores for text phrases and OCR scores for words and word stems through a network of rules. It employs a pattern matcher [5] for the detection of phrases. In addition, this pattern matcher is used for the extraction of information out of fragmentary sentences. Recognition of sender and recipient is done by applying a standard island-driven parser for unification-based context-free grammars [6]. Up to now, we are not able to extract the whole product information since it is mostly kept in tables requiring a specific treatment.

At this point, the task of document analysis and understanding is completed and the whole extracted information is represented electronically. Therefore the conversion to a desired representation by means of standardized document descriptions (SGML, ODA, ODMA, DMA, etc.) is obviously possible.

In the following, the three components for document understanding are explained in more detail in order to answer the following questions:

- What knowledge is required about incoming documents? All the abovementioned DAU components require knowledge on document types and typical document contents to work correctly.
- Where does this knowledge come from? In the existing system *OfficeMAID* nearly all the knowledge required has been acquired during training phases conducted initially at system installation. Of course, this scenario contradicts the intention of an OMIS to provide services even with a minimum of initially given application specific knowledge. Hence, detailed analysis should clarify in how far *application class* specific knowledge and available data can be used instead.
- What is the internal representation of this knowledge? Document analysis procedures are computation time intensive. Hence, the data they use has to be given in a format enabling an efficient use. This

aspect is essential to understand problems maintaining corporate knowledge distributed over the OMIS and the document analysis system.

2.1 Information Extraction by Pattern Matching

The pattern matcher is used for the extraction of information in stereotypical or fragmentary phrases. It matches all occurrences of so-called text patterns. These are templates specifying syntactic and semantic features of word contexts. In contrast to a parsing component, the pattern matcher carries out a more shallow analysis but reaches a higher performance.

The Pattern Language

Our pattern language aims at combining words with special features into more complex phrases. Such combinations involve nestings of conjunctions, disjunctions, skips (up to n words), and optionality operators.

The two atomic units of our language are

- words attributed with features:
 - The word is mandatory, indicated by the word itself or the word written within quotation marks.
 - The word is optional, indicated by parentheses ().
 - The word specified is a substring of the word to be located in the text, indicated by :sub.
 - The word specified is a morphological word stem, indicated by :morph_stem.
- information to be extracted (= information unit) which is specified by {F= *type_of_extraction*} where the data to be extracted currently are restricted to numbers, prices, proper names of people or cities, telephone numbers, and dates.

These atomic units may be concatenated to patterns according to the following rules:

- If two words follow each other (horizontally) in a text phrase, they are written down consecutively. If they follow each other vertically, this is indicated by the keyword :down. If they may follow either horizontally or vertically, the keyword :both_dir is used.
- A disjunction of words is specified by a vertical bar | between the different alternatives.
- Word skips are defined by giving a maximal word distance <n> between two words.

If a pattern or a group of patterns is given a proper name (indicated by the keyword :sub_pattern followed by this name), this group of patterns can be used as a subpattern within other patterns (indicated by square brackets and the name of the subpattern). This enables the definition of lexical categories and synonyms. In order to meet document analysis requirements, a Levenshtein distance [7] may be defined for each pattern. It controls the degree of tolerance in word matching. To enable a better understanding of the syntax, we now give some English examples out of our domain:

(in) reply | answer to your letter dated of {F= get_date}

confirmation of :both_dir order :both_dir

Thank you (very much) for your order {F= get_number} <2> Mr. | Mrs. | Ms. {F= get_name}

A pattern can be linked to a template-like *concept* to capture the semantic content of a pattern. It is defined by assigning a set of slots to be filled by the pattern matcher. For each pattern, one can specify which slots should be filled by which kind of information.

Pattern Acquisition

When establishing the DAU system within a new domain, *off-line learning* results in the definition of generic patterns to be searched for. Therefore, several conditions have to be fulfilled: New information units which are characterized by a common intra-word structure (e.g., a date) have to be defined as regular expressions manually. Furthermore, training documents have to be provided. Images of these documents are shown to a human trainer. She or he then marks the locations of new pattern instances by dragging a frame and typing in the correct ASCII-text of the pattern.

A generic pattern is generated the following way: At least one common word must appear in each pattern instance. Then, the word positions of all pattern instances are aligned according to the position of the common word. This word is inserted as first part of the generic pattern. Now, the generic pattern is completed by moving a pointer stepwise to the left until the start of the shortest instance is reached. At each position, the algorithm computes inter-word similarities between two words in different instances occurring at similar positions. These similarities are based on the existence of different relationships between two words, e.g., identity, matching morphologic word stem, matching hyperonym, matching regular expression, matching part-of-speech. Inter-word similarities between two words are summed for all possible combinations, i.e., for each instance, all similarities for exactly one word are incorporated. The word combination with the maximal sum is chosen as the most similar word pattern. The corresponding common word features are inserted into the generic pattern on the left-hand side, including word skips. Having reached the start of the shortest instance, the procedure is repeated for a pointer moving from the first common word to the right until the end of the shortest instance is reached. Finally, a description of the generic pattern is obtained.

Pattern Compilation

Before pattern search is invoked the original pattern file is split into different representations. First, it is transformed into a more efficient representation. Second, all of the mandatory words in the patterns are selected, split into trigrams, and compiled into a finite state automaton for characters. This splitting permits a parallel approximate string matching. The compilation is based on the Aho-Corasick algorithm for efficient string search. Thirdly, all words used in the definitions of patterns are picked up and stored in a wordlist for faster access.

Pattern Processing

Pattern processing with the pattern matcher can be divided into several steps as shown in Figure 2. The left-hand side shows at which step the different pattern representations are utilized by the pattern matcher while the right-hand side shows how analysis results are transferred to and from the *OfficeMAID*'s global blackboard.

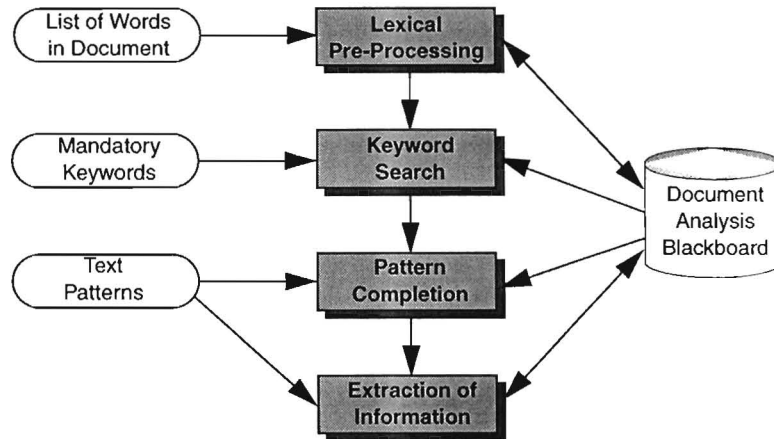


Figure 2: Pattern Processing

Because all of the words occurring in the text patterns were extracted during the compilation of patterns, lexical pre-processing [8] is now able to verify the results of OCR with respect to a list of valid words. Thus, dictionary look-up returns valid word hypotheses based on the weighted edit distance including the consideration of splits or merges of word segments. Moreover, lexical pre-processing checks OCR results for the appearance of distinct types of information units.

Now the pattern matching process begins. All mandatory words appearing in patterns and present in the actual document are located in one pass. First, all word hypotheses in the document are split into trigrams. Then, the finite state automaton which contains all possible trigram sequences of mandatory words, successively checks the initial letters of all trigram hypotheses of the actual document. If a character matches, the next one is tested and so on until either a terminal state is reached, i. e. a trigram is matched, or the relative Levenshtein distance is exceeded. After all trigrams are located, the system tries to match the entire word

within a maximum Levenshtein distance constraint.

Then, all patterns are tested successively for completion: Patterns containing subpatterns are expanded and split into several patterns. All mandatory words and substrings in a pattern are checked from left to right. This takes into account horizontal and vertical directions, disjunctions of words, and possible word skips. If all of these words are matched at an appropriate position, the pattern is checked further to locate optional keywords and to match information units. Finally, the system generates an instantiated set of detected patterns. Redundant patterns are removed and pattern probabilities are calculated bottom-up by multiplying word probabilities.

Finally, the information units of an instantiated pattern are used to instantiate the appropriate concept, i. e. , they are transformed into values for the different concept slots. Note that the slots need not conform to exactly one information unit, but the information they capture may as well be implicitly given by the pattern. Finally, instantiated concepts are stored in the blackboard structure and build up a part of the so-called message of the letter.

2.2 Rule Interpreter

This component takes text features of a document as input and propagates their certainty factors through a rule network to determine the correct class.

Rule Acquisition

A classifier rule consists of either a text pattern or a single word on the left-hand side, and a class name on the right-hand side. Moreover, a certainty factor is attached to the rule. Two different procedures are invoked for rule acquisition:

- For the generation of rules where single words are important text features, acquisition can be done automatically. A training set is required which contains the assignment of correct classes to training documents. First, conditional probabilities for all the words in the training set are computed with respect to the distinct classes. Afterwards, a threshold is applied: Whenever a word occurs significantly often in a certain class, a rule is created for this word. The probability of the rule is calculated on the basis of the conditional probability.
- The abovementioned off-line learning for patterns must take place before the generation of pattern rules. Then, patterns are treated as text features in the same way as described above to enable their integration into a rule.

Classification

Our text classifier is based on the rule interpreter FuzzyClips [9] for representing and manipulating fuzzy or uncertain facts and rules. FuzzyClips is freely available for educational and research purposes.

At the beginning of the classification process, the rule interpreter is provided the necessary information: Words, together with recognition scores, are entered as facts, the rulebase is loaded and checked for identifiers of text patterns which appear in rule conditions. These identifiers are collected and the pattern matcher is invoked to locate appearances of these text patterns in the original text. Afterwards, pattern instances, along with their recognition scores, are entered as facts in the factbase. At this point, all rules and facts have been inserted and the rule interpreter is started. Now, the recognition scores of single words and patterns are propagated through the rule network to provide and refine an estimate of the certainty of the different classes. The system computes the certainty of a rule action based on the certainty of the rule condition and the rule certainty itself. Moreover, the certainty of a fact there are several evidences for must be computed. In both cases, several evidences for each hypothesis are combined. Therefore, we use an approach similar to the one of the expert system MYCIN. At the end of the inference process, the fact base contains new facts which consists of classes in conjunction with certainty factors expressing how well the respective class corresponds to the actual document.

2.3 Information Extraction by Parsing

Like the pattern matcher is used especially for information extraction of stereotyped phrases, the analysis of highly structured parts (i. e., parts showing regular linguistic structures in some sense) can be performed by a parsing component. For the application domain of printed business letters, we exemplarily use this technique for identification of correspondents, i. e., sender and recipient of a message. Therefore, we have built a component upon a standard parsing system for feature grammars which starts with the text block expected to hold the relevant information, i. e. the address, and in performing the following steps: lexical pre-processing, address parsing, and data base matching. Before we describe these analysis steps, we say a few words on preparatory steps, i. e., creating the lexicon, grammar, and address data base. Such preparations show some special relation to corporate knowledge.

2.3.1 Preparatory Steps

In document analysis, recording of the so-called Ground Truth (GT) data is performed in order to enable the evaluation and benchmarking of the system. This is comparable to the related procedure in speech recognition, where acoustic speech data is transcribed into electronic text. This text transcription corresponds to GT for OCR in document analysis.

For information extraction, we additionally record GT data at the conceptual level. In particular, this GT comprises – beneath the ASCII text – the logical arrangement of a letter's parts, such as recipient, sender, body and relational data. Furthermore, message level information is given to describe the content, e. g., *"this is an order sent by company x for the items y to sell"*.

Starting from this GT data which comprises both message-level and person-specific (address) data, a lexicon and an address data base are compiled. In order to reduce manual effort, GT data is recorded at some aggregational level between words and phrases. For addresses, this means actually that the slots to be filled are organization's name and type, person's name, etc. Because GT is not always given at word level and not part-of-speech tagged, several heuristics have to be used within the lexicon generation step. A typical example are such cases where a company's name consists of its owner's name plus the company type, e. g., "Peter Stuyvesant & Sons, Ltd.". Here, the occurrence of the keywords "Sons" and "Ltd." is sufficient for the hypothesis that "Peter" is the forename and "Stuyvesant" is the surname of the company's owner. Anyway, such heuristics may fail since there are several possible deep structures for a single form; e. g., compare "Peter Stuyvesant Corporation" and "General Electric Company".

In addition to this semi-automatically generated lexicon, a small hand-crafted lexicon of function words is drawn up manually during the construction of the grammar. In detail, each word having a special role within the address grammar is attached as an entry in this lexicon. Such function words are the German equivalents to "P. O. Box", "Mister", "Miss", etc.

Within the given context of an OMIS, perviously analyzed documents and respectively recorded results are used in the same way which GT data is used for system preparation.

2.3.2 Lexical Pre-Processing

The first step in the analysis of a given text block is the recognition of its text. After OCR, the dictionary generated from the GT data is used with the above-mentioned word recognition component. This leads to a word hypotheses lattice comparable to those used in speech processing.

Since not all words are known in advance, a heuristic word description is assigned additionally. The respective default categories depend on the character contents of the recognized words, e. g., 5-digit numbers are assumed to be ZIP codes, words with upper-case initial letter are assumed to be names, and so on. Sometimes, more interesting word categories can be assumed, e. g., street names typically ending in "street", "avenue", and town names having typical endings like the English "town" or "ton".

2.3.3 Address Parsing

Subsequently, the parser is invoked to compute valid parses. The parsing itself is standard and therefore

needs no detailed description. It works on a feature-based grammar formalism related to PATR-II and consists of an island-driven chart parser, i. e., a parser driven by recognition probabilities. We have enhanced this standard parsing technique by adding the possibility of partial parsing which means that input to the parser may be fragmentary. This is important for document analysis where OCR errors or deficiencies in the lexical data base occur frequently.

Partial parsing is performed in the following way. Since the components of an address are easy to delimit, a kind of case frame is defined consisting of the parts town, street/P. O. Box, name, department, and institution. This frame is mapped to the resulting feature structure of a complete parsing result and thus a uniform post-processing can be done for partial and complete parses. For partial parses, strict rules are given on how to fit a partial parse into this frame. Thus, partial parses can be combined to more complete ones by applying these rules.

2.3.4 Data Base Matching

Data base matching is carried out as follows. For each pair of parse and a particular data base entry in question, a formula is generated; this formula computes the total matching weight of the parse/data-base pair mainly by summing the matching weights of the parse's components. Some primitive matching weights are defined for names and numbers; the more complex weights are aggregated from a few primitives.

The principle of these formulas will be described by an example. A partial parse yields a case frame with department, name, and city; in particular, the corresponding organization and the street or P. O. Box address are unknown. For each data base entry where – in this case – the person's name matches the GT data, the respective formula is determined. The formula's denominator is determined only by the parse. It ensures that the resulting weight is not influenced by the amount of GT data available for one entry (which is, in general, a more accidental property of a GT entry's content). Thus, the best possible weight occurs for those data base entries containing the same frame slots as the parse. In our example, this happens for entries with department, name, and city given. A loss of weight occurs for those cases where one of those slots is empty, e. g., if the department is not given.

This way, a list of weighted pairs of parses and data base entries is computed. Actually, the total weight of such a pair is calculated by simple multiplication of the parse weight and the match weight. In a final step, the most probable parse-to-data-base match is determined by looking at some of the best weighted matches, taking into account their differences in content (e. g., persons with the same name in one company but with different departments), and the difference of their weights.

For the final result, we also consider the relative differences between the best parsing results. Assume that a well weighted parse is computed but no data base match is found for it. Instead of returning some possible match for the second best parse, it may be better to reject this parse, i. e., to say "*no result found*". In such cases, the lexical heuristics described in Section 2.3.2 may have major influence on the resulting weights. Finally, the component outputs exactly one result or a reject.

3 Integrating in an Organizational Memory.

As mentioned in the introduction, the relation of DAU and an OMIS is twofold. On the one hand, DAU helps to keep formalized knowledge in organizational memories up to date providing the services described in Section 2. On the other hand, DAU techniques require knowledge from the organizational memory in order to work reliably. Consequently, the DAU system has to be linked to the embedding OMIS in both directions: Information retrieved from incoming documents has to be related to the data and knowledge bases in which it has to be stored. Furthermore, the system needs to know how to retrieve useful information from the OMIS. Unfortunately, the different kinds of knowledge will suitably be represented in different formalisms. For instance, generic document types and knowledge on how to perform DAU tasks are parts of the DAU system whereas corporate knowledge typically is given in terms specific to that corporation. Hence, integrat-

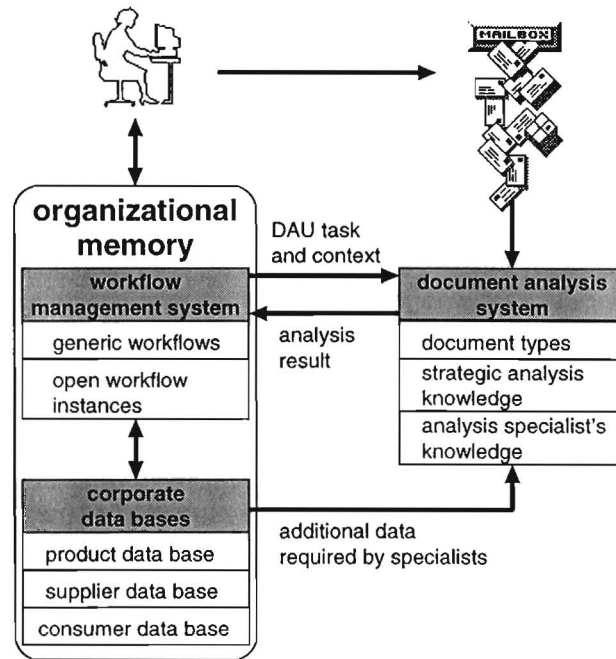


Figure 3: DAU within an organizational memory

ing DAU systems into a corporate memory is a non-trivial task (cf. Figure 3) and is very similar to the problem of knowledge base integration in general [10].

The workflow model plays a key role in this scenario. On the one hand, the main services of DAU systems assign analysed documents to open processes in the corresponding workflow. For instance, after finishing an "order some goods" task a workflow management system will usually generate a "handle invoice" workflow instance referring to exactly this order. If DAU is able to assign the incoming invoice to this open workflow instance, all further processing can be left to the workflow management system. Hence, available workflow models can help building a bridge from DAU to the organizational memory. On the other hand, open workflow instances concerning DAU tasks provide valuable task and context information for the improvement of DAU reliability. *Task information* prevents the system from performing unnecessary analysis steps. If e. g. a classification of documents is required there is no need to extract further information from the document. Another property of tasks states whether to optimize accuracy or response time of DAU steps. *Context information* addresses knowledge about what is expected rather than what is to do. Returning to the example mentioned above, the DAU system needs to know whether an open workflow instance is waiting for an invoice or not. If the workflow management system does not expect any invoices but the DAU system thinks to have found one there must be something wrong.

The following sections cover these problems and chances of tight interaction between DAU and organizational memory.

3.1 Different knowledge types

Having a more detailed view on DAU in the context of a corporate memory, three kinds of knowledge play an important role. These knowledge portions differ in content, terminology and life time. However, providing reliable document analysis services requires extensive use of all these knowledge sources.

Corporate Knowledge comprises the contents of the OMIS, workflow model, databases, organizational diagrams and ER-diagrams etc. This knowledge is represented in terms specific to the corporation. Thus, knowledge on corporation specific conventions is necessary to understand the meaning of proper names and identifiers occurring in the representation of corporate knowledge.

Document Knowledge is composed of structural knowledge about documents, i. e., layout objects, logical

labels, relevant message types etc. Document knowledge is represented due to the more generic terms of the document analysis domains like for instance business letters. Representations of document knowledge are therefore understandable beyond the context of a specific corporation. Proper names and identifiers are commonly used in the application domain, e. g., business letters.

Figure 4 sketches our idea on how to represent generic knowledge on documents (within the dashed box). Basically, the representation is built of an aggregational and taxonomical hierarchy enriched by inherited attribute/value pairs and constraints between attribute values. Attributes are listed in the box representing a term followed by constraints on attribute values. Additionally, an aggregation represents part/whole relations and is denoted by thin lines labeled with an identifier of the relation and a range of cardinal numbers stating how many components can be part of a whole due to the named relation.

As an example consider class TIFF_Document. Relevant attributes of global interest concern the questions: Are colours represented (attribute colours)? Has the pixel graphic been normalized by standard procedures (attribute preprocessed)? Instances of another concept Word_Hypotheses represent word hypotheses. Lexicons that have been used to prove hypotheses can be parts of these instances. However, the most important concepts concern document types. Attributes of subsumptions of the concept Document_Type typically are possible goals of information extraction tasks. Parts of a document type refer to additional data useful for accomplishing document analysis tasks like pattern descriptions and grammars. In fact, such data is compatible with the intention of a taxonomy because patterns and grammars can be considered as constraints on texts. Hence, demanding consistency with a pattern or a grammar can be viewed as concept specialization.

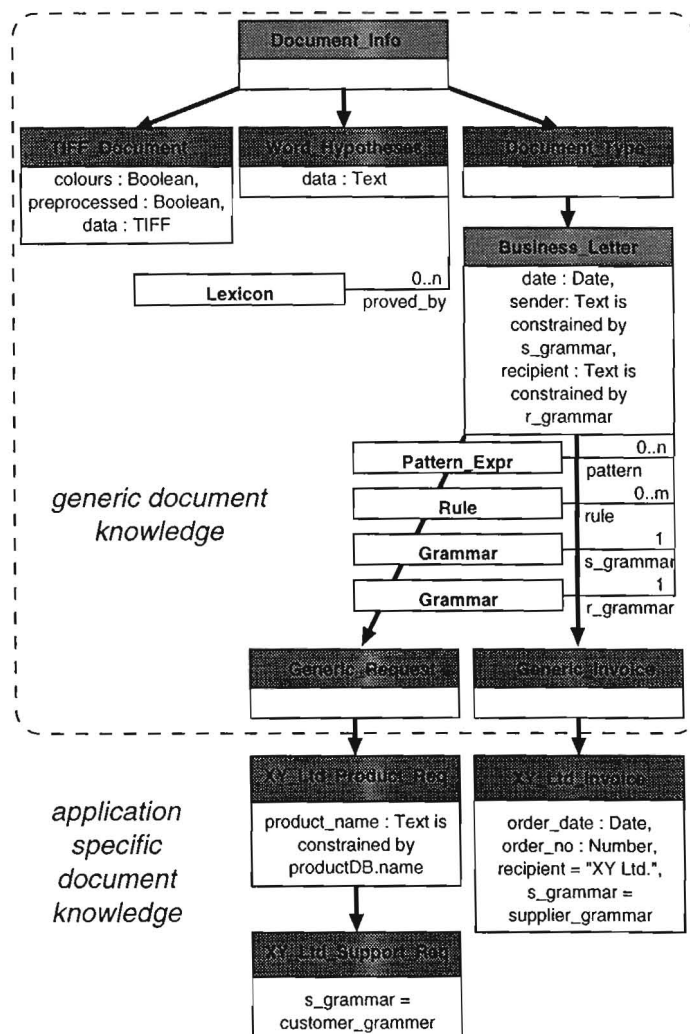


Figure 4: Concepts of document knowledge - a small cut

Viewing documents as instances of document concepts enables application specific specializations of generic document types as gateways to available corporate knowledge. An example is given in Figure 4. Three document types specific to the XY Ltd. company have been added to the knowledge base inheriting most of their properties from generic document types. Either attributes are added or inherited attributes are constrained to more concrete expectations. Document type XY_Ltd_Product_Req for instance adds an additional product_name attribute that is constrained by the available products. An example for providing more detailed information on documents by a more specific concept is XY_Ltd_Support_Req. Here, senders of such a request are assumed to have bought a product and, hence, have an entry in the customer database of XY Ltd. Consequently, the sender attribute is constrained to a grammar retrieved by the customer database. Such information is of obvious interest for enhanced document classification and information extraction operations.

The sketched formalism can be considered as an ontology for sharing knowledge on documents [10] appropriate for the following two purposes: On the one hand the ontology is a specification of data structures for storage and communication of document analysis inputs and results. On the other hand these concepts are required for specifying document analysis tasks and characterizing program modules accomplishing document analysis tasks describing their input requirements and their deliverables. This use of ontologies is very similar to their use for example in system for knowledge-based design[11].

Strategic Knowledge refers to the latter point. As pointed out in Section 2, document analysis typically is accomplished by an application specific configuration of program modules. In contrast to state of the art applications, document analysis within a corporate memory can be expected to concern wide application domains (like the whole correspondence of a company by snail mail and e-mail) with many document types. As mentioned before, access to a large variety of useful corporate knowledge is required. The opportunity to flexibly choose document analysis modules and their parameters with reference to the current application context is desired. An abstract notation can be used to support linking generic knowledge on the performance of document analysis modules to the corporate memory and, as we think, to automatically configure document analysis processes according to dynamically changing requirements.

Our ideas on how to specify document analysis processes are sketched in Figure 5. Again, a concept language is employed with the same elements as in the document knowledge. Concept DAU_Op is the most general concept of document analysis program modules resp. tasks. Some attributes of global interest like a qualitative characterization of the precision or the recall can be used to represent either properties of or requirements on document analysis program modules. Parts of that concept represent either provided resp. required input (part input) or provided resp. desired output (part output).

Subconcept DAU_Plan describes possible compositions of document analysis operations. Instances of DAU_Interface are used to represent something like a consistent resource-oriented configuration [12][13] where all inputs required by analysis programs (the parts of op) are either inputs of the whole plan or provided by another program.

Subconcepts of Specialist_Op describe the available program modules. Hence, a reference to the data base of available programs is maintained. Optionally, additional constraints on the input parts can be used to represent applicability of a program module, whereas constraints on the output parts represent deliverables. Note, that the program modules gain access to relevant data through the properties of their input and output parts. E.g. the rule classifier can retrieve relevant rules and patterns from the concept description of its output (crf. Figure 4), the island parser reads the relevant grammar from its output concept representation. Hence, representing concepts of document analysis operations enables application specific tuning with corporate knowledge without detailed knowledge on document analysis techniques. Application specific extensions of this ontology can be used to specify application specific document analysis tasks. Subconcepts of Specialist_Op introduce application specific analysis modules resp. modified usage of the original modules with e.g. specific parameters. Additionally, subconcepts of DAU_Op may be specified in order to create new application specific analysis tasks. As an example consider concept Analyse_Invoice_Op. This new task assumes a TIFF_Document as input and is supposed to classify a XY_Ltd_Invoice (as Defined in Figure 4) as output and extract the attribute order_no. The *VirtualOffice* project aims at developing methods founded on the configuration of technical systems [12][14][15] for finding realizations of such application specific tasks

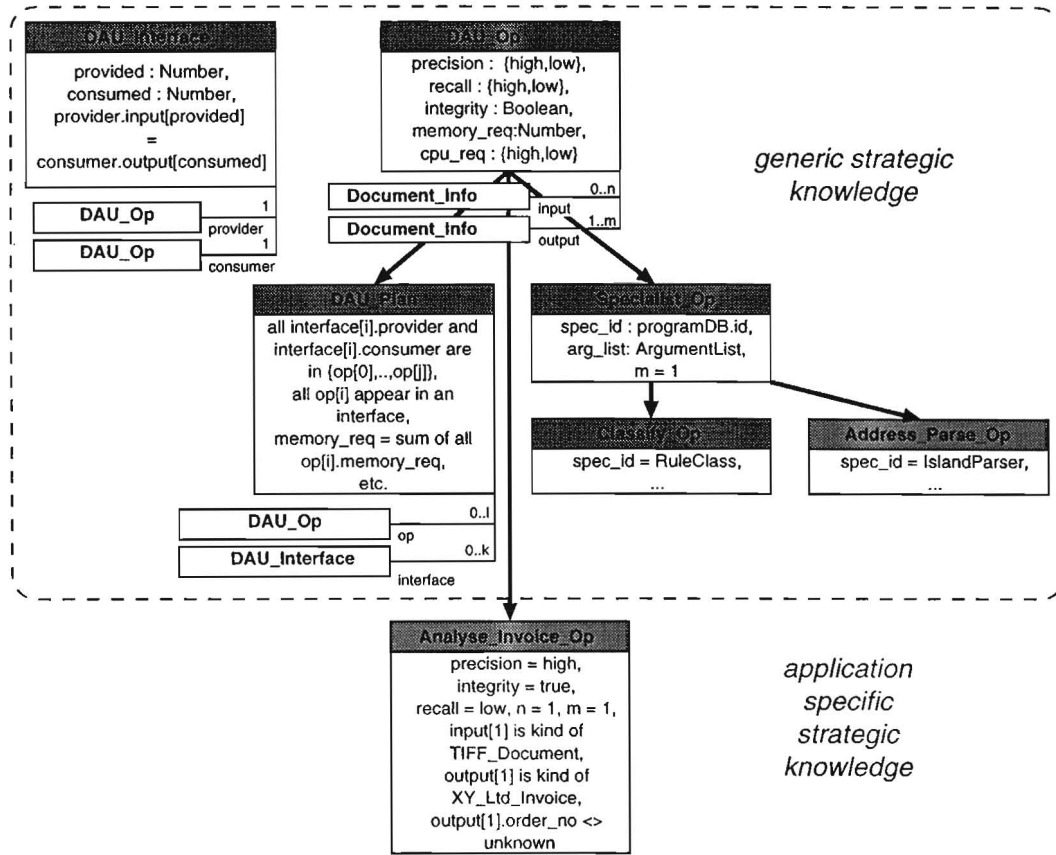


Figure 5: Concepts of strategic knowledge - a small cut

either automatically or in close dialog to an expert. More detailed: Building a configuration of an abstract concept like Analyse_Input_Op means finding an instance of the composed concept Analyse_Input_Op and DAU_Plan.

An intelligent DAU system (with the ability to configure analysis processes automatically due to abstract task concepts) enables use of task and context information as described above. Exploiting task information is relatively easy. For instance in concept Analyse_Invoice_Op the only required information extraction task is to find out the number of the order preceeding the current invoice. Neither the sender nor the date attribute have to be extracted in order to complete the task Analyse_Invoice_Op. Hence, task specification is appropriate to avoid unnecessary computations and optimize performance of the DAU system.

Moreover, context information aims at improving reliability of DAU processes by providing assumptions on incoming documents. Consider an order that has been submitted to a supplier. Consequently, beneath always expected product requests the OMIS is waiting for an invoice referring to the submitted order. Hence, the DAU is told to conduct an application specific procedure where the output is constrained to the document types XY_Ltd_Product_Req and XY_Ltd_Invoice. Automatic configuration now chooses appropriate patterns and rules to be processed and uses expected data for internal consistency checks. After having received the invoice, the corresponding document type is removed from the goal concept. The configuration of the DAU system changes again because there is no open order left.

Besides the usual problems of knowledge-based configuration systems [16] automatic and dynamic configuration of DAU tasks requires to cope with unexpected behaviour of the employed DAU algorithms. Fig-

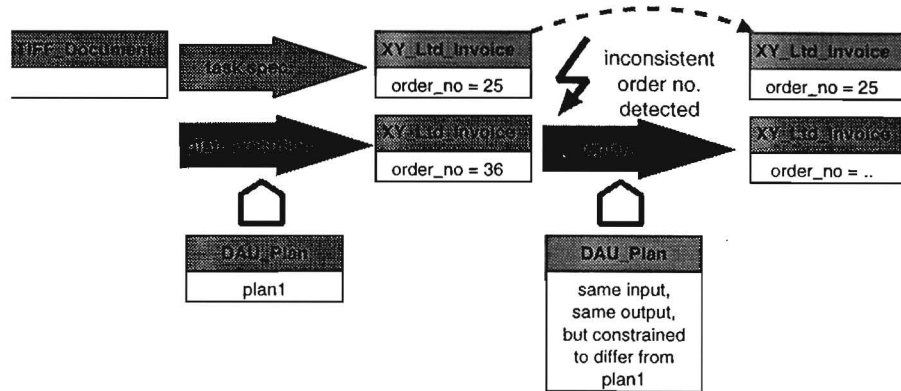


Figure 6: Replanning because of an unsatisfied integrity constraint

Figure 6 gives an example. Plan1 has been expected to provide a XY_Ltd_invoice result with order_no constrained to 25. Unfortunately, the result of plan1 violates this constraint perhaps because of an OCR failure. The system recognizes this inconsistency and starts a replanning operation with a new constraint on the new plan. It is required to differ from plan1.

3.2 Interaction between DAU and the process model

The different knowledge types and their relevance for conducting DAU in a corporate memory have been discussed in chapter 3.1. Furthermore, we have sketched their representation. For the purpose of context-oriented instantiation and specialization of these knowledge types in an application domain (e.g. purchasing) we want to make use of given workflows specifying the daily work processes. When talking about workflows one has to make a clear distinction what our interest in this paper is.

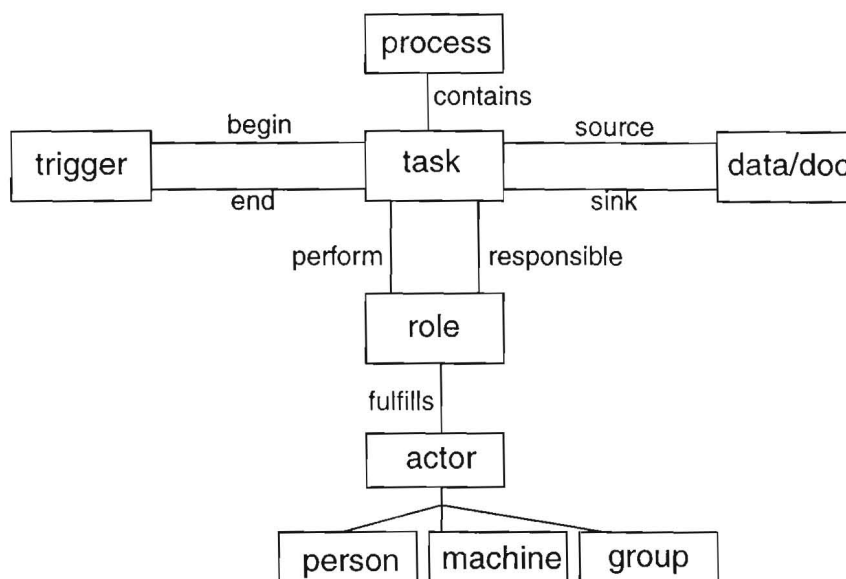
There exist a large variety of different workflow models, representation techniques and their implementation through WFMS in research and industrial use. In this paper, we are interested in the common concepts of workflow models with respect to their applicability as valuable knowledge source for document analysis and understanding. Therefore we will start in this section with a short overview on common modeling elements and show how to use them for gaining application-specific document knowledge as well as providing input for the handling of DAU tasks and contexts. The latter will be explained for both, the specification of a generic workflow with integrated DAU tasks at buildtime and the benefits of using the context of workflow instances at runtime.

Workflow models may be classified regarding their *design centers* – the main concepts used for implementing a workflow. Most commercial systems are obviously centered around the process, but also agent and data-centered systems are known [17]. Due to this fact modeling elements for the specification of workflows slightly differ. In the following we want to concentrate on common elements. First of all a workflow model should contain the following perspectives [18] on a process-centric view of work:

- **Organization:** this perspective describes the organizational structure of a company and the different individuals (actors) being involved. Links between specific organizational units and tasks specify responsibility and performing actors (role concept). In case of automated tasks the required applications are triggered by the workflow engine without user interaction, resp. the role indicates an organizational unit representing a system (automated actions, agents).
- **Data:** this perspective on data, resp. documents describes the work objects which are required to perform tasks, resp. generate or modify data. Therefore the notion of input/output container or source and sink relations may be used.
- **Tasks:** tasks represent a collection of actions or one elementary action which describe the work content. The tools or applications required for working and therefore creating or modifying objects have to be specified. Trigger conditions are necessary to define the beginning and completion of a task.

Here logical expressions on contents of the sink and source container are used to determine these trigger conditions or events.

- **Process:** Using the abovementioned modeling concepts allows a specification of the entire process. The individual tasks are nested to each other by triggers and the source/sink relations. The role concept specifies the embedding into the company's organizational structure.



These concepts are supported by standard WFMS and used to specify the work process of interest. In our scenario of an integrated DAU and corporate memory system document knowledge and strategic knowledge can be located in the following way:

Document knowledge

- **Data / documents:** if data is represented in input or output documents it may contain additional document knowledge (e.g. appearance of a form of supplier XY and relevant data field descriptions). Furthermore the description of application-specific document types (e.g. an invoice consists of: products delivered, extension, reference to order.no) represents document knowledge. On the other hand links to data descriptions as contained in corporate databases (e.g. product specifications can be accessed via the oracle-product-DB) represent corporate knowledge.
- **Role / Actor:** the human actors being specified in the organizational structure (this part has to be included in almost every WFMS) appear often as recipient / sender in incoming and outgoing documents and represent at a first look corporate knowledge. But also document and specialist knowledge can be derived from this source (e. g., the recipient_grammar)

DAU tasks

- **Tasks:** if a task consists of actions for information extraction out of documents or generating or modifying documents it may contain DAU-relevant tasks, i.e. strategic knowledge (e.g. extract the extension of an invoice, generate an inquiry via MS-WORD containing demanded products and the running inquiry number).

The concepts of strategic knowledge support different kind of granularity for representing DAU tasks in the workflow description. Workflow designers without knowledge about DAU design their workflow in a task-driven manner. Actions concerning information extraction or classification tasks are specified as automated actions. The performing actor is the entire DAU system (which performs automated configuration on demand using a standard DAU_Plan). By denoting a specific Specialist_Op (see Figure 5) as performing actors the experienced workflow designer has the possibility to specify concrete DAU techniques.

DAU context

As we mentioned earlier an efficient way to support DAU is the usage of workflow context at runtime. Instances of the generic workflow will be handled by a workflow engine which implements the underlying execution semantics. If the trigger condition for a DAU action fires the workflow engine makes a switch of control to the DAU system and all available context stemming from the set of running workflow instances are given to the consuming DAU operation. This context may include the following informations, resp. expectations (actually the contents of the internal workflow database):

- actors having performed previous working steps of this workflow instance
- acquired data
- expected document types
- expected sender, recipient
- references (date, documents)

As a final step the extracted results are passed to the data fields (optionally after human expert verification) of the corporate database or document container as specified in the workflow. In this way DAU can be embedded as an information supplier into a WFMS, resp. corporate memory.

3.3 Changes in Knowledge Sources

An organizational memory can be expected to change dynamically during use [1]. It permanently incorporates new knowledge and maybe forgets old and now irrelevant knowledge chunks. However, such behaviour although desired causes problems for all software components like the DAU system that exploit the mutating knowledge represented in the OMIS. These problems concern the consistency between the organizational memory and the results of the component. Possibly, some relevant knowledge changed while computing the latest results. There are principally two extreme opportunities to cope with this situation:

1. Forget all intermediate results of the current computations after possibly relevant changes in the organizational memory.
2. Ignore changes in the organizational memory and accept a somehow fuzzy validity of results.

In the *VirtualOffice* system we will try to find some ways to represent opportunities in between these extremes as a property of DAU tasks.

4 Conclusion

Summing up, this paper presented fundamental arguments to employ document analysis methods for keeping an organizational memory up to date with newly received unstructured documents. This is necessary to answer questions concerning for instance a company's correspondence by snail mail and e-mail. However, the corresponding analysis tasks require much more flexibility than state of the art document analysis applications and the ability to exploit corporate knowledge by the DAU system. Our suggestion to cope with these requirements is to apply methods coming from knowledge-based systems.

In this paper we sketched an ontology of documents, document analysis tasks, and operations in order to bridge the terminological gap between more generic knowledge on typical documents and the application specific representation of organizational memories. This method enables document analysis modules to access corporate knowledge if needed avoiding spending too much effort on integrating the DAU system into an existing OMIS.

Applicability of the DAU system is increased by a modular software architecture that is configured due to the dynamically changing requirements and expectations of the OMIS. Information on the current context of document analysis can be retrieved from a workflow model. Abstract specification of DAU tasks enables the application of a knowledge-based configuration system in order to determine the configuration of the system that complies best with the current context.

In fact, both problems are probably not specific to the application of document analysis systems. Kühn and Abecker proposed a modular architecture for generic OMISs [1]. All these modules require and deliver knowledge chunks. They have to communicate and must be kept consistent. Hence, this paper may be viewed as a case study on the integration of such modules using ontologies enhanced by application specific concepts for knowledge interchange between these modules and knowledge-based configuration system to implement context sensitive behaviour of the modules.

Acknowledgements

This work has been supported by the German federal ministry of education, science, research and technology (BMBF) under contract FKZ-ITW-9702 (Project *VirtualOffice*).

References

- [1] Otto Kühn and Andreas Abecker. Corporate memories for knowledge management in industrial practice: Prospects and challenges. *Journal of Universal Computer Science*, 1997. submitted.
- [2] S. Baumann, M.H. Malburg, and C. Wenzel. May document analysis tools bridge the gap between paper and workflows? In *Proceedings IFCIS: International Conference on Cooperative Information Systems (CoopIS-96)*, Brussels, Belgium, pages 135-142, June 1996.
- [3] A. Dengel, R. Bleisinger, F. Fein, R. Hoch, F. Hönes, M. Malburg. OfficeMAID — A System for Office Mail Analysis, Interpretation and Delivery. In: L. Spitz and A. Dengel (eds.), *Document Analysis Systems*, World Scientific Publishing Co. Inc., Singapore (1995), pp. 52-75.
- [4] C. Wenzel, R. Hoch. Text Categorization of Scanned Documents Applying a Rule-based Approach. *Fourth Annual Symposium on Document Analysis and Information Retrieval*, Las Vegas, Nevada, April 24-26, 1995, pp. 333-346.
- [5] C. Wenzel. Supporting Information Extraction from Printed Documents by Lexico-Semantic Pattern Matching. *Fourth International Conference on Document Analysis and Recognition*, Ulm, Germany, August 18-20, 1997. To appear.
- [6] M. H. Malburg. Address Recognition with Robust NLU Technology. *Florida Artificial Intelligence Research Symposium*, Special Track on Real-World Natural Language Understanding, Key West, Florida, Mai 1996.
- [7] V. I. Levenshtein. *Binary Codes Capable of Correcting Deletions, Insertions, and Reversals*. Soviet Phys. Dokl., vol.10, no.8, 1966, pp. 707-710.
- [8] A. Weigel, S. Baumann, J. Rohrschneider. Lexical Postprocessing by Heuristic Search and Automatic Determination of the Edit Costs. *Third International Conference on Document Analysis and Recognition*, Montreal, Canada, August 1995, pp. 857-860.
- [9] Knowledge Systems Laboratory, Institute for Information Technology, National Research Council Canada. *FuzzyCLIPS Version 6.02A User's Guide*. September 1994.
- [10] Thomas R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. In Nicola Guarino and Roberto Poli, editors, *Formal Ontology in Conceptual Analysis and Knowledge Representation*. Kluwer Academic Publishers, 1993.
- [11] Thomas R. Gruber, Jay M. Tenenbaum, and Jay C. Weber. Towards a knowledge medium for collaborative product development. In John S. Gero, editor, *Proceedings on the Second International Conference on Artificial Intelligence in Design*, pages 413--432. Kluwer Academic Publishers, 1992.
- [12] M. Heinrich and E.-W. Jüngst. A resource-based paradigm for the configuring of technical systems from modular components. In *CAIA-91: Proc. of the 7th IEEE Conference on AI Applications*, 1991.
- [13] O. Najmann and B. Stein. A theoretical framework for configuration. In *Proceedings of the 5th IEAAIE*, 1992.
- [14] A. Günter. *Wissensbasiertes Konfigurieren - Ergebnisse aus dem Projekt PROKON*. Infix Verlag, Sankt Augustin, 1995.
- [15] F. Baader, H.-J. Burckert, A. Günter, and W. Nutt, editors. *WRKP-96: Proceedings of the Workshop on Knowledge Representation and Configuration*, number D-96-04 in DFKI Document, August 1996.
- [16] Harald Meyer auf'm Hofe. What is still to do in order to solve configuration problems in practice? In [15], pages 25-32, 1996.
- [17] K. Schwab. *Conception, Development and Implementation of a Computer Based Office System for the Modeling of Activity Classes and Execution of and Control of Activities*, Diss., University of Bamberg, Germany 1993.
- [18] S. Jablonski, C. Bussler. *Workflow Management - Modeling Concepts, Architecture and Implementation*. International Thomson Computer Press, 1996.

Knowledge-based Interpretation of Business Letters¹

Karl-Hans Bläsius, Beate Grawemeyer,

Isabel John, Norbert Kuhn

Fachbereich Angewandte Informatik

Fachhochschule Trier

Postfach 1826

D-54208 Trier

E-Mail: {blaesius | grawemeyer | john | kuhn}@informatik.fh-trier.de

Abstract

The performance of document analysis systems significantly depends on knowledge about the application domain that can be exploited in the analysis process. Typically, one has to deal with different sources of knowledge like syntactic knowledge, semantic knowledge or strategic knowledge guiding the analysis process.

We present a knowledge based document analysis system based on a knowledge representation language specially designed for document analysis tasks. It allows to model and to interpret structural knowledge about documents and knowledge about the analysis process declaratively in a common framework.

Keywords: Document Analysis, Document Knowledge

1. Introduction

A major problem in office environments is to deal with the huge amount of information that has to be accessible from each office workspace. The exchange of information mainly takes place

¹ This work has been supported by the Stiftung Rheinland-Pfalz für Innovation under grant 8031-38 62 61/228

through electronic or paper documents. To cope with this huge amount of information automatic selection and interpretation processes for these documents are needed. Document analysis systems can help to find the information that is required in a concrete situation. So, document analysis systems or applications become more and more important for integrated office automation systems.

One task that can be tackled by the use of document analysis systems is to group together documents into office procedures. The user can access documents of a concrete procedure by searching such documents which fulfil a set of predicates. These predicates can either refer to content portions of the documents, e.g. those containing the string "Fachhochschule", or to attributes attached to a document, e.g. those containing the string "Fachhochschule" in the address block. The difference between these two search patterns is that the latter one requires for a logical analysis of the documents under consideration. Accessing documents in office procedures often requires to have documents tagged with a certain type, e.g. tags like order or invoice. Document analysis systems can be used to produce this information almost automatically, either for incoming or for outgoing documents of an office. Thus, a larger amount of information can be attached to documents providing a more detailed structure in the storage of documents. By that, document analysis systems are a valuable tool for setting up an enterprise information system.

However, the question arises why document analysis systems are not widely used in today's offices. One of the reasons for that may be that analysis tasks are rather domain specific and therefore, one cannot expect a general purpose system. Consequently, most existing document analysis systems are restricted to relative small domains. To adapt an existing system to a new domain is often as time consuming as developing a new system from scratch.

To our belief, this effort can be significantly reduced when certain design guidelines are considered. For us, this has led to a knowledge based formalism to describe documents. It should allow to separate general analysis knowledge from domain specific one. While the former knowledge can be reused in different applications only the latter one has to be reconstructed when moving from one application to another one.

For a new application one has to encode knowledge about the new domain. Usually this includes to describe the contents of the documents, i.e. which logical structure underlies the documents, where these (logical) parts can usually be found or which textual information can be used to identify parts. This is similar to other formalisms for representing documents, e.g. the international standards ODA (Open Document Architecture, [ISO8613]) and SGML (Standard Generalised Markup Language, [ISO8879]).

However, for guiding the analysis process additional knowledge is necessary: one has to specify which knowledge should be used in which situation. Often, this knowledge is encoded directly into the analysis algorithm. We prefer to model the "How to analyse" explicitly by what we call strategic knowledge.

In the rest of this paper we present our initial application domain and try to motivate the concepts underlying our document analysis system. This is done in chapter 2. In chapter 3 we describe the knowledge representation language we have designed for that purpose. Chapter 4 presents some experimental results and we finish our paper with some concluding remarks in chapter 5.

2. The Application Domain

At the Fachhochschule Trier we want develop document analysis systems which can be adopted to new applications rather easily. Up to now we have essentially treated the domain of business letters and implemented our system called WINDOK (Knowledge based Interpretation of Documents). Tasks to be solved by our system are

- the classification of the document type
- the determination of relevant parts of a document, like the sender or the recipient and their constituents.

For analysing these documents we model layout knowledge and logical knowledge. Layout knowledge is related to the physical structure of documents. It concerns knowledge like "The recipient is almost always in the upper left quarter of the letter" or "An invoice usually contains an invoice table and items". Logical knowledge regards the content or the meaning of the physical parts like "The recipient of an invoice must be a customer (and must be found in the customer database)" or "A letter is usually signed by the sender".

This knowledge can be used for different analysis tasks for a class of documents. A major task for analysis processes is to attach logical parts to layout objects. Other tasks can be to identify only the address block(s) in a text or to find the total sum in an invoice. Therefore, we need to express relationships between layout and logical knowledge. Sometimes this is expressed by adding knowledge about relative positions of parts within a document or about typical textual contents but it can also be expressed through (mathematical) relations.

The possibility to specify complex relations between parts of documents (e.g. the name in the address part equals the name in the salutations of a letter) is an essential requirement for knowledge representation in our approach. Such relations can hardly be handled by very common knowledge representation frameworks, like frames [Minsky 74] or terminological logics, like KL-ONE based languages [Brachman&Schmolze 85]. The Problems occurring with the use of standard knowledge-representation Formalism and Standard Inference Mechanisms are:

- inadequacy of representation of documents because of the relation between physical structure (Layout) and semantics (Logic) of the Document.
- inefficiency of inference because document analysis specific strategic knowledge cannot be expressed

In other words: we do not mean that no existing, more or less general knowledge representation formalism could be used for document analysis purposes. However, we think that document analysis is a rather specific task where more efficient inference mechanisms can be designed when a special framework is used.

System Environment

Figure 1 illustrates the environment for the use of the system.

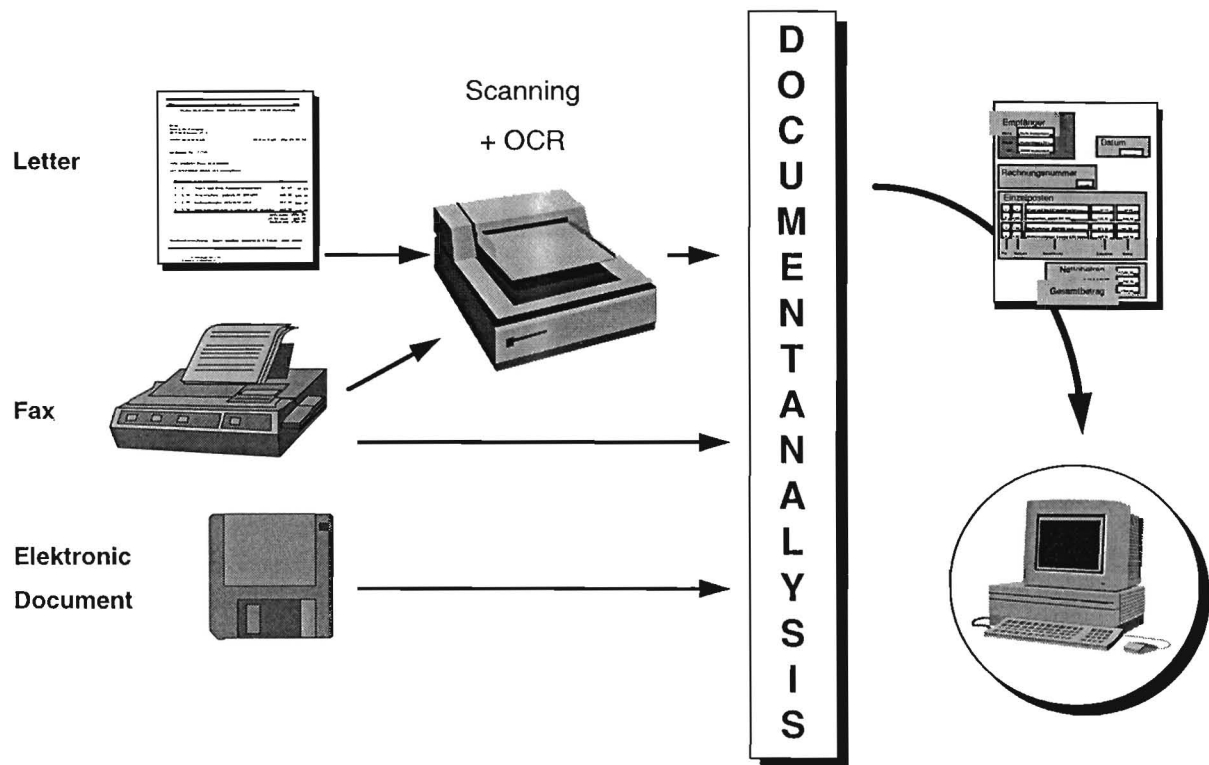


Figure 1. The environment of our document analysis system

System input may be an ASCII file or a printed (paper or fax) document. If a printed document has to be analysed, it is scanned (if necessary) and a commercial OCR-system is used to get ASCII text, which may be enriched with geometric information. This is input to the analysis process.

The WINDOK system consists of the components depicted in *Figure 2*.

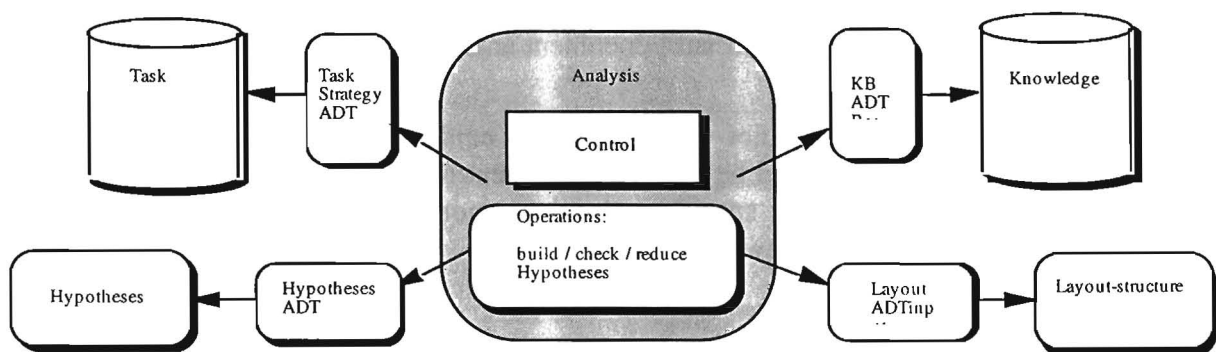


Figure 2. The main components of the WINDOK system

These components are described below.

Analysis Component

The main task of the analysis component is to build and to settle hypotheses about the meaning of the elements of the input document. The analysis component is divided into two main submodules: analysis-control and analysis-operators. The analysis-operators perform basic operations like building hypotheses for the meaning of certain parts of the document, or checking or rejecting certain hypotheses. Analysis-control guides the whole analysis process. That means according to the strategy definitions this component decides which operators are to be applied at which phase of the analysis. For that purpose the analysis-control component uses the information in the strategy definitions, which contain domain dependent knowledge about proper strategies for analysing certain parts of the input document.

The analysis-component has access to the following other components of the system:

- task and strategy definition
- layout structure of the input document
- knowledge base
- hypotheses

Access to these components is only allowed by certain selection-, creation- or modification-functions. That means, the data structures are realised as "Abstract-Data-Types" (ADT). In the first three cases access is only permitted to get some information (read-only access), these components are not changed in any way by the analysis operations, so this is the static part of the analysis. The access to the hypothesis component is performed in both directions, the hypotheses are created and deleted dynamically during analysis. Analysis-operators need information about the current state of interpretation and produce new understanding about the content of the input document, which is represented in the hypotheses component. By that, hypotheses are extended or refined until a final state is reached where the given tasks are solved, or no further conclusions are possible.

Task and Strategy Definition

The task definition specifies the definite task, i.e. it contains information, whether the type of the documents to be treated is known or whether this type has to be determined. Furthermore the task definition specifies which parts of the documents are to be searched for.

Strategy definitions may be specified for any class of documents or their parts and should contain domain dependent heuristic information, which is used by the analysis-component. Such heuristics may concern the order in which certain parts are to be searched for, which generic properties are to be considered first, or in which order hypotheses should be built, checked or reduced.

Layout Structure

The layout structure contains the result of pre-processing, building a special representation of the input document, consisting of text blocks. Each text block may contain several lines which are built up by words. Words, lines and blocks may be enriched by information about their geometric position on the input document.

Knowledge Base

The knowledge base contains the information of the typical content and structure of documents of a certain class like *letter* or *invoice*. These classes are described declaratively, including parts typically occurring in such documents, as well as relations between these parts. In order to be able to represent such information adequately, a special knowledge representation language has been designed, which is described in section 3.

The knowledge base is used by the analysis component to interpret the input information, i.e. objects of the input document (layout objects) like words, lines or text blocks are related to generic concepts or classes of the knowledge base. By that, the meaning of the layout objects is determined.

Hypotheses

The hypotheses component contains a description of the current state of analysis. For certain parts which are to be searched for or to be analysed, alternatives of interpretation are stored together with probability values. These intermediate solutions are refined by the analysis component until a terminating state is reached, representing the final solution. So, in object-oriented terminology, the hypotheses are partially filled instances of the frame templates which have to be completely filled during analysis.

3. Knowledge Representation

In this section we describe the knowledge representation language in more detail. With this language we can express logic, layout and strategic knowledge. To express strategic knowledge we use the *defstrategy* and *deftask* definition but we will not go into further detail for that. With a different strategy and task definition, different document analysis problems can be formulated and solved with more or less the same or similar knowledge bases.

We expect from knowledge representation language that it allows declarative and object-centered descriptions of analysis applications. The knowledge-base should be easy modifiable and good to understand. It should be possible to describe aggregation of objects, uncertain and vague knowledge. Different kinds of knowledge like layout, logic and strategic knowledge should be easy to express as well as relations between objects and parts of (other) objects. In the following sections we describe our solution to achieve these requirements.

For a declarative and object-centered knowledge representation, a frame-based language or a semantic net like representation could be chosen for example. We decided for a frame-based approach because of the built-in inheritance concepts and the possibility for integration of other kinds of concepts like predicate logic. In our language the standard frame concept is extended by descriptions of parts (part-of hierarchies), uncertainty and relations.

A description of an object in our document analysis language is built from the following (optional) elements:

- a name for the object (mandatory)
- the superframes of the object
- the parts of the object

- attributes of the object and its parts
- relations between parts

With these features a definition of a frame invoice can look like shown in figure 3.

```
(defframe Invoice
  (superframes business_letter)
  (parts (recipient_of_invoice      (frame recipiet))
         (number_of_invoice        (frame number_of_invoice)
         (invoice_table             (frame invoice_table))))))
```

Figure 3. An invoice frame

That means an invoice is a business letter with parts recipient of invoice, number of invoice and an invoice table . The structure of a recipient of an invoice is described in the frame recipient.

This frame corresponds to a semantic net like notation as follows (description of attributes see below).

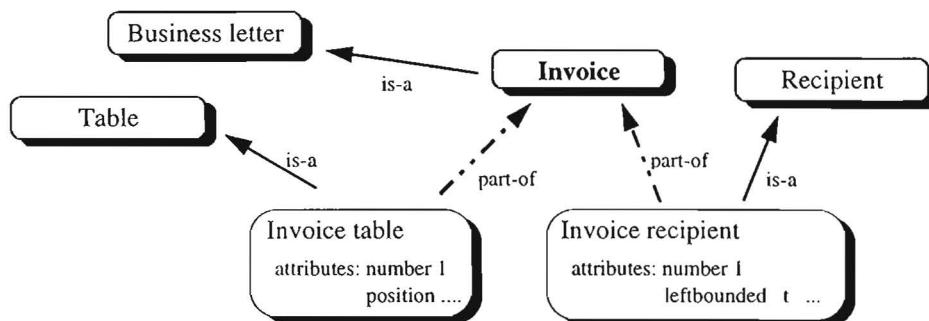


Figure 4. Semantic net representation of frame invoice

All frames or part frames can have parts again, so there is an aggregation (= part) hierarchy in addition to the superframes (= is-a) hierarchy which contains all objects from document down to word or character.

To express the knowledge which is needed during the analysis process we have integrated attributes with several annotations into the frames. Predefined annotations are:

- type (e.g. integer, boolean, real, string....)
- value, used for fixed values like page-width

- range, used to restrict the domain of values (enumeration or interval)
- relevance, for relevance expressions (see below)
- compute-function, used to determine a value for instantiation and testing

For the expressions of uncertainty we use certainty factors [Shortliffe&Buchanan 75] in relevance annotations of attributes. Annotations can have a measure of belief and a measure of disbelief which get reckoned up during analysis. The frame for an invoice table can be described as follows:

```
(deframe invoice_table
  (number (type integer)
    (value 1))
  (parts (headings (frame heading)
    (number (type integer)
      (value 1))
    (position_first_line (type integer)
      (range (20 25))
      (relevance ((20 22) 0.8 0.0)
        ((23 25) 0.4 0.0))))
    (number_of_words (type integer)
      (range (5 7))
      (relevance (5 0.7 0.0)
        ((6 7) 0.9 0.0))))
    (item (frame item)
      (number (type integer)
        (value 1))
      (position_first_line (type integer)
        (range (21 26))
        (relevance ((21 23) 0.8 0.0)
          ((24 26) 0.4 0.0))))
    (amount (frame amount) .....))
  (sum (frame sum)....))
```

Figure 5. An invoice table frame

This means that there is only one invoice table with part headings where the relative position of the first line is between 20 and 25. For other values of position, no hypothesis can be built and so no relevances are given. The compute-function or other annotations of the attribute may be defined somewhere else in the knowledge base, either in a superframe of this frame or in an additional defattribute construct for global definition of annotations. The other parts and their attributes are defined similar.

The relations between frames and parts mentioned until now regard only is-a and part-of relations. A description of all relationships between possible objects that could be useful (or helpful) for document analysis tasks. In our knowledge representation language it is possible to

model arbitrary relations between frames and their parts or frames and other frames. The relations can either be used to reduce hypotheses or to build new hypotheses with instances which fulfil the constraints given by the relations.

Relations are, like attributes, defined within the frame definitions. But as they normally have an arity greater than one, they refer to several parts or the frame itself. In Figure 6 is an example of some of the relations of the frame whose attributes we have already shown above.

The first relation describes that the headings, must begin above the items the third relation describes that the part item number of the heading must be located above the item number part of any single item. Here, parts of parts of frames are needed to properly express the relation.

```
(deframe Invoice_table ...  
  .(attributes and parts see above)  
  (relations      (< (headings position_first_line)  
                    (item position_first_line))  
                  (< (item position_first_line)  
                    (amount position_first_line))  
                  (over (headings number_of_item)  
                        (item single_item number_of_item))  
                  (over (headings description)  
                        (item single_item description)))
```

Figure 6. Relations of an invoice table frame

With these relations all instances (words, lines...) that fulfil the relations can be found and can serve as good hypotheses for analysis. Ideally, when modelling of the relations is done well, there are only a few candidates to be checked further, in strategy definitions concerning other frames or other relations.

4. Experimental Results

So far we have tried to solve the following problems:

- Analysis of ASCII texts with addresses of companies and authorities in text flow
- Classification of document types (Invoice, Order, Offer)

- Analysis of Invoices with address, table and items.

As an example we want to show here the analysis of an invoice (similar work was done by [Köppen&al 96]). The document image is pre-processed by a common OCR-software providing text and layout information which is then transformed into our internal representation.

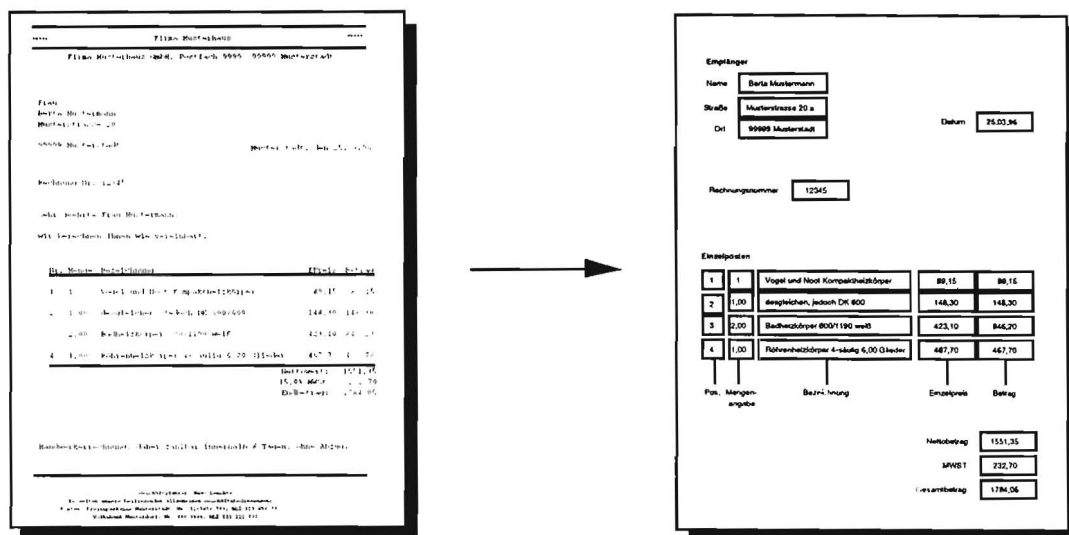


Figure 7. Example of processing an invoice

The analysis of address date and invoice table is done according to the strategies as described in chapter 2 and assisted by knowledge about their structure given in the knowledge base. Address, date and table are analysed following the strategies which do not rely on a certain invoice template but are flexible in order to analyse all kinds of invoices. The results we obtained are shown in figure 7. This data obtained through Document Analysis can be put into a data-base and can accomplish the company memory.

We tested our System with a test sample of about 50 Business Letters (mostly invoices) and obtained the results shown in figure 8.

%	correct	incorrect	not analyzed
recipient	69,57	2,17	28,26
date	83,72	0	16,28
single items	71,80	2,56	25,64
sum	84,09	6,82	9,09

Figure 8. Test Results

For the analysis of different problems, tasks and strategies have to be changed and the knowledge base has to be adapted to the application domain.

5. Conclusion and Outlook

We consider knowledge representation as an essential and central activity for the development of document analysis systems. We model knowledge about the structure and the content of documents of the actual application domain and strategic knowledge to guide the analysis process itself.

Our specialised knowledge representation formalism allows for a specification of all these kinds of knowledge. Another issue of this approach is that descriptions of documents can be understood and maintained also by non specialist users. Furthermore, existing descriptions can be reused more easily in new application domains.

The system is implemented in Allegro Common Lisp and runs on different platforms, like Apple Macintosh, Sun and IBM PC.

Currently, we are working on several domains (Letters, Invoices, Fax-Messages, ...) to gain some insight in the effort that has to be spent when a system for a new domain is set up. Our first experiences are encouraging. Furthermore, we work on extensions of the knowledge representation language which are necessary to cope with other classes of documents. Furthermore, some enhancements concerning uncertainty will be made by implementing uncertainty formalisms which fit better for our document analysis tasks.

6. Acknowledgements

All authors wish to thank the DFKI GmbH for giving them the opportunity to participate in the OMEGA and the PASCAL 2000 projects of the Document Analysis group. This has inspired our work and helped us to achieve the results presented in this paper.

7. References

- [Bayer 93a] T. Bayer. *Understanding Structured Documents By a Model Based Document Analysis System*. Proceedings of the 2nd International Conference on Document Analysis and Recognition (ICDAR), pages 565 - 568, Tsukuba Science City, Japan, 1993
- [Bayer 93b] T. Bayer. *Ein modellgestütztes Analysesystem zum Bildverstehen strukturierter Dokumente*. Disk 44, Infix, St. Augustin, Germany, 1993. *In German*
- [Bayer&al 94] T.A. Bayer, U. Bohnacher, H. Mogg-Schneider. *InfoPortLab - An Experimental Document Analysis System*. Proceedings of the IAPR-Workshop on Document Analysis Systems, Kaiserslautern, Germany, 1994
- [Bläsius&Hönes 95] K.-H. Bläsius, F. Hönes. *A Mechanism for Generating Image Analysis Algorithms*. Proceedings of the Workshops KI'95, Bielefeld, Germany, 1995
- [Brachmann&Schmolze 85] R.J. Brachmann, J.G. Schmolze. *An Overview of the KL-ONE Knowledge Representation System*. Cognitive Science 9, pages 171-216, 1985
- [Dengel 89] A. Dengel. *Automatische Visuelle Klassifikation von Dokumenten*. Doctoral Thesis, Universität Stuttgart, 1989. *In German*
- [Dengel&al 94] A. Dengel, R. Bleisinger, F. Fein, R. Hoch, F. Hönes, and M. Malburg. *OfficeMAID - A System for Office Mail Analysis, Interpretation and Delivery*. Proceedings of First International Workshop on Document Analysis Systems (DAS'94), pages 253-275, Kaiserslautern, Germany, October 18-20 1994
- [Dengel&Dubiel 95] A. Dengel and F. Dubiel. *Clustering and Classification of Document Structure- A Machine Learning Approach*. Proceedings of the Int. Conference of Document Analysis and Recognition (ICDAR '95), Montreal, Canada, 1995
- [ISO8879] ISO 8879. *Information Processing- Text and Office Systems - Standard Generalised Markup Language (SGML)*. 1986
- [ISO8613] ISO 8613, *Information Processing- Text and Office Systems -Office Document Architecture (ODA) and Interchange Format*. vol. I-III, parts 1-8, 1988
- [Köppen&al 96] M. Köppen, D. Waldörtl, B. Nickolay. *A System for the automated evaluation of invoices*. Proceedings International WS on Document Analysis Systems (DAS '96), pages 3-21, Philadelphia, 1996

- [Minsky 74] M.Minsky. *A Framework for Representing Knowledge*. MIT AI Lab. AIM 306, Cambridge, 1974
- [Pasternak 94] B. Pasternak. *Processing Imprecise and Structural Distorted Line Drawings by an Adaptable Drawing Interpretation Kernel*. Proceeding of the IAPR-Workshop on Document Analysis Systems, pages 349 - 365, Kaiserslautern, Germany, 1994
- [Pasternak & Neumann 93] B. Pasternak, B. Neumann. *Adaptable Drawing Interpretation Using Object-Oriented and Constraint Based Graphic Specification*. Proceedings of the 2nd Int. Conference on Document Analysis and Recognition (ICDAR), pages 359 - 364, Tsukuba Science City, Japan, 1993
- [Shortliffe&Buchanan 75] E.H Shortliffe, B.G Buchanan.. *A Model for Inexact Reasoning in Medicine*. Mathematical Biosciences 23, pages 351 -379, 1975
- [Wenzel&al 96] C. Wenzel, S. Baumann, T. Jäger. *Document Classification by Voting of Competitive Approaches*. Proceedings of International Workshop on Document Analysis Systems (DAS '96), pages 352-374, Philadelphia, 1996

ERBUS - Towards a Knowledge Management System for Designers¹

Manfred Daniel^a, Stefan Decker^b, Andreas Domanetzki^c, Elke Heimbrot-Habermann^e, Falk Höhn^f, Alexander Hoffmann^d, Holger Röstel^d, Regina Smit^c, Rudi Studer^b, Reinhard Wegner^e

^aFachhochschule Schmalkalden, D-98574 Schmalkalden
e-mail: daniel@informatik.fh-schmalkalden.de

^bInstitute AIFB, University of Karlsruhe (TH), D-76128 Karlsruhe
e-mail: {decker | studer}@aifb.uni-karlsruhe.de

^cSmit & Partner, Schwabacher-Str. 59, D-90763 Fürth
e-mail: CIDes@aol.com

^dWerkstatt für Design und Informatik, Bernsdorfer-Str. 210-212,
D-09126 Chemnitz
e-mail: {hoffmann | roestel}@tcc-chemnitz.de

^eHochschule für Kunst und Design, Neuwerk 7, D-06003 Halle
e-mail: {haberman | wegner}@burg-halle.de

^fFachhochschule Hannover, D-30419 Hannover
e-mail: falk@kd.fh-hannover.de

Abstract

The goal of the BMBF project WORKS is to develop a methodology and a prototype for knowledge management systems: ERBUS is aimed at the support of designer and engineers, especially in ergonomic questions. Because of the nature of the knowledge available in ergonomics, it turned out that a conventional knowledge based approach was insufficient. Instead an integrated approach is necessary, which supports the whole design process and different kinds of knowledge. This paper delivers first results of the project: requirements for a general methodology for building knowledge management systems and tools and techniques for ERBUS.

1 Introduction

1.1 Outline of the project

The aim of the project WORKS (**W**ork **O**riented Design of **K**nowledge **S**ystems) was to build a knowledge based system ERBUS (**E**rgonomie-**B**eratungs und **U**nterstützungs-**S**ystem) to support industrial designers concerning ergonomic question. This is due to the fact, that a large amount of ergonomic knowledge exists, but this knowledge is often neglected from the designer of indus-

1. The work reported in this paper is partially supported by the BMBF (German Ministry of Education, Science, Research and Technology) under grant number 01HS014

trial products. However, in the course of the project it turned out that "just" a KBS is not enough: large portions of knowledge can only be represented informally, knowledge is built up through lessons learned and the made experience, and many process steps in the design process have to be supported and integrated to avoid media breaks and to encourage the designer to use the system. Therefore we are aiming now for a knowledge management system, incorporating many techniques.

1.2 Knowledge Management

The development of knowledge-based systems is dominated by a technology-oriented approach, aiming at a complete modelling and implementation of human problem solving methods. This approach does not address the question of usefully supporting the human work processes. To remedy this knowledge management is aiming at the improvement of knowledge work processes (cf. [Dav96]). So achieve this goal several techniques have to be integrated, rather than focusing on one single technique: e.g. business process reengineering (cf. [Sch94]), knowledge based systems with problem solving methods, workflow management systems, CSCW-techniques like answer garden systems (cf. [AcM90][AcM96]), hypertext and database resp. document management systems (cf. [Sku97]). However, the task to support are not pretty standard, then specialized techniques are probably not applicable. So techniques like those mentioned above have to be integrated in an overall framework, but this framework has to be so flexible, that it can be adjusted to most business processes. Furthermore it is usually too expensive to build a knowledge management system from scratch - instead existing products and components have to be used to construct a knowledge management system (KMS).

The rest of the paper is organized as follows: at the first step we model the design process used in industrial design. At the next step we identify possible tasks to support and approaches for this support. From the results of the former steps drawbacks of current knowledge engineering methodologies, namely MIKE (Model Based Incremental Knowledge Engineering), with respect to building knowledge management systems, are shown.

2 Modelling the Design Process

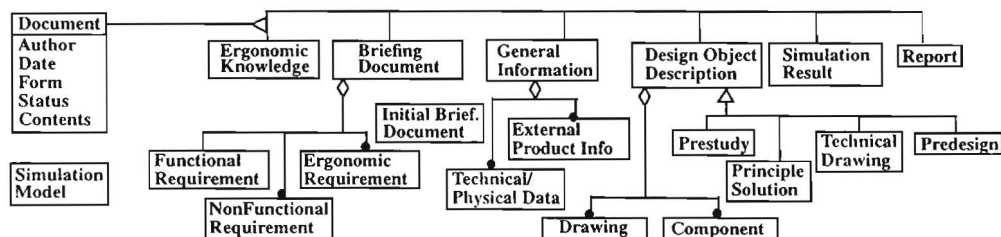


Fig. 1 . Data View

As Kühn and Abecker stated in [KüA97] the first thing to ask is: *What are the tasks to be supported?* To answer this question, i.e. for the requirements elicitation of the overall system the process of industrial design was modelled and analysed for support needs in some design enterprises. This is also the first step, where a general methodology has to start: it has to support the model-

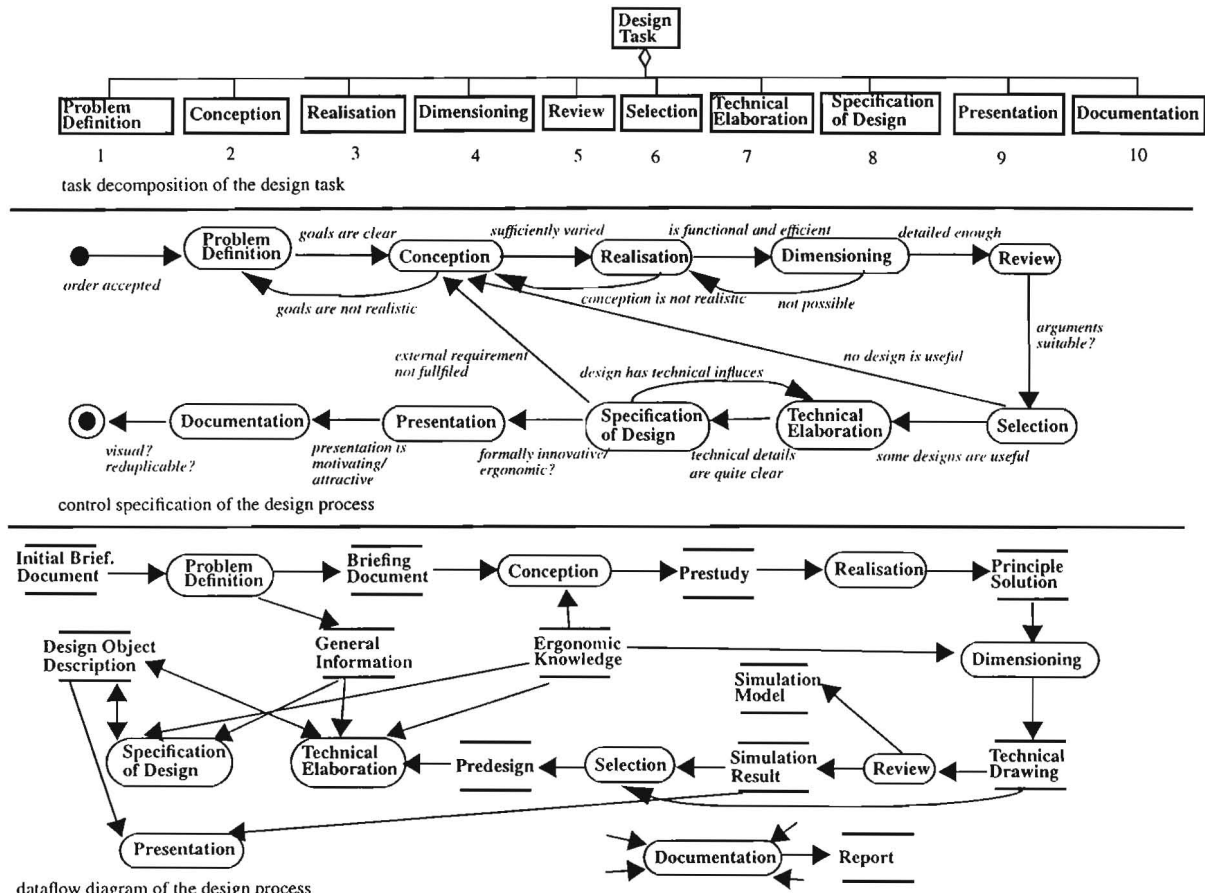


Fig. 2 . Design Process

ling process with modelling primitives and tools. Therefore the first step is just business process modelling as performed in [Sch94] or described in [KiB94]. The process is described in OMT (cf.[RBP+91][KKM95][BKM94]), using an generic reference schema for enterprises (cf. [DES97]). Because only small companies were examined, we only modelled a small subset of the whole set of views: the data view (cf. figure 1), describing the structure of objects and documents used in the processes and the process view itself (cf. figure 2). For the description we use a task decomposition, a statechart to express control over the subtasks, and a dataflow diagram to express the dataflow between the subtasks. Of course is design a creative and highly cyclic process, however, certain behaviour patterns can be identified. Depending on the type of tasks and the class of design (cf. [VDI86]) some stages or the process model may be left out altogether or may deserve less emphasis.

3 Identifying Tasks to Support

Using the modelled data view and the process (cf. figure 1 and figure 2) possibilities for supporting the task can be identified. Realising these possibilities usually introduces new data objects or new process steps.

E.g. throughout the stages 1 and 2 support should be given for the elaboration of briefing and requirements documentation as well as for conceptualisation. This can be done by reference design documents, collected in former projects and reused in the actual project. For this purpose similar properties and problem solutions are identified within the reference design documents. The KMS will then provide a proposal which can be modified. At the same time, ergonomic

issues (problems) are identified and solutions are proposed based on the reference designs and rules established for the ergonomic consultation. Eventually, a list of requirements is being drafted, containing properties and their respective values. Created documents should be managed by intelligent document/database management systems and they should be available for use in further projects.

The designer's required knowledge and abilities are particularly complex during stages 4 - 8. In some areas, such as ergonomics or production technology and construction, specific information is difficult to obtain and probably not contained in to KMS. Therefore consulting an expert would be necessary and should be supported.

During stages 4 to 6, detailed information and refined knowledge the elaboration of the tasks is required. This could be derived from information obtained from available reference designs as well as from the properties defined in stages 1 and 2.

The technical elaboration can be a rather cumbersome task, because the overall structure is defined but the details are still open. However, because components and requirements of the design object (the goal) are well known in stages 7 and 8, one alternative is to support the elaboration with a KBS for parametric design (cf. [MoZ96]).

Support throughout stages 1,2,9 and 10 will be given mostly in the form of provision of information and processing of documents (resp. old designs). Needs for ergonomic consultation can arise in stage 1 and 2 in the context of the definition of requirements.

During all other stages need for ergonomic facts can emerge. This knowledge is complex and in part highly sophisticated. A comprehensive processing which focuses on practical needs can cover most of the knowledge required by the designer. The demand for the knowledge-based component depends upon the depth of consultation desired during the various stages.

The industrial design process, in general, is very sensitive to input from the outside world. Therefore, means of support, ranging from information research to consultancy, should be adaptable to changes in the real world.

A task not directly modelled in figure 2 is the maintenance and revision of the design object. The collection of the major design decisions during the whole design process together with a documentation helps to avoid doing work twice.

4 Tools & Techniques for Realising Knowledge Management in ERBUS

4.1 General System Requirements

The results of single design stages in the modelled process are represented by more or less structured documents. These are usually simple textfiles as well as complex CAD-models or SGML-documents. To enable an efficient document management the retrieval of documents has to be supported. The documents are annotated with parts of an ontology (cf. [FES97]). This can be used for retrieval operations. Furthermore the documents linked using a hypertext system, where

links are enriched with semantics (for example: attribute values like "solves a problem ..." or "in opposition to ...").

In the same way ergonomic knowledge is available: by means of an ontology the actual project documents are linked to the document base, which contains the ergonomic knowledge in a semi-formal form.

4.2 Design Decision Storing

We consider the (almost) complete and retrievable storing of (almost) every design decision case made as an important knowledge support task in the design field. This enables the reusability and maintainability of former designs, because reasons for design decisions are known and can be reused (cf. [Lan94]). Traceability of design decisions should be primary based on links respectively link chains between documents (respectively their structured elements). A correspondingly designed user interface has to support searching and assignment of these links. Primary goal is the (almost) complete and retrievable storing of design decision cases. The semantic network suggested a "well-suiteable track" to cultivate this *Corporate Memory* by means of techniques developed for case based reasoning. By this means we are expecting to reach a significant higher degree of knowledge reusing (directly and/or after adapting), when a comparable task will appear in the future.

At current stage we are implementing these functionality as an ERBUS-client able to store not only the results but also the history and starting-point of design decisions in a complete and retrievable manner as an extension of an existing Document Management System.

4.3 Case-Based Reasoning

Reference designs lead naturally to the area of case based reasoning: designs and requirements elaborated in former projects are stored in a case base, and retrieved with appropriate similarity measures. To apply case based reasoning appropriate attributes of the documents have to be defined, which are useful for case retrieval.

4.4 Workflow & Design Process Management

Although the modelled design process is often carried out by one person, the introduction of a workflow management system often makes sense: the design department is usually integrated in enterprise wide business processes, e.g. with an engineering department etc. So the seamless integration of the design processes can be guaranteed.

4.5 Parametric Design

At the "Technical Elaboration" step of the design process most of the design object components, possible assignments for the components, constraints and requirements are known. This configuration enables the usage of a problem solving method for parametric design ([MoZ96]). The tasks of parametric design can be described as follows: given a set of parameter (e.g. of a design object), a valid value range for each parameter, and a set of constraints with respect to the values of the parameters. Find an assignment from the value range to the parameters, such that all con-

straints are satisfied. To link the KBS's for parametric design to the overall process, the input/output of the problem solving method used by the parametric design problem solver has to be connected to the overall design process. For an example cf. [DES97].

4.6 CSCW-Techniques

Because the ergonomic knowledge is large, informal, sophisticated and changing a system can never have complete knowledge. Therefore techniques have to be established, which enable a user to ask questions easily to appropriate experts. This is the focus of systems like answer garden ([AcM90][AcM96]): a knowledge map of the enterprise is build and if a question can not be answered by the system, the question is routed forward to an expert. This realizes up another part of the organizational memory.

5 Drawbacks of MIKE

5.1 MIKE: a Methodology for Developing Knowledge Based System

The current MIKE (Model-based and Incremental Knowledge Engineering) methodology as described in [AFS96][Neu93] defines an engineering framework for eliciting, interpreting, formalizing, and implementing knowledge in order to build KBSs. It aims at integrating the advantages of life cycle models, prototyping, and formal specification techniques into a coherent framework for the knowledge engineering process. Subsequently, we will discuss the main principles and methods of MIKE.

In contrast to other approaches which assume that the expert creates the model himself, it is assumed that the knowledge engineer is the moderator of this modelling process. Within the modelling process a large gap has to be bridged between informal descriptions of the expertise which have been gained from the expert using knowledge elicitation methods and the final realization of the KBS. Dividing this gap into smaller ones reduces the complexity of the whole modelling process because in every step different aspects may be considered independently from other aspects.

The knowledge gained from the expert in the elicitation phase is described in natural language. It mainly consists of interview protocols, protocols of verbal reports, etc. These knowledge protocols define the *elicitation model* ([Neu93]). This knowledge represented in natural language must be interpreted and structured. The result of this step is described semi-formally in the so-called *structure model* ([Neu93][Neu94]), using predefined types of nodes and links.

According to the KADS approach the knowledge-level description of the functionality of the system is given in the *model of expertise* (cf. [SWB93]). For describing the model of expertise in a formal way the formal and operational specification language KARL ([Ang93][Fen95]) has been developed. KARL is based on first order logic and dynamic logic and offers language primitives for each of the three different layers of the model of expertise. The contexts of the structure model correspond to the domain-, task-, and inference layer of KADS model of expertise.

For designing KMS's the techniques used in MIKE can be used in two ways:

- MIKE can be used for modeling and specifying the knowledge management system itself.

-
- MIKE can be used during the operation of the knowledge management system, e.g. for modeling of new knowledge, that has to be integrated somewhere into the system. According to the philosophy of MIKE this means incorporating a Knowledge Engineer into the process. We therefore have to distinguish two modeling tasks: modeling mission critical knowledge and modeling simple knowledge as used in everyday questions (like in Answer Garden). MIKE is usually more suited to the first task.

In the following we mainly focus on the first aspect.

5.2 Drawbacks of MIKE

MIKE as described above has several drawbacks when developing knowledge management systems:

- MIKE is aiming at conventional KBS. Also business process modeling primitives for MIKE are defined, there is no possibility to model other techniques besides KBS's. But as shown above the development of a knowledge management system includes many different techniques.
- Knowledge is modeled by a knowledge engineer in the modeling phase. Furthermore the knowledge representation in the modeling phase and in the final system is usually totally different. Knowledge Management Systems live from the fact, that knowledge is integrated in the whole system in a dynamic way, e.g. the organizational memory is extended when necessary within the normal workflow.
- MIKE (resp. the concept of a problem solving method) presupposes only the existence of one user who only partially interacts with the final system. This, however, is not true for knowledge management systems: they are oriented towards the maximum support for knowledge workers, but this is only possible with a high degree of interaction. This interaction is done with the system as well as with other humans, often in a nonformal way.

6 Conclusion and Future Work

The design process for industrial design has been modeled and analysed. From this analysis conclusions for a knowledge management system for process support are drawn. Techniques usable for a knowledge management system are e.g. case based reasoning, document management, conventional knowledge based systems, and CSCW-tools. Also there are many proposals for realizing corporate memory with each single technology, there isn't an integrated approach like our proposal. However, the final proof of concept is the implementation, which is ongoing work.

Also work is necessary to extend MIKE to a methodology for knowledge management systems: using results presented in [Sta96], where tools are integrated into the framework by converting them to agents. Then they are modeled with a special agent modeling tool. Together with the enterprise reference scheme presented in [DES96][DES97] for business processes and knowledge based systems this could deliver a general framework for modeling knowledge management systems. To enable prototyping the resulting model should be executable in some sense, at least it should enable validation.

7 References

- [AcM90] M. S. Ackerman, T. W. Malone: Answer Garden: A Tool for Growing Organizational Memory. In ACM Conference on Office Information Systems (Cambridge, MA, April, 1990), pp. 31-39.
- [AcM96] M. S. Ackerman, D. W. McDonald: Answer Garden 2: Merging Organizational Memory with Collaborative Help. In: Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'96), pp. 97y-105.
- [AFS96] J. Angele, D. Fensel, and R. Studer: Domain and Task Modeling in MIKE. In: Proceedings of the IFIP WG 8.1/13.2 Joint Working Conference, Domain Knowledge for Interactive System Design, Geneva, May 1996.
- [Ang93] J. Angele: Operationalisierung des Modells der Expertise mit KARL (Operationalization of the Model of Expertise with KARL), Ph.D. Thesis in Artificial Intelligence, No. 53 , infix, St. Augustin, 1993 (in German).
- [BKM94] B.M. Bauer, C. Kohl, H.C. Mayr, J. Wassermann: Enterprise Modeling using OOA Techniques, In: Proceedings Connectivity '94: Workflow Management - Challenges, Paradigms and Products. München: R. Oldenbourg-Verlag, 1994, pp. 96-111.
- [BrV94] J.A. Breuker and W. Van de Velde (eds.). The CommonKADS Library for Expertise Modeling. IOS Press, Amsterdam, 1994.
- [Dav96] T. Davenport: Some Principles of Knowledge Management, URL:<http://www.bus.utexas.edu/kman/kmprin.htm>, 1996
- [DES96] S. Decker, M. Erdmann, R. Studer: A Unifying View on Business Process Modelling and Knowledge Engineering. In: Proceedings of the 10th Knowledge Aquisition Workshop (KAW 96), Banff, Canada, November 1996.
- [DES97] Stefan Decker, M. Erdmann, R. Studer: An Enterprise Reference Scheme for Integrating Model Based Knowledge Engineering and Enterprise Modelling. In: Proceedings of the EKAW'97, LNCS, Springer, 1997
- [Fen95] D. Fensel: *The Knowledge Acquisition and Representation Language KARL*, Ph. D. Thesis, University of Karlsruhe, 1993. Kluwer Academic Publisher, Boston, 1995.
- [FES97] D. Fensel, M. Erdmann, R. Studer: Ontology Groups: Semantically Enriched Subnets of the WWW, Workshop Intelligent Information Integration, 1997, URL:<http://www.informatik.uni-bremen.de/~wache/i3-ki97/papersD.html>
- [KiB94] M. Kirikova and J.A. Bubenko: Software Requirements Acquisition through Enterprise Modeling. In: Proc. 6th Int. Conf. on Software Engineering and Knowledge Engineering (SEKE'94), Jurmala, 1994.
- [KKM95] R. Kaschek, C. Kohl, H. C. Mayr: Cooperations - An Abstraction Concept Suitable for Business Process Re-Engineering. In: J. Györköös, M. Krisper, H.C. Mayr (eds.): ReTIS'95 - Re-Technologies for Information Systems. OCG Lecture Notes 80, Oldenbourg, 1995.
- [KüA97] O. Kühn, A. Abecker: Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges, Journal of Universal Computer Science, submitted.
- [Lan94] D. Landes: DesignKARL - A language for the design of knowledge-based systems. In: *Proc. 6th International Conference on Software Engineering and Knowledge Engineering SEKE'94*, Jurmala, Lettland, 1994, 78-85.
- [MoZ96] E. Motta, Z. Zdrahal: Parametric Design Problem Solving. In: Proceedings of the 10th Knowledge Aquisition Workshop (KAW 96), Banff, Canada, November 1996.
- [Neu94] S. Neubert: Modellkonstruktion in MIKE - Methoden und Werkzeuge. Ph.D. Thesis in Artificial Intelligence, No. 60, infix, St. Augustin, 1994 (in German).
- [Neu93] S. Neubert: Model construction in MIKE (Model-based and Incremental Knowledge Engineering). In: N. Aussenac et.al. (eds): Proceedings of the 7th European Workshop EKAW'93 (Toulouse), LNAI 723, Springer-Verlag, Berlin, 1993.
- [RBP+91] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, W. Lorensen: Object-Oriented Modeling and Design. Prentice Hall, Englewood Cliffs 1991.
- [Sch94] A.-W. Scheer: Business Process Engineering: Reference Models for Industrial Enterprises, Springer, 2nd ed. 1994.
- [Sku97] D. Skuce: Hybrid KM: Integrating Documents, Knowledge Bases, Databases, and the Web. In: Proceedings of AAAI Spring Symposium on Artificial Intelligence in Knowledge Management. 1997. URL: <http://ksi.cpsc.ucalgary.ca/AIKM97/AIKM97Proc.html>
- [Sta96] J. Stader: Results of the Enterprise Project. In: Expert Systems '96, the 16th Annual Conference of the British Computer Society Specialist Group on Expert Systems, 1996. URL: <http://www.aiai.ed.ac.uk/~enterprise/enterprise/pub.html>

-
- [SWB93] G. Schreiber, B. Wielinga, and J. Breuker (eds.): *KADS. A Principled Approach to Knowledge-Based System Development*, Knowledge-Based Systems, vol 11, Academic Press, London, 1993.
- [VDI86] VDI 2242: Konstruieren ergonomiegerechter Erzeugnisse. pp. 1-2, Düsseldorf, 1986J.

Wissensmanagement mit CoMo-Kit

Barbara Dellen, Harald Holz, Gerd Pews

Universität Kaiserslautern

Postfach 3049, 67653 Kaiserslautern

email: {dellen | holz | pews}@informatik.uni-kl.de

In diesem Papier beschreiben wir, welche Teile eines Unternehmensgedächtnisses mit dem Werkzeug CoMo-Kit verwaltet werden, wie dies geschieht und welchen Nutzen ein Unternehmen davon hat. Als flexibles Workflowmanagement-System kann CoMo-Kit Arbeitsprozesse und deren Abwicklung sowie Gründe für Entscheidungen dokumentieren, vorstrukturieren und über ein WWW-basiertes Informationssystem präsentieren. Weiter bildet die Reaktion auf Änderungen, sowie Um- und Weiterplanung einen Kernpunkt des Systems.

1 Einleitung

In den letzten Jahren hat sich ein Trend fortgesetzt, der Unternehmen in besonderer Weise herausfordert: steigende Anforderungen an die Qualität von Produkten und Dienstleistungen, steigende Komplexität der Arbeitsabläufe, während gleichzeitig der weltweite Markt die Unternehmen dazu zwingt, ihre Ressourcen optimal zu nutzen.

Vor diesem Hintergrund gewinnt der rechnergestützte Umgang mit der Ressource „Unternehmens-Know-How“ eine besondere Bedeutung. Das Werkzeug CoMo-Kit (Concept Modelling Kit), das an der Universität Kaiserslautern entwickelt und prototypisch implementiert worden ist, verwaltet Wissen über Arbeitsabläufe als einen der zentralen Aspekte eines Unternehmensgedächtnisses. Das Spektrum reicht hierbei von der Dokumentation, in der ein Unternehmen festlegt, wie Arbeitsprozesse durchgeführt werden sollten, über das Projektmanagement für ein konkretes Vorhaben bis hin zur Bereitstellung von Informationen über frühere Projekte.

Dokumentation typischer Arbeitsabläufe. Arbeitsabläufe werden mit CoMo-Kit durch elementare Arbeitsschritte beschrieben. Bei der Definition von Arbeitsschritten werden zu ihrer Durchführung benötigte Informationen, Produkte und Ressourcen berücksichtigt. Die Reihenfolge, in der die Arbeitsschritte ausgeführt werden, kann beliebig spezifiziert werden: feste Sequenzen, willkürliche Anordnungen sowie Nebenläufigkeit sind realisierbar. CoMo-Kit stellt dazu eine Modellierungssprache und ein Modellierungswerkzeug bereit.

Unterstützung bei der Projektabwicklung. CoMo-Kit stellt darüber hinaus Projektmanagementfunktionalität zur Verfügung. Die modellierten Arbeitsabläufe werden interpretiert und von dem Managementwerkzeug koordiniert. Durch Arbeitslisten erfahren die Bearbeiter anstehende Arbeitsschritte, haben Zugriff auf Kontextinformationen, führen die Arbeiten durch und teilen dem Managementwerkzeug die Ergebnisse ihrer Arbeiten mit. Das Werkzeug unterstützt dabei

- die Verteilung von Arbeitsschritten über ein Rechnernetz,

- den Informationss Austausch zwischen den Bearbeitern,
- die gezielte Bereitstellung von Informationen,
- die Konfiguration und den Aufruf benötigter Werkzeuge,
- die Dokumentation des Projektablaufs und
- die Änderung der Arbeitsabläufe „on the fly“.

Der Fortschritt des Projektes und Änderungen im Arbeitsablauf erfordern eine laufende Anpassung des verwalteten Wissens. Dazu gehören z.B. die Aktualisierung der Arbeitslisten nach Beenden eines Arbeitsschrittes sowie das Benachrichtigen von einer Änderung betroffener Bearbeiter. Um solche Koordinationsaufgaben übernehmen zu können, muß das Werkzeug Kenntnis von den kausalen Zusammenhängen im Projekt haben. Die Entwicklung von Techniken zur Aquisition und Verwaltung kausaler Abhängigkeiten ist daher zentrales Thema unserer Forschungsarbeiten.

Konservieren der Produkte und der Entscheidungsgründe. Bei der Planung und Durchführung von Arbeitsschritten werden Entscheidungen getroffen. Diese Entscheidungen determinieren den Stand und den Fortgang des Projektes. Um Entscheidungen (insbesondere solche, die den Projektverlauf ändern) treffen zu können, ist Kontextwissen notwendig. Dazu gehören insbesondere die Kenntnis des bisherigen Projektverlaufs und die Gründe für vorangegangene Entscheidungen. In CoMo-Kit werden daher Entscheidungsketten protokolliert. Zu jeder Lösungsalternative können Gründe für oder gegen ihre Wahl formuliert und vom System verwaltet werden.

WWW-Information zu Produkten und Arbeitsschritten. Um den Projektbeteiligten über das für einen Arbeitsschritt benötigte Wissen hinaus Einsicht in den Zustand des Projektes zu gewähren, kann auf Projektinformationen über das WWW zugegriffen werden. Der Benutzer hat die Möglichkeit, den Gesamtzusammenhang zwischen den Arbeitsschritten dargestellt zu bekommen, den Zustand einzelner Arbeitsschritte zu erfragen, sowie bereits erstellte Produkte einzusehen.

Im folgenden werden wir näher beschreiben, welche Bereiche eines Unternehmensgedächtnisses durch CoMo-Kit verwaltet und welche Techniken dazu eingesetzt werden.

2 Wissensmanagement mit CoMo-Kit

Abbildung 1 skizziert die Grobarchitektur von CoMo-Kit. Es lassen sich drei Komponenten unterscheiden:

- Der *Modeler* zur Definition und Modifikation allgemeiner, typischer Arbeitsabläufe, die im Unternehmen vorkommen. Die hier beschriebenen Abläufe sind generisch in dem Sinne, daß beispielsweise lediglich die Struktur der auftretenden Produkte in einem Datenmodell festgelegt und für jede in dem Ablauf vorkommende Aufgabe mehrere zulässige Vorgehensweisen zur Bearbeitung der Aufgaben vordefiniert werden können. Die Auswahl einer bestimmten Vorgehensweise für ein konkretes Projekt erfolgt erst bei dessen Planung.
- Der *Scheduler* zur Unterstützung der Planung und Abwicklung von konkreten Projekten. Der Server verwaltet den aktuellen Projektzustand, kausale Abhängigkeiten und alle im Projektverlauf erzeugten Informationen. Die einzelnen Clients entsprechen den Arbeitsplätzen einzelner Bearbeiter und übernehmen die Kommunikation mit dem Server. Durch Rückgriff auf den Modeler kann der aktuelle Plan für ein Projekt jederzeit verfeinert oder verändert werden.

- Der *Information Assistant* erlaubt den Zugriff auf den aktuellen Projektzustand und die Begründungen von Entwurfsentscheidungen über das WWW. Durch den Information Assistant kann insbesondere eine Dokumentation des Projektverlaufs inklusive der erstellten Produkte erzeugt werden, auf die über einen herkömmlichen WWW-Browser zugegriffen werden kann. Neben dem generischen Unternehmenswissen über Arbeitsabläufe, welches im Modeler repräsentiert, werden durch den Information Assistant damit projektspezifische Informationen aufbereitet und visualisiert. Im Rahmen unseres Projektes „Intelligenter Bebauungsplan“ bildet er beispielsweise die Basis eines Bürgerinformationssystems.

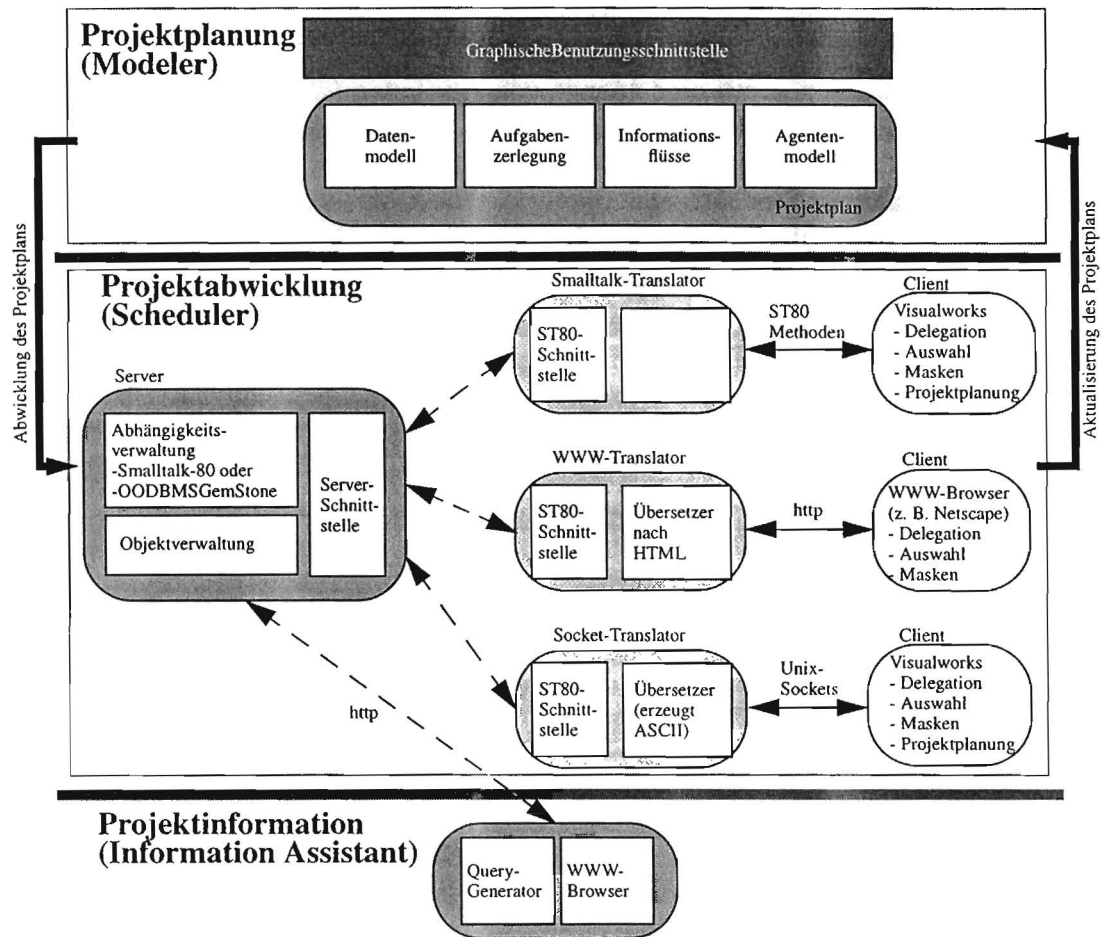


Abbildung 1: CoMo-Kit Architekturübersicht

In den folgenden Abschnitten werden wir auf die Komponenten im einzelnen eingehen. Die Kopplung von Modeler und Scheduler in der Architekturskizze spiegelt unsere Beschäftigung mit Domänen wider, in denen Arbeitsabläufe nur bis zu einem bestimmten Grad planbar sind und eine detailliertere Planung der Aufgaben erst während der Projektabwicklung möglich ist, aufgrund der aktuellen Situation sowie der konkreten, bereits erstellten Produkte (z.B. Anforderungsdokumente in einem Softwareentwicklungsprojekt). Daher sehen wir die Möglichkeit vor, während der Projekt- abwicklung in den Modeler zu wechseln und die benötigten Aufgaben, Aufgabenzerlegungen usw. zu spezifizieren. Wegen der interpretativen Vorgehensweise

des Schedulers können vorgenommene Erweiterungen des Modells bereits für das in Abwicklung befindliche Projekt berücksichtigt werden.

2.1 Wissen über typische Arbeitsabläufe

Für die explizite Modellierung typischer Arbeitsabläufe mit CoMo-Kit stehen vier Hauptkonzepte zur Verfügung: Produkt, Aufgabe, Methode und Agent. Im folgenden gehen wir kurz auf diese Sprachkonzepte ein; eine detaillierte Beschreibung findet sich in [DMMV97].

Produkt: Die Beschreibung der in einem Arbeitsablauf verwendeten oder erzeugten Produkte erfolgt über eine Produktklassenhierarchie, analog zur Klassendefinition in objekt-orientierten Programmiersprachen. In einer Produktklasse wird die Struktur ihrer Instanzen durch eine Menge von Attributen (*slots*) samt Wertebereichen festgelegt. Entlang der Spezialisierungsrelation, die zwischen zwei Produktklassen bestehen kann (Produktverfeinerung), werden Attribute vererbt. Abbildung 2(a) zeigt einen Ausschnitt aus der Produkthierarchie für die Bauleitplanungsdomäne; ein geöffneter Editor für die Produktklasse „Maß der baulichen Nutzung“ ist in Abbildung 2(b) dargestellt.

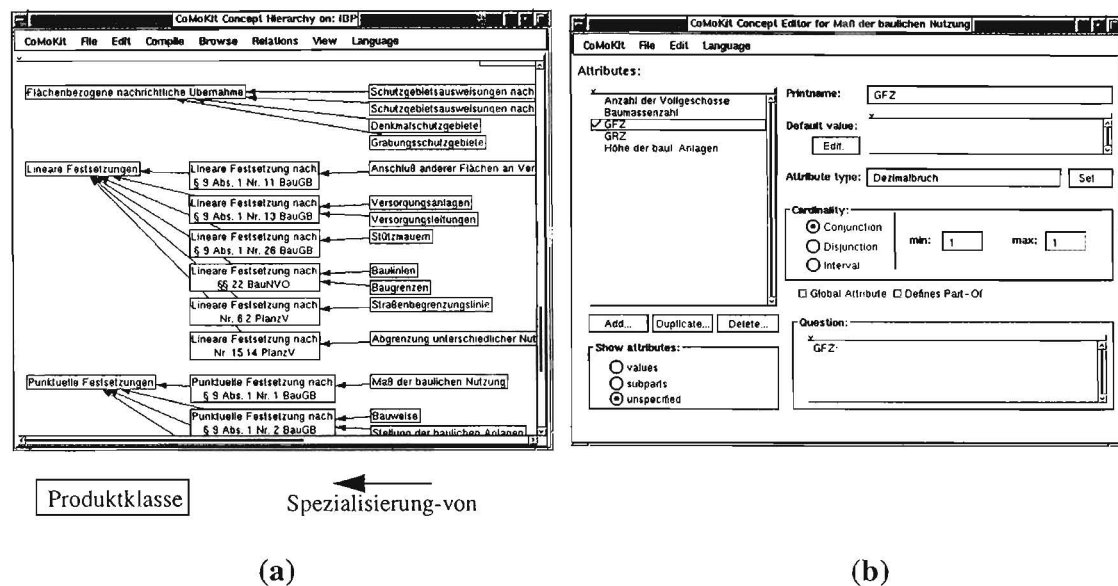


Abbildung 2: (a) Produktklassenhierarchie und (b) Editor für eine Produktklasse

Zusätzlich zu der Definition von Produktklassen, die als Typbeschreibung für während des Projektablaufs zu erstellende Produkte dienen, erlaubt CoMo-Kit die Definition von Produktkonstanten schon in der Modellierungsphase. Diese Konstanten dienen zur Wissensspeicherung von Informationen, die projektübergreifend für den Arbeitsablauf gelten, z.B. Gesetzestexte, die eine Aufgabe in der Bebauungsplanung relevant sind, oder die Beschreibung einer Technik, die im Rahmen des Entwicklungsprozesses einer Software-Firma eingesetzt wird. Die Produktkonstante *Technologiepaket NRL* in Abbildung 3 ist ein Beispiel für eine solche Technik.

Aufgabe: Die Beschreibung einer Aufgabe in einem Projekt enthält neben einer dem Aufgabenziels („Was soll erledigt/getan werden?“) eine Auflistung der Ein- und Ausgabeparameter. Eingabeparameter stehen für Produkte, die zur Bearbeitung der Auf-

gabe notwendige Informationen enthalten; Ausgabeparameter stehen für Produkte, die Resultat der Aufgabenbearbeitung sind. Zusätzlich zu Eingabeparametern, die als Platzhalter für während des Projektablaufs zu erstellende Produkte dienen, können Produktkonstanten als Eingabe für eine Aufgabe definiert werden. Es können Vor- bzw. Nachbedingungen für eine Aufgabe spezifiziert werden, durch die beispielsweise beschrieben werden kann, wann eine Aufgabe bearbeitet werden bzw. eine Aufgabe als erfolgreich abgeschlossen gelten kann. Desweiteren kann durch eine Beschreibung der erforderlichen Qualifikationen und Rollen die Menge der zulässigen Bearbeiter für eine Aufgabe eingeschränkt werden.

Methode: Eine Aufgabe wird durch die Auswahl einer Methode bearbeitet. Durch die Spezifikation verschiedener Methoden können alternative Vorgehensweisen zur Bearbeitung beschrieben werden. Dabei unterscheiden wir zwischen *atomaren* und *komplexen Methoden*. Eine atomare Methode entspricht einer nicht weiter zerlegbaren (oder zumindest nicht näher spezifizierten) Vorgehensweise; ihre Anwendung resultiert in einer Belegung der Ausgabeparameter der entsprechenden Aufgabe. Eine komplexe Methode beschreibt eine Zerlegung der Aufgabe in „kleinere“ Teilaufgaben sowie den zwischen diesen Teilaufgaben bestehenden Produktfluß. Abbildung 3 zeigt einen Ausschnitt aus der Zerlegungshierarchie für Aufgaben aus dem Bereich Softwareentwicklung sowie den Produktfluß zwischen den Teilaufgaben, in die die Aufgabe *Anforderungsbeschreibung erstellen* durch die Methode *NRL/SCR* zerlegt wird.

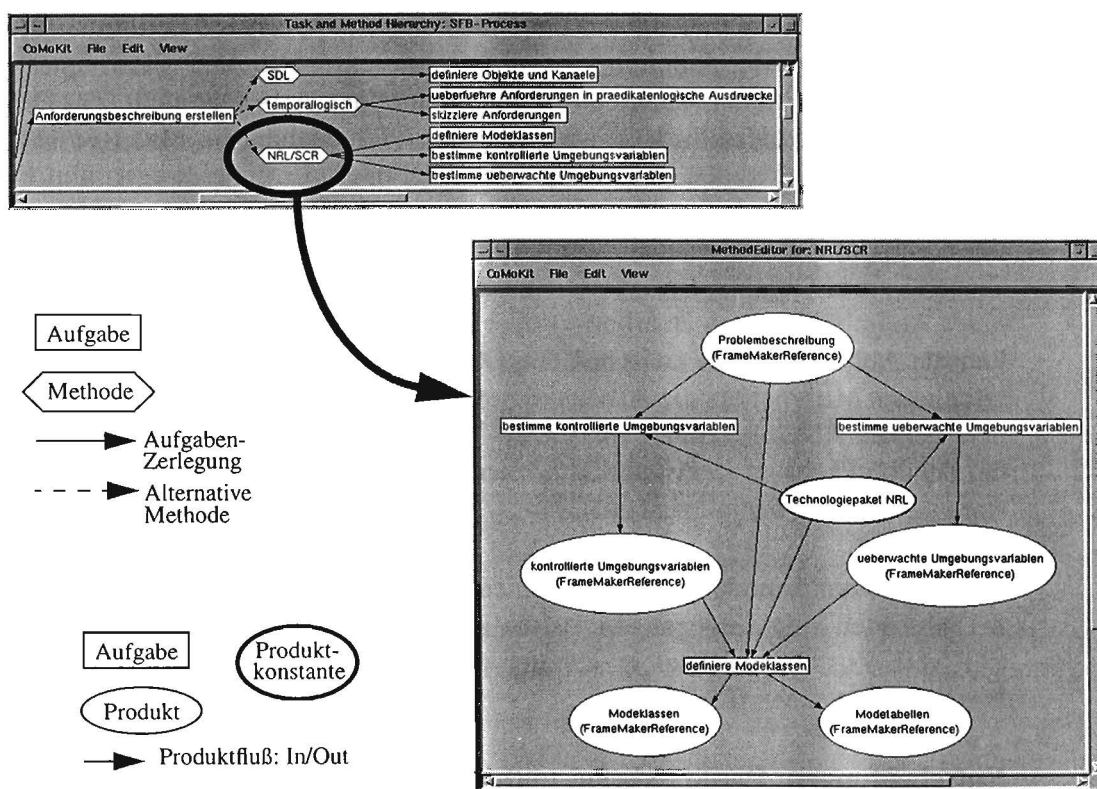


Abbildung 3: Aufgabenzerlegungen (oberes Fenster) und Produktflußgraph

Auch für Methoden lassen sich Vor- und Nachbedingungen spezifizieren sowie Anforderungen an Bearbeiter aufstellen, die die Methode anwenden können sollen.

Agent: Aufgaben werden von Agenten bearbeitet. Als Agenten können in CoMo-Kit sowohl Menschen als auch Computer auftreten. Letzteres ist jedoch nur dann sinnvoll,

wenn die Methoden zur Bearbeitung der Aufgabe ausführbaren Programmcode enthalten. Wie bereits oben erwähnt, können Aufgaben und Methoden mit Anforderungen (Rollen, Qualifikationen und Eigenschaften) assoziiert werden, die von möglichen Bearbeitern erfüllt werden müssen. Entsprechend können für jeden Agenten Rollen Qualifikationen und Eigenschaften beschrieben werden, die dieser besitzt bzw. einnimmt und die mit den Anforderungen während der Projektabwicklung abgeglichen werden.

Die modellierten typischen Arbeitsabläufe bilden den Ausgangspunkt für die Planung eines konkreten Projektes, indem sie grundlegende Aufgaben und alternative Vorgehensweisen zu ihrer Bearbeitung vordefinieren. Bei der Erstellung eines konkreten Projektplanes kann auf diese vordefinierten Elemente zurückgegriffen werden, entweder indem sie für den aktuellen Projektplan ausgewählt oder weiter verfeinert werden. Hierauf werden wir im folgenden Abschnitt weiter eingehen.

2.2 Wissen über in Abwicklung befindliche Projekte

Ein weiterer Teil des Unternehmengedächtnisses dient der Bereitstellung des Arbeitskontextes für die einzelnen Mitarbeiter. Dazu gehören die anstehenden, in Bearbeitung befindlichen und bereits beendeten Aufgaben, die möglichen Lösungswege, bereits getroffene Entscheidungen zusammen mit etwaigen Begründungen, in Erstellung befindliche Produkte und die zur Durchführung einer Aufgabe benötigte Information, wie z.B. Dokumente, Anleitungen und Gesetzestexte. Die Aktivitäten eines Mitarbeiters ändern den Projektzustand und damit auch seinen Arbeitskontext. Zur Dokumentation des Projektzustandes haben wir für die verschiedenen, im Managementwerkzeug verwalteten Objekttypen (d.h. Aufgaben, Methoden, Entscheidungen und Parameter) Zustandsmodelle definiert. Zu jedem Zeitpunkt befindet sich ein Objekt in genau einem der durch das Zustandsmodell seines Typs definierten Zustände. Ereignisse ändern den Zustand eines Objekts. Ereignisse sind Aktivitäten der Benutzer (z.B. das Delegieren einer Aufgabe an einen Bearbeiter) oder Zustandswechsel abhängiger Objekte. Durch kausale Abhängigkeiten zwischen den Objekten eines Projektes können lokale Aktivitäten eines Mitarbeiters die Arbeitskontexte anderer Mitarbeiter verändern. Daher werden kausale Abhängigkeiten von CoMo-Kit verwaltet. Dies ermöglicht darüber hinaus bei der Rückkehr zu einem vorher verworfenen Lösungsweg die automatische Restaurierung des alten Projektzustands.

Auf die Zustandsmodelle der unterschiedliche Objekttypen und deren Bedeutung wird im folgenden kurz eingegangen. Anschließend werden kausale Abhängigkeiten diskutiert, die Zustandswechsel in abhängigen Objekten bewirken. Für eine detaillierte Beschreibung siehe [DMPe97].

Entscheidungen: Die Mitarbeiter treffen im Projektverlauf Entscheidungen, die vom System dokumentiert werden. Dadurch bleiben Lösungswege nachvollziehbar. Wir unterscheiden zwischen *Planungs-*, *Abwicklungs-* und *Delegierungsentscheidungen*. Eine Planungsentscheidung repräsentiert die Auswahl einer Methode. Eine Abwicklungsentscheidung wird bei der Anwendung einer atomaren Methode getroffen. Mit einer Abwicklungsentscheidung werden die Ausgabeparameter einer Aufgabe mit Produkten belegt. Durch Delegierungsentscheidungen werden den Aufgaben verantwortliche Bearbeiter aus dem Pool der qualifizierten Agenten zugeordnet. Eine Entscheidung ist entweder gültig (*valid*) oder ungültig (*invalid*). Der aktuell gül-

tige Projektplan wird durch die Menge aller Entscheidungen im Zustand *valid* spezifiziert. Mit dem Rückzug einer Entscheidung wechselt diese in den Zustand *invalid*.

Aufgaben: Im Zustand *valid* ist eine Aufgabe Teil aktuellen Projektplans. Diesen Zustand verläßt sie, sobald sie nicht mehr Teil des aktuell gültigen Projektplans ist. Sie wechselt dann in den Zustand *invalid*.

Der Zustand *valid* besteht aus verschiedenen Teilzuständen; unter diesen gibt es sowohl Zustände, die den Planungsstatus beschreiben, als auch Zustände, die den Bearbeitungsstatus einer Aufgabe wiedergeben. Eine Aktivität, die den Planungszustand einer Aufgabe ändert, ist z.B. das Treffen einer Planungsentscheidung. (Wechsel von *accepted for planning* nach *performing*). Mit den Zuständen für die Bearbeitung wird der Stand der Ausführung protokolliert. Durch Revision von Entscheidungen können Aufgaben in alte Zustände zurückgesetzt werden. Verwirft der planende Agent beispielsweise den gewählten Lösungsweg, kehrt die Aufgabe von *performing* in *accepted for planning* zurück.

Methoden: Jede Methode repräsentiert einen möglichen Lösungsweg, der im Rahmen des Projektes ausgewählt (*selected*) oder zurückgewiesen (*retracted*) werden kann. Eine ausgewählte Methode durchläuft verschiedene Teilzustände, die den Abarbeitungsstand der Methode reflektieren. Bei einer komplexen Methode muß z.B. die Bearbeitung der Teilaufgaben überwacht werden, bei einer atomaren Methode der Fortschritt bei der Erstellung der Produkte.

Parameter: Zu Beginn eines Projektes ist ein Parameter unbelegt (*unassigned*). Mit der Zuweisung eines Produktes wechselt dieser in den Zustand *assigned*. Durch den Rückzug von Entscheidungen kann eine Parameterbelegung ungültig werden. Dies wird durch den Rücksprung nach *unassigned* repräsentiert.

Zwei Arten von Ereignissen können Zustandswechsel in einem Objekt auslösen: Aktivitäten der Benutzer (z.B. das Treffen einer Entscheidung) und Zustandswechsel von Objekten, die in kausalem Zusammenhang zu dem betrachteten Objekt stehen. Kausale Abhängigkeiten werden automatisch aus dem Modell der Arbeitsabläufe abgeleitet und im Scheduler verwaltet. Da die abgeleiteten Abhängigkeiten über- oder unterspezifiziert sein können, kann der Benutzer diese erweitern oder einschränken. Im folgenden wird ein grober Überblick über die verwalteten Abhängigkeiten gegeben. In [DKM96] findet sich eine detaillierte Beschreibung.

Abhängigkeiten im Produktfluß: Bei der Operationalisierung der Pläne werden zwei Typen von Abhängigkeiten etabliert: (i) Abhängigkeiten zwischen den von einer Aufgabe konsumierten Produkten und der für sie getroffenen Planungsentscheidung (ii) Abhängigkeiten zwischen einer Abwicklungsentscheidung und der durch sie erzeugten Produkte.

Beim ersten Abhängigkeitstyp gehen wir davon aus, daß die zu treffende Entscheidung vom aktuellen Kontext abhängt. Dies bedeutet, daß die Entscheidung für die Wahl einer Methode von den konsumierten Produkten abhängig ist. Ändern sich die Eingabeprodukte, muß die Entscheidung überdacht und gegebenenfalls revidiert werden.

Da durch eine Abwicklungsentscheidung die Ausgabeparameter der bearbeiteten Aufgabe belegt werden, hängt die Gültigkeit dieser Belegung von der Gültigkeit der

Abwicklungsentscheidung ab. Mit der Revidierung der Entscheidung wird diese Belegung rückgängig gemacht. Das Produkt und die etablierte Abhängigkeit werden jedoch nicht gelöscht. Kehrt der Benutzer später zu dieser Entscheidung zurück, werden die alten Belegungen automatisch wieder gültig.

Abhängigkeiten zwischen einer Aufgabe und ihren Teilaufgaben: Mit der Entscheidung für eine komplexe Methode legt der Planer fest, welche Teilaufgaben auszuführen sind. Diese bleiben nur so lange gültig, wie die komplexe Methode Teil des aktuell gültigen Plans ist.

Delegierung: Bei der Delegierung wird die Bindung eines Agenten an eine Aufgabe von der Delegierungsentscheidung abhängig gemacht. Beim Rückzug der Entscheidung wird diese Bindung wieder aufgelöst. Dadurch schafft man die Voraussetzung, Agentenbindungen bei geänderten Rahmenbedingungen (z. B. Krankheit eines Agenten) modifizieren zu können.

Produkte und Teilprodukte: Parameterhierarchien spiegeln die hierarchische Struktur von Produkten wider. Die Belegung eines Parameters setzt sich daher aus den Belegungen seiner Teile zusammen.

Benutzerdefinierte Abhängigkeiten: Die bisher aufgeführten Abhängigkeiten werden automatisch aus dem Projektplan generiert. In der Praxis zeigt sich, daß Projektpläne unter Umständen nicht alle Abhängigkeiten vorgeben. Ein detaillierter Projektplan ist an vielen Stellen von den Ergebnissen vorangegangener Schritte der Projektdurchführung abhängig. Wurde zum Beispiel bei der Entwicklung einer bestimmten Komponente eine vorgegebene Anforderung aus dem Anforderungsdokument berücksichtigt, so ist diese Abhängigkeit bei der Erstellung des allgemeinen Arbeitsablaufes noch nicht bekannt, sondern kann erst bei der Abwicklung formuliert werden. Eine solche Abhängigkeit bezeichnen wir als „Design Rationale“. Wir haben Techniken entwickelt, Design Rationales während der Abwicklung formulieren und durch den Scheduler verwalten lassen zu können. Für eine detaillierte Darstellung siehe [DKM96].

Technisch umgesetzt wurden die Zustandsmodelle in Anlehnung an REDUX [Pet91]. Durch den Einsatz eines Truth Maintenance Systems (*TMS*) [Doy79] werden Zustandswechsel von Objekten entlang der etablierten Abhängigkeiten effizient propagiert.

2.2.1 Projektentwicklung - Ein Beispiel

Das folgende Szenario zeigt Ausschnitte aus der Abwicklung eines Projekts aus Bearbeitersicht.

Abbildung 4 zeigt die Arbeitsliste der Bearbeiterin „Barbara Dellen“. Die Bearbeiterin kann zwischen verschiedenen Sichten auf den Arbeitskontext auswählen. Über das Symbol „Ablagekästchen“ kann sie sich die zur Abarbeitung bereitstehenden Aufgaben, über „Zettel“ die in Abarbeitung befindlichen Aufgaben anzeigen lassen. Die unter dem Symbol „Ablagekästchen“ liegenden Aufgaben stehen zur Bearbeitung an und können von ihr ausgeführt werden. In Abbildung 4 ist das „Ablagekästchen“ selektiert.

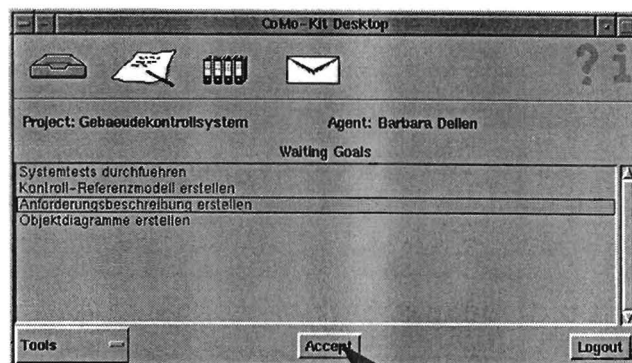


Abbildung 4: Arbeitsliste einer Bearbeiterin

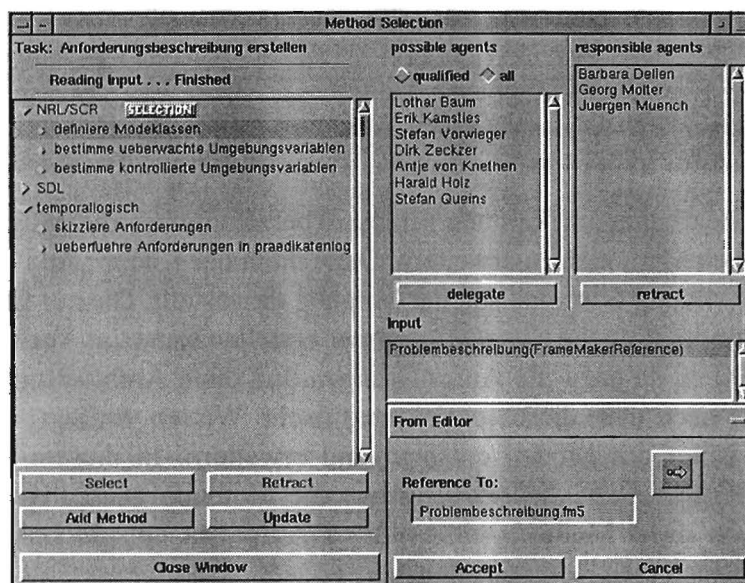



Abbildung 5: Fenster zur Methodenselektion

In diesem Szenario entscheidet sich die Bearbeiterin für die Aufgabe *Anforderungsbeschreibung erstellen*, indem sie den Knopf „Accept“ drückt. Daraufhin wird sie angewiesen, eine Methode zur Erstellung der Anforderungsbeschreibung auszuwählen. Dazu werden ihr in einem Fenster (siehe Abbildung 5) die dem System derzeit bekannten Lösungswege angeboten (vergleiche auch Abbildung 3): *NRL/SCR*, *SDL* oder *temporallo-*

gisch. Die jeweils von einer Methode eingeführten Teilaufgaben sind eingerückt dargestellt. Auf die aktuellen Eingabeprodukte kann im rechten unteren Teil des Fenster zugegriffen werden. Im Beispiel wird der Bearbeiterin die Referenz auf die Problembeschreibung (ein Framemaker-Dokument) angezeigt. Über den Knopf  kann das Dokument geöffnet werden. Die von der Bearbeiterin ausgewählte Methode ist durch **SELECTION** markiert. Im rechten oberen Teil des Fensters kann die Bearbeiterin die einzelnen Teilaufgaben delegieren. Im Beispiel ist die Aufgabe *definiere Modeklassen* an *Barbara Dellen*, *Georg Molter* und *Juergen Muench* delegiert worden. Die Aufgabe erscheint nun automatisch auf den Arbeitslisten dieser BearbeiterInnen. Das Erstellen von Produkten ist nicht dargestellt: entscheidet sich die Bearbeiterin für eine atomare Methode (falls möglich), werden dem Produkttyp entsprechende Editoren geöffnet.

Jeder Bearbeiter hat die Möglichkeit, seine Entscheidungen zurückzunehmen. Im Beispiel könnte die Bearbeiterin durch Drücken des Knopfes „Retract“ (Abbildung 5) die Entscheidung für die Methode *NRL/SCR* oder Delegierungsentscheidungen zurücknehmen. Insbesondere wenn das Projekt schon weit fortgeschritten ist, können derartige Aktivitäten globale Auswirkungen auf den Projektzustand haben.

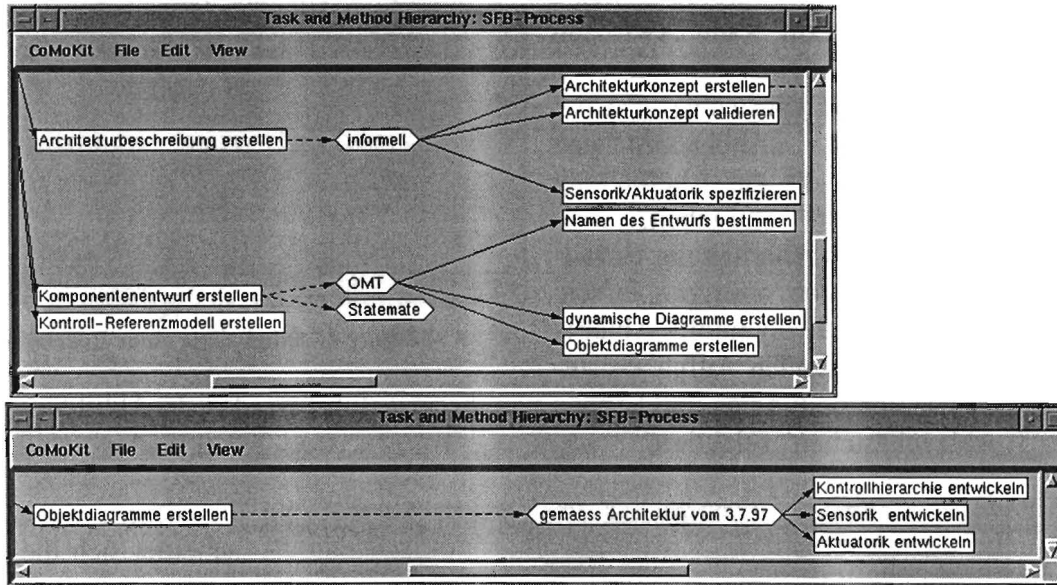


Abbildung 6: Detailplanung während der Projektabwicklung

Desweiteren hat der Bearbeiter im Rahmen seines Arbeitskontextes die Möglichkeit, die Liste der alternativen Methoden zu erweitern. Dazu dient der Knopf „Add Method“ in Abbildung 5. In Abbildung 6 ist ein solcher Vorgang dargestellt. Die zur Entwurfsmethode *OMT* gehörende Aufgabe *Objektdiagramm erstellen* kann im Vorfeld nicht feiner spezifiziert werden, da dazu die Anforderungen und erste Architekturkonzepte benötigt würden. Erst nach dem dieses projektspezifische Wissen vorliegt, kann der verantwortliche Bearbeiter das Modell entsprechend erweitern. In diesem Szenario definiert er für das Projekt *Gebäudekontrollsystem* die Methode *gemäß Architektur vom 3.7.97*. Die durch diese Methode eingeführten Aufgaben entsprechen den im Architekturkonzept festgelegten Komponenten.

2.3 Wissen über frühere Projekte

Mit Hilfe des Information Assistant kann automatisch ein Netz von WWW-Seiten erzeugt werden, die den Endzustand eines Projektplanes dokumentieren sowie Zugriff auf die bei der Abwicklung erstellten Produkte gewähren. Dadurch besteht für Bearbeiter in einem aktuellen Projekt die Möglichkeit, auf die in den früheren Projekten enthaltenen Erfahrungen (etwa in Form der Design Rationales damals getroffener Entscheidungen) zurückzugreifen. Dies stellt einen ersten Schritt in Richtung einer „Experience Factory“ [Bas93] dar, wobei vorerst lediglich ein navigierender Zugriff auf für einen Bearbeiter relevante Informationen aus früheren Projekten möglich ist. Abbildung 7 zeigt zwei durch den Information Assistant erzeugte WWW-Seiten aus der Dokumentation des in Abschnitt 2.2.1 skizzierten Projektes zur Entwicklung eines Gebäudekontrollsystems.

Das Fenster links oben liefert Zustandsinformationen über die bearbeitete Aufgabe *Anforderungsbeschreibung erstellen*; das Fenster rechts unten zeigt die bereits aus Abbildung 3 bekannte Aufgabenzerlegung in Form einer maussensitiven Abbildung, die den Zugriff auf die dargestellten Objekte erlaubt.

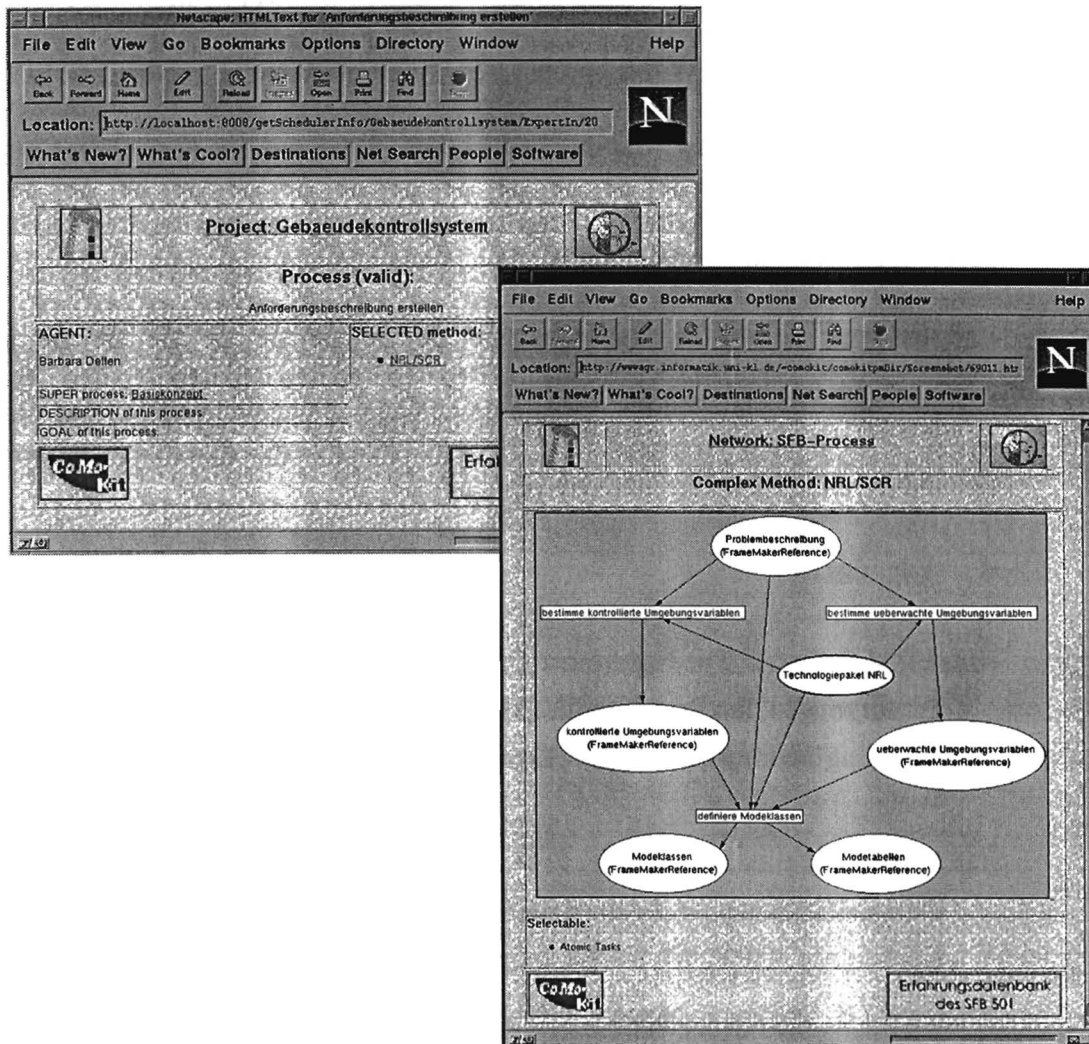


Abbildung 7: Durch den Information Assistant erzeugte Projektdokumentation

3 Zusammenfassung und Ausblick

Spätestens mit der Einführung der ISO 9001 wurde der Einfluß des Entwicklungsprozesses auf die Produktqualität anerkannt. Voraussetzung für die Verbesserung des Entwicklungsprozesses ist die Dokumentation, Kontrolle und Koordination der Arbeitsschritte. Das Projektmanagementwerkzeug CoMo-Kit unterstützt Unternehmen bei dieser Aufgabe. Es erlaubt die Modellierung von Arbeitsabläufen und koordiniert deren Abwicklung. Durch den Einsatz wissensbasierter Techniken wird die starre Trennung zwischen Modellierung und Durchführung aufgehoben und Änderungsprozesse „on the fly“ unterstützt. Da Planungsentscheidungen zur Projektlaufzeit getroffen werden können, kann die Gefahr von Fehlentscheidungen reduziert werden. Durch die Anbindung an das WWW kann bei der Abarbeitung der Arbeitsschritte auf extern ver-

fügbare Information, wie Vorschriften, Anweisungen und Normen zur Qualitätssicherung und Sicherheitsgewährleistung zugegriffen werden.

Um nicht der Vielzahl von Modellierungssprachen beliebig Vorschub zu leisten, arbeiten wir zur Zeit in Kooperation mit der hiesigen Arbeitsgruppe „Software Engineering“ an einer Integration der dort zur Modellierung von Software-Entwicklungsprozessen entworfenen Sprache MVP-L und der von uns verwendeten Sprache [DMMV97].

Neben der Verteilung der Arbeitsschritte beschäftigen wir uns desweiteren mit der Verteilung der Workflowmaschine (via Local und Wide Area Networks). Die Idee ist, die entwickelten Techniken mit agentenorientierten Ansätzen, wie sie z. B. in [Pet93] vorgeschlagen werden, zu verknüpfen.

Langfristiges Ziel ist die Einbindung von Mechanismen zur Wiederverwendung von Projektplänen. Dazu müssen Pläne in einer „Experience Factory“ [Bas93] verwaltet und effiziente Retrieval- und Adaptionstechniken integriert werden. Auf diese Weise würde eine weitere Voraussetzung für eine langfristige Qualitätsverbesserung von Geschäfts- und Entwurfsprozessen geschaffen.

4 Literatur

- [Bas93] V. R. Basili: The Experience Factory and its Relationship to Other Improvement Paradigms, in: Ian Sommerville, Manfred Paul, eds., Proc. of the 4th European Software Engineering Conference, Lecture Notes in Computer Science Nr. 706, Springer Verlag, 1993, 68-83
- [DKM96] B. Dellen, K. Kohler, F. Maurer: Integrating Software Process Models and Design Rationales, in: Proc. of the Eleventh Knowledge Bases Software Engineering Conference KBSE 96: IEEE Computer Society, U. S., 84- 93
- [DMPa95] B. Dellen, F. Maurer, J. Paulokat: Verwaltung von Abhängigkeiten in kooperativen, wissensbasierten Arbeitsabläufen, in: M. M. Richter & F. Maurer (Hrsg.), Expertensysteme 95: Proceedings in Artificial Intelligence, infix, Sankt Augustin, 1995.
- [DMMV97] B. Dellen, F. Maurer, J. Münch, M. Verlage: Enriching Software Process Support by Knowledge-based Techniques: erscheint in International Journal on Software Engineering and Knowledge Engineering, 1997
- [DMPe97] B. Dellen, F. Maurer, G. Pews: Knowledge Based Techniques to Increase the Flexibility of Workflow Management: to appear Data & Knowledge Engineering Journal, 1997
- [Doy79] Doyle, J.: A Truth Maintenance System: Artificial Intelligence, 1979
- [Pet91] Ch. Petrie: Planning and Replanning with Reason Maintenance, Dissertation, University of Texas, Austin, 1991
- [Pet93] Ch. Petrie: The Redux' Server, Proc. ICICIS, Rotterdam, 1993

The Data Warehouse as a Means to Support Knowledge Management

Michael Erdmann

Institut für Angewandte Informatik und Formale Beschreibungsverfahren
University of Karlsruhe (TH)
D-76128 Karlsruhe (Germany)
e-mail: erdmann@aifb.uni-karlsruhe.de

Abstract: This paper tries to provide a new view on the currently vastly discussed and successfully employed concept of a Data Warehouse. This view presents it in the light of Knowledge Management, i.e. a Data Warehouse can serve as a storage medium for keeping the corporate memory, or at least concerning certain types of data. It helps gaining new knowledge by delivering well integrated data to analysis tools, e.g. On-Line Analytical Processing or Knowledge Discovery in Databases, and thus becomes an important part of Decision Support Systems or Executive Information Systems. In this way a Data Warehouse, storing only data, results in growth of knowledge and may lead to enhance the enterprise's success.

The paper does not claim, that a Data Warehouse is the only thing an enterprise needs to perform successful Knowledge Management.

1 Introduction

During the last months several workshops, symposia etc. dealt with a new (or not so new) topic: "Knowledge Management" (KM). The term seems to embrace several existing research areas, which are all tied together by their common application environment, namely the enterprise. Some topics gathered under the new label are workflow management, business process modelling, document management, data bases and information systems, knowledge based systems, and several methodologies to model diverse aspects relevant when dealing with knowledge —or the like— in an enterprise environment.

One key term when discussing knowledge management became the "Corporate Memory" or "Organizational Memory". This memory serves for storing the enterprise knowledge which has to be managed. Analogous to the diverse approaches summoned together as knowledge management the corporate memory also contains several kinds of information, e.g. know-how in the heads of employees; case-knowledge, such as lessons learned; atomic, raw, or low level data, such as lists of customers, suppliers, or products, which are stored in data bases; or several documents stored as natural language texts in files. [Kühn, Abecker 97] define a corporate memory as "an enterprise-internal application-independent information and assistant system [which ...] stores large amounts of data, information, and knowledge from different sources of an enterprise."

In this paper we will show how a Data Warehouse (DWh) smoothly matches this definition and thus should be considered during KM decision processes. Although the "D" in DWh suggests that only data is stored in a DWh, this data can become valuable knowledge for the enterprise by analysing the large amounts of data with Knowledge Discovery (KDD) or On-Line Analytical Processing (OLAP) mechanisms.

Because we think "knowledge managers" should be aware of some differences between data, information, and knowledge we will try to define these three terms in section 2, although we will not back up these definitions with a comprehensive philosophical discussion. The next section then, will present the fundamental principles underlying a DWh and its contribution for knowledge mining through data analyses. In section 4 the DWh is related to KM without assuming that a DWh may solve every problem arising whilst KM processes and without presenting it as *the ultimate KM system*.

2 Data, Information, and Knowledge

In this section we will present three terms widely —but often unreflectingly— used in several IT-related (and other) communities, i.e. 'data', 'information', and 'knowledge'. The definitions will be oriented according to the three dimensions of semiotics (the theory of signs), i.e. syntax, semantics, and pragmatics [Morris 71].

[Aamodt, Nygard 95] state "there is, in general, no known way to distinguish knowledge from information or data on a purely representational basis." As we see it, this is due to the fact, that any representation is restricted to using signs (e.g. ASCII-characters, bits, or handwriting), thus there simply cannot be any distinctions. It is only through relations, that signs or representations can be separated into data, information, or knowledge. Signs can be interpreted along three dimensions. (1) The relation among signs, i.e. the syntax of 'sentences' does not relate signs to anything in the real world and thus, can be called one-dimensional. In our eyes, signs only viewed under this dimension equal data. (2) The relation between signs and their meaning, i.e. their semantics adds a second dimension to signs. It is only through this relation between sign and meaning, that data becomes information. (3) The third dimension is added by relating signs and their 'users'. If this third dimension including users, who pursue goals which require performing actions, is introduced, we call the patterns, or signs knowledge. This relationship, which is called pragmatics, defines an important characteristic of knowledge, i.e. only knowledge enables actions and decisions, performed by actors.

To illustrate these distinctions we will give an example: *What does the sign "25" mean?* Because we can only perceive the syntactical dimension it is nonsense to ask for the meaning of this **data**. After adding a relation between the sign "25" and the real world concept of "25 meters", we can assign a meaning to the given pattern; we yielded a bit of **information** but we do not know what to imply from this information. The information does not induce or suggest any actions. So, we can ask *What does this information mean to us or any other person?* Assuming, that the sign "25" is shown on the display of an instrument indicating the distance of a landing plane from the floor underneath this information must be interpreted by the pilot in an appropriate way. His **knowledge** then may lead to certain actions to successfully finish the landing manoeuvre. As we see, knowledge —on the representational level— does not differ from data, but provided a concrete context and more knowledge to interpret raw data it makes actions possible.

Transferring these semiotically motivated definitions into the area of knowledge management, there can be seen plain analogies. It does not matter whether patterns were represented in data bases, information systems, knowledge based systems, or any other (computer) systems; they are all alike, i.e. they are all represented by signs. It is only through usage of these signs, including their various roles, contexts, and users, that they become data, information, or knowledge. In [Aamodt, Nygard 95] this kind of distinction is called *frame of reference* and states who uses patterns in what way, e.g. patterns stored in an information system are used (i.e. interpreted) by human users, whereas domain models of knowledge based systems (KBS) are used by (automated) problem solving methods (PSM) for inferencing. Thus, in the first case the user of the IS knows or learns something, whereas in the latter case one could claim, the KBS contains knowledge.

According to the three semiotical dimension identified for signs, a pattern (as data) has to be interpreted to yield information, i.e. data with meaning. This interpretation requires knowledge, i.e. knowledge has to take an active role during the interpretation process. Thus, we can further distinguish knowledge from data and information. Data, information, and knowledge embody passive objects which have to be handled within knowledge management. Knowledge alone has the capability to support knowledge management actively. Knowledge, or its owners/users are the subjects capable of acting. The enabled actions can be manifold, e.g. as we have seen processing, interpreting, and understanding data and information; learning, i.e. gaining new knowledge; or any external actions, such as selling stocks, cancelling a project, or rating the credit-worthiness of customers etc.

In the next section we will present the main features of a Data Warehouse and show how the stored DWh data can support effective actions through data analyses.

3 The Data Warehouse

This section will describe some basic concepts of Data warehouses. The term Data Warehouse (DWh) has been defined by Bill Inmon —the *father of Data Warehousing*— as follows [Inmon 96]:

"A Data Warehouse is a subject-oriented, integrated, time-variant, and non-volatile collection of data in support of management's decision-making process."

This definition reflects the main purpose, a DWh has to support. It contains data and delivers it to executives as knowledge, they can built their decisions upon. The four named adjectives characterizing a DWh distinguish DWhs as informational systems from so called operational systems.

- A DWh is *subject-oriented* because the data it contains is structured in a way reflecting the business objects of the company (e.g. products, clients, sales). Operational systems on the other hand tend to be "organized around the applications of the company" [Inmon 96], e.g. databases handling all data relevant for booking passengers for flights. This system contains several subjects; a fact which complicates data analyses. The subject-orientation, on the other hand especially supports analytical tasks (see below) and thus, the production of knowledge.

- The second aspect, the *integration*, is the main characteristic of a DWh. A DWh contains data stemming from several sources (i.e. operational systems) which are spread all over the enterprise. These heterogeneous sources have to be integrated to access data in a uniform and clear way, i.e. all data has to be represented in an integrated way. Integration means, all data that is loaded into the DWh is transformed into a unique representation, e.g. no matter how the gender of persons is represented in several operational (source) systems (e.g. male/female, m/f, 0/1, X/Y etc.), one representation is selected and all others are transformed into this unique one. Integration of heterogeneous data sources has been investigated for some time in the IT-area [Saltor et al. 93], esp. since the growing importance of the internet and its numerous information sources. Only by defining an integrated representation analytical processing in the large amounts of data stored in a DWh becomes possible.
- A DWh is a *time-variant* collection of data, i.e. it contains current data as well as historic data. Due to that analytical processing can be done along the time dimension, thus trends and developments can be identified concerning the subjects of the enterprise, e.g. the development of sales of several products in several regions may be compared for the last twelve months. In contrast, operational systems only contain up-to-date data, thus no trends are recognizable within such a system. The DWh contains a sequence of snapshots taken periodically from operational level data.
- *Nonvolatility* of a DWh means, everything put into a DWh remains there in one way or another. Operational systems are highly volatile, i.e. records are frequently added, accessed, updated, or deleted. These read and write accesses require special mechanisms to prevent deadlocks, to prevent loss of information, and to ensure consistency. A DWh is essentially accessed read-only with the exception of loading new data into the DWh by taking snapshots at well defined points in time. This read-only access is due to the purpose to support analytical needs in "management's decision-making".

A DWh is organized within at least two orthogonal dimensions, a dimension of time (see above) and a granularity dimension. Data loaded into the DWh from an operational system enters as up-to-date, detailed data (see figure 1). All detailed data can be aggregated under several criteria to yield lightly summarized data. These summaries can further be aggregated to yield highly summarized data, etc. E.g. daily sales could be stored at the detailed level (i.e. one snapshot of sales data is taken each day), the lightly summarized data represents weekly and the highly summarized data represents monthly aggregation. Thus several levels of granularity are stored in a DWh, although this produces some redundancy. Because of the enormous amounts of data stored in a DWh some analytical tasks only are computable within an acceptable time, if some required data is pre-aggregated. Since all data remains in the DWh it ages with time and simultaneously its importance and the chances of accessing decrease. The time horizon for a DWh (normally 5 to 10 years) is significantly longer than that for operational systems (normally 60 to 90 days) [Inmon 96]. Despite the data's age it actually *may* be accessed in the future, so it stays in the DWh but moves to external (slower but cheaper) storage media, e.g. optical disks, tapes, or micro fiches, while the more interesting data is stored on direct access storage devices, e.g. hard disk. Even data stored in these external media is considered part of the DWh, because these data can be accessed for analyses, if needed.

Besides raw and aggregated data a DWh contains metadata describing its contents, the sources of data, and the transformation procedures converting raw data into aggregated

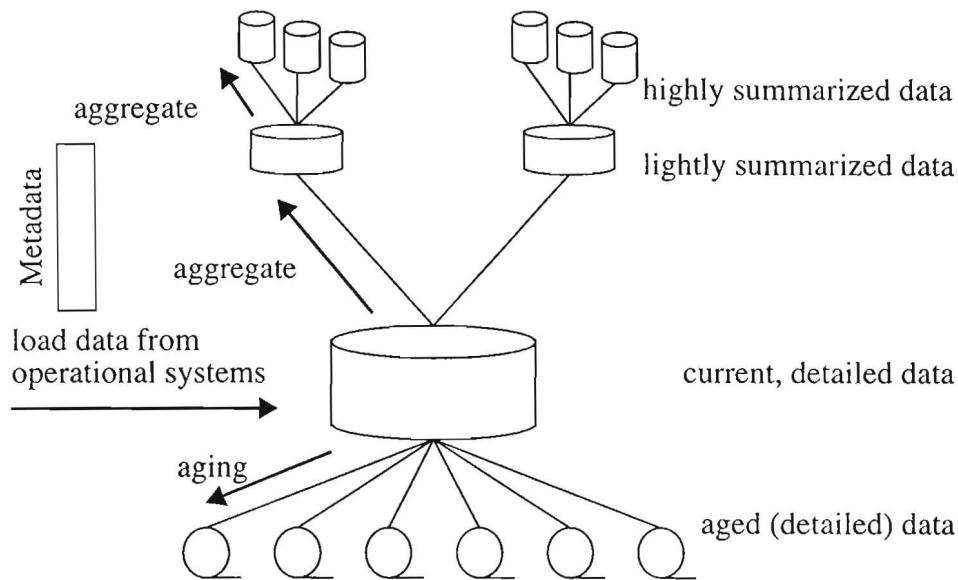


Figure 1 Structure of a Data Warehouse

data or source data into integrated, cleansed data. Metadata also serves as a navigation aid for the DWh-users, i.e. the data analysts. The analysts will consult metadata when planning data analyses.

The DWh has been defined as a "collection of data" with the goal to support "decision making processes. Essentially the DWh contains several kinds of data which are accessed through analysis front ends, such as OLAP tools or KDD workbenches, i.e. the DWh provides data for analyses which then support decision making. The possibilities provided by data analyses will be presented in the next section as one contribution of Data Warehousing to knowledge management.

4 Data Warehouse and Knowledge Management

After stating what a DWh looks like, we will point out in which way the DWh could contribute to a company wide knowledge management. In fact, a DWh could serve as one main component in a knowledge management system. The data contained in a DWh represents a large part of a company's knowledge, e.g. the company's clients and their demographic attributes. The DWh represents an enterprise wide data collection, which is central and defines a common basis for several enterprise units accessing it. From the stored data new knowledge can be derived using technologies such as On-Line Analytical Processing (OLAP) or Knowledge Discovery in Databases (KDD).

Data analyses may consist of several reporting and visualisation mechanisms of the data, presented on different levels of aggregation, from different angles (i.e. dimensions), and using different graphical types of diagrams. These reporting facilities can be exploited interactively using OLAP-technology. Through OLAP the data analyst is enabled to formulate queries and to decide on further queries depending on the outcome of his former queries. In this way, the analyst wanders through the DWh collecting information, which he presents to the management. Recalling the definitions of data, information, and knowledge, we can recognize a similar schema. Data is stored in the DWh. The data ana-

lyst interprets parts of the data, which is represented in a way more adequate for human users. The process of interpreting data needs some knowledge and if the yielded information leads to decisions or actions performed by the management this information becomes knowledge.

Another way of gaining knowledge out of the DWh's data are algorithms provided by Knowledge Discovery in Databases (KDD). These mostly mathematical and statistical methods are able to detect knowledge previously unknown to the owners of the data. [Fayyad et al. 96] define KDD as follows:

"Knowledge Discovery in Databases (KDD) is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data."

To be able to gain valid and useful patterns out of data, it is necessary for the underlying database to contain as less noise as possible. A DWh guarantees —through its integration mechanisms— that all data is correct, so that data mining algorithms will work properly. On the solutions produced by these algorithms the management may found its decisions upon.

These technologies —KDD and OLAP— represent core mechanisms exploited within Decision Support Systems (DSS) or Management and Executive Information Systems (MIS, EIS). It is through these systems, that managements decisions are based on assured, enterprise wide, real data.

Data analyses influence or yet enable management's decisions. As an example we will present a clothing manufacturer who employs a DWh, basing on an SQL database and tools for perform OLAP. The manufacturer provides several stores across the USA with clothes in different models, in several sizes, and several colours. The first success story of the employment of the DWh and the OLAP tools relates to the number and designs of clothes delivered to the stores. The company's goal is to avoid to deliver too less as well as too much units of clothes with specific designs, sizes, and colours to each individual store, because both would result in an decreasing income, because some clothes cannot be sold, and some which could be sold, were not in stock. After establishing the DWh, a simple OLAP analysis delivered that clothes of a certain colour are sold ten times more often in Miami than in New York. Before using a DWh no queries of this type could be asked, so that a turnover of at least 30% has been lost. After gaining this new knowledge the company now can better fulfil their goals.

5 Conclusion

Successful knowledge management needs to integrate data bases, information systems, and knowledge based systems. As we have presented, a DWh can connect these three kinds of systems. It provides a wide basis of integrated data; this data can be presented via Management or Executive Information Systems (MIS, EIS). It could be interpreted as knowledge if analysis algorithms discover currently unknown patterns in the large amounts of DWh data. Newly derived knowledge or visualized information may be incorporated into the management's decision making process.

The DWh and several other more technical points —naturally discussed in the CS and AI communities— only represent one aspect of knowledge management. [Sierhuis, Clancy

97] write "knowledge management is not just about modelling problem solving and expert knowledge [or the like]. Knowledge Modelling is also about modelling the dynamics, social and cognitive, of a human activity system", i.e. the people in an enterprise must not be forgotten. In KM they play the central role as *carriers* of knowledge.

Concerning the DWh, this means that the DWh must be complemented by several other technologies and ways of working to yield successful knowledge management, i.e. a DWh is not *the ultimate solution*. Yet, there seems not to exist such an ultimate approach to knowledge management due to its immense wideness.

6 References

- [AAAI 97] **AAAI-97 Spring Symposium: Artificial Intelligence in Knowledge Management.** Working Notes. Stanford University, Stanford, California, March 1997.
- [Aamodt, Nygard 95] **A. Aamodt, M. Nygard:** Different Roles and Mutual Dependencies of Data, Information, and Knowledge - An AI Perspective on their Integration. in: *Data & Knowledge Engineering* 16 (1995). Elsevier, North-Holland 1995. pp.191-222
- [Abecker et al. 97] **A. Abecker, A. Bernardi, K. Hinkelmann, O. Kühn, M. Sintek:** *Towards a Well-Founded Technology for Organizational Memories.* in [AAAI 97] pp 1-7
- [Chaudhuri, Dayal 97] **S. Chaudhuri, U. Dayal:** An Overview of Data Warehousing and OLAP Technology. in: *SIGMOD Record*, Vol. 26, No. 1, March 97. pp. 65-74
- [Fayyad et al. 96] **U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurasamy:** *Advances in Knowledge Discovery and Data Mining.* MIT Press, Cambridge 1996.
- [Inmon 96] **W.H. Inmon:** *Building the Data Warehouse.* 2nd edition. John Wiley & Sons, Inc., New York 1996.
- [Inmon et al. 97] **W.H. Inmon, J.D. Welch, K.L. Glassey:** *Managing the Data Warehouse.* John Wiley & Sons, New York 1997.
- [Inmon, Hackathorn 94] **W.H. Inmon, R.D. Hackathorn:** *Using the Data Warehouse.* John Wiley & Sons, New York 1994.
- [Kühn, Abecker 97] **O. Kühn, A. Abecker:** Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges. submitted to: *Journal of Universal Computer Science (JUCS)*, 1997.
- [Morris 71] **Ch. W. Morris:** *Writings on the General Theory of Sign.* Den Haag 1971
- [Saltor et al. 93] **F. Saltor, M.G. Castellanos, M. Garcia-Solaco:** Overcoming Schematic Discrepancies in Interoperable Databases. in: **D.K. Hsiao, E.J. Neuhold, R. Sacks-Davis (eds.):** *Proceedings of the IFIP WG2.6 Database Semantics Conference on Interoperable Databases.* Elsevier, North-Holland, 1993. S.191-205.
- [Sierhuis, Clancy 97] **M. Sierhuis, W.J. Clancey:** Knowledge, Practice, Activities, and People. in [AAAI 97] pp. 142-148

[Wu, Buchmann 96] **M.-C. Wu, A.P. Buchmann**: Research Issues in Data Warehousing. in: *Proceedings of the BTW 1996*.

Multimediale Wissensräume - Werkzeuge für Aufbau, Wartung und Nutzung eines gemeinschaftlichen Unternehmens-Gedächtnisses

Hans-Jürgen Frank
Soxhletstr. 6, D-80805 München, Germany
Phone ++ 49 89 363156 Fax ++49 89 362848

Outline of the Talk:

- Die Bedeutung der Darstellung von Wissenselementen und Wissenszusammenhängen in Bildern und räumlichen Strukturen
- Vorführung multimedialer Wissensräume, Beispiele aus Industrie, Gesellschaft, Politik
- Themen und Erfahrungen aus der praktischen Projekt-Arbeit:
 - Komplexitäts-Bewältigung in Problemlöse- und Entscheidungsfindungs-Prozessen
 - Entwicklung von Pattern der wenigen wirklich für ein Projekt oder einen Prozeß wichtigen Kernpunkte aus einer Masse von Daten
 - Wissensräume und ihre Möglichkeiten gedankliche und emotionale Potentiale zu verknüpfen
 - Multimediale Wissensräume als Dialog-Instrument für die gemeinschaftliche Nutzur von Know-How
 - Synergiepotentiale zwischen Personen und Wissenselementen
 - Einrichtung persönlicher Wissensräume als individuelle Spur in einem gemeinschaftlichen Unternehmens-Gedächtnis
 - Katalytische Begleitung von Prozessen der Strategie-Entwicklung
- Zukunftsperspektiven räumlicher Wissensdarstellung

Knowledge Management in Global Health Research Planning

C. Greiner & T. Rose

Research Institute for Applied Knowledge Processing (FAW), Helmholtzstr. 16, 89081 Ulm, Germany
Phone ++49 731/ 501 - 950 Fax ++49 731/ 501 - 999, E-mail {greiner,rose}@faw.uni-ulm.de

1 Introduction

It is increasingly recognized that the efficient supply of information and expertise is of critical importance for the success of many organizations. The adequate representation, utilization, and marketing of existing experience, expertise and competence of any organizations, which can be summarized by the term knowledge management, can be viewed in this context as a prerequisite, critical success factor, and even as commodities of modern organizations and enterprises.

This paper goes even a step further by raising the hypothesis, that „an adequate knowledge management can also improve the underlying business process“. To verify this hypothesis a specific organization -the World Health Organization (WHO)- with a strategic mission is investigated. However, WHO is one representative of a typical user community with distributed knowledge resources; consequently, the gained findings can be generalized and in particular, the corporate memory identification and utilization process can be transferred.

2 Knowledge Domains in Health Research Planning

Before presenting those knowledge domains in health research planning (viewed as a strategic decision making process) that influence its process design and that can benefit from the adequate utilization of a „corporate memory“ the specific role of WHO in health research planning is described.

As an organization mandated and committed to the improvement of global health, WHO needs to encourage and call upon the world's scientists to conduct research in the field of global health, and to facilitate the dissemination and application of major research results (WHO, 1986, 1993a, and 1993b). Noting the existence of continuing problems and of emerging risks to health, the need is recognized to improve the manner by, and the extent to, which scientific resources can support the needs of public policy makers, and to facilitate and provide incentives for more systematic research efforts from the scientific and academic communities. Therefore, for the purpose of promoting a positive and ongoing dialogue between public policy makers and the scientific community, WHO must act as a research catalyst and a facilitator of discussions between these two important stakeholder groups and must assist in building research capacity at national and international levels. It is recognized that WHO serves an important function in coordination of health research and research planning, particularly if provided with efficient central scientific support in both organizational terms and with appropriate technical facilities. Therefore, it is one of the responsibility of the organization to act as a „broker“ between the health development needs of the society (as voiced by governments) on the one side and expressed by the scientific community on the other.

WHO recognized these challenges and initiated the preparation of a guiding document concerning the contribution of science and technology towards the amelioration of the global situation with specific emphasis on the global health development. This document, called „Science and Technology for Global Health Development: A Research Agenda“ (hereinafter referred to as the Research Agenda) is accompanied by an ongoing research planning process and an implementation methodology. Fulfilling the above tasks require an adequate knowledge management. In particular,

- population health status indicators must be identified, analyzed and the corresponding problems clearly delineated,

- population health indices (profiles) must be created and research priorities set,
- research targets must be specified, dates set, and progress measures established, and
- research activities must be initiated and health programs developed, implemented and evaluated.

The WHO network of experts encompasses several organizational units, each composed of their own experts. A significant part of WHO's work is carried out by Regional Offices, each of which supports a number of countries within a region. There are six such Regional Offices (Washington, Copenhagen, New Delhi, Brazzaville, Alexandria and Manila). This paper is mainly concerned with ACHR (Advisory Committee on Health Research). The work of the ACHR is usually carried out by discussion to make recommendations. The ACHR-system typically meet only a few times a year. Due to these long intervals, the preparation of recommendations and the processing of drafts often requires more time than acceptable and actually necessary. Lead time, which is critical for long development cycles of health programs and the high costs of advanced research programs, is lost. However, no centralized research planning approach appears feasible without appropriate regional or local support. Consideration of national and regional expertise is indispensable.

2.1 The corporate memory in health research planning

The mandate and tasks of WHO, especially, its global health research strategy planning, can be viewed as a corporate business process of a distributed organization. All kind of the activities within this process are influenced by its underlying knowledge management. The knowledge management activities can be structured in the same three broad categories that encompass managerial activities: strategic, managerial (tactical), and operational (H. Simon, 1977 and Gorry, Scott-Maorton, 1971).

- **strategic**: e.g. identification of global research opportunities including their implementation
- **managerial**: e.g. control of various health programs, such as AIDS and malaria programs
- **operational**: e.g. databases management concerning the health status of the world

Another, orthogonal, taxonomy of categories range from structure to content

- **content**: the knowledge representation of the health domain: What is health? How do specific aspects of health influence each other? Which health measures and programs are available? What knowledge is missing? Where are the priorities for interventions?
- **structure**: existing organizational structure (global headquarters, regional offices, system of advisory committees, external consultants, etc.) Who is doing what? Whom to ask and for what purpose? Who has which responsibilities and/or experience?

These dimensions together create the corporate memory of WHO, which is the core basis of the reputation and competence of WHO on health issues. However, the effectiveness and competitiveness of a modern organization can substantially benefit from the effective support and coordination of knowledgeable contributions by means of adequate information and communication technology for infrastructure and services.

2.2 Knowledge domains in the health research planning process

The research planning process of WHO centers around an adequate **description of the health status**, a **detailed analysis of health deficits**, the **identification of priorities in health research** in respect to global needs, and the **transfer of this knowledge to programs** that reduce these deficits. The process has two flows - one is the identification of knowledge deficits and choosing ways of reducing them, the other to apply new knowledge to on- going programs (or by creating new programs). At the same time the way targets are reached can vary depending on the region, groups of researchers, and earlier outcomes. In the following we describe these elements in some more detail. However, we restrict ourselves to the strategic dimension from the content point of view in this section (see 2.1.):

- **Visualization of the knowledge domains (global health profile):** In order to adequately record the health status, there is the need to identify major reliable parameters describing the health status based on the latest available data and information. Based on the identification of relevant information servers, background information retrieval, previous work of WHO, and by expert hearings a selection of most significant indicators describing the health status is achieved.
- **Analysis of knowledge domains (global health analysis):** The next step is a differentiated analysis of the health status in order to identify potential areas for health research. Input for this analysis is a classification scheme developed in earlier WHO studies and an adequate visualization of the problem landscape. This analysis includes a structuring of causes, among others, along the following guideline: knowledge deficit, information deficit, and application deficit.
- **Assessment of the knowledge domains (evaluation of health deficits):** In order to find priorities for health research, the health deficits have to be evaluated and even ranked. Whereby, the importance of criteria used in this process and the resulting priorities may vary from region to region. An incomplete list of criteria for selecting priorities include the prospect for success and impact of success, the desired health care outcome: primary, secondary, tertiary prevention, the time-frame: long, medium, or short term, research type and alternative intervention options, cost benefit analysis, etc.
- **Utilization of knowledge (implementation of research projects):** All this finally leads to project proposals. The implementation of research activities and the transfer, application, and utilization of research results and programs hopefully improves the future health status. Monitoring the success of implemented measures closes this kind of a feedback loop for another cycle, beginning again with description of the health status.

The types of knowledge dealt with in the research planning process of WHO range from rather descriptive to more explanatory and evaluation-oriented aspects, but cover also knowledge concerning the quality, representation, utilization, acquisition, and dissemination aspects of these knowledge domains (in the following referred to as meta-knowledge). In particular, the tackled knowledge domains can be structured as follows:

- **description of the status quo:** knowledge mainly based on statistical data e.g. mortality data;
- **explanation of status quo:** knowledge describing the relationship and linkages between elements and factors that inform about the status quo, e.g. (mathematical) models;
- **application and utilization of existing knowledge:** knowledge concerning measures that could change (improve) the status, e.g. vaccine programs;
- **identification of knowledge gaps:** knowledge on what we do not know (meta-knowledge), e.g. the research agenda;
- **generating of new knowledge:** development of knowledge to implement new measures, e.g. research and development for a new vaccine.

It is worth noting that all these knowledge domains have not only a **static** dimension, but are very **dynamic**. For instance, the health status of a region constantly changes over time, which results in the necessity to permanent update the status quo description, with further impact on measures to improve the health status and on the identification of those knowledge gaps, if appropriate measures are not yet available.

3 The Impact of the Knowledge Domains on Process Design

3.1 Process specifications

Chapter 2 started with the mission of WHO or more abstract the strategic goal of an organization, continued with the identification of the key elements in order to achieve the strategic goals, and ended with an analysis of the involved knowledge management processes. In brief, only the knowledge management aspects concerning the „product“ (content of the research agenda) were tackled. This analysis did not incorporate the organizational structure and the involved human

competence, which is also a sort of know-how, and therefore, it does not reflect the specific value or quality which is added to the „product“ due to the various contributions of the „knowledge providers“. Also the impact of the process design on the quality of the „product“ is neglected.

Although most organizations are hierarchical, many strategic decisions are made by groups (**group decision making**). This is a rather time consuming process, especially, because of the high communication demands between members of the group. The resulting group decisions rely among others on the composition of the group, the knowledge, opinion and judgment of its participants, the decision making process used by the group, and the group dynamics.

The essential element of decision making in groups is to reach consensus concerning the decision (or identified solution) which later reduces the likelihood resistance against measures that will be taken during the implementation phase (**need for consensus**). Facilitators or moderators are often required to support strategic planning processes in order to increase the quality of the result and to speed up the process, since it is required to attain simultaneously objectives that might conflict with each other (**multiple goals**). Therefore, an adequate presentation (e.g. visualization) of those goals for comparability and assessments purposes is requested. This is especially true, since strategic planning activities are normally conducted by high level executives, and not by specialists. Thus, a knowledge representation for non-expert usage (i.e. simple, understandable, objective, standardizable) is essential.

3.2 Knowledge Management

To make the better use of the expertise and competence of an acting network, often requires to increase its efficiency. This can be attained by:

- an improved support of the collaborating tasks of the network,
- a better design of the collaborative tasks.

Both necessitate improved knowledge management. This includes, especially :

- an easy-to-use access to relevant knowledge by members of the network,
- more than satisfactory notification schemes concerning news in the corporate memory of the organization,
- the adequate structuring of the knowledge domains, and
- and the adequate acquisition and dissemination of knowledge to all interested network participants.

Moreover, an improved knowledge management has an impact on the collaboration of a network and on their „product“.

4 Utilization of the corporate memory: INTERNET based knowledge access, dissemination, and acquisition

It is becoming clear that the dire nature and extreme urgency of many global health problems represent mounting threats to global health and world stability, and therefore warrant serious attention by all who are in a position to effect positive changes. In this respect, the Research Agenda is essentially a call to arms to the members and representatives of the global science and technology community, and offers a plan according to which effective and efficient solutions to global health problems can be sought (Fliedner et al, 1995 and 1996). This initiative was designed to provide a transparent research planning and review process with which to solicit and foster the research which will be required to guide and support subsequent remedial and evaluative action.

In order to provide some degree of guidance as to the relative urgency and potential impact of respective solutions, WHO has been proposing to establish a process of identifying and assessing extant research issues so that a **continuously updated** list of the most critical research targets, referred to herein as *research imperatives*, can be provided by WHO as a research guide. (This

research planning process is based on the methodology suggested by Carrese and Baker, 1967). The research issues identified as of particular relevance and importance at any point in time is, in other words, represent a 'shopping list' from which the world's scientists, research funders, and decision-makers are urged to make a selection for their research and development activities.

To be both effective and efficient, the design, conduct and application of such research must be based on new (and global) forms of cooperation and collaboration among researchers and those involved in health-related policy- and decision-making. In today's environment, this is not possible without the utilization of advanced information and communication technologies (ICT) (Greiner et al 1996). For this reason, WHO has started to identify and publicize the above-mentioned global research imperatives, but will also provide and maintain a facilitating mechanism with which they can be most effectively addressed. (A prototype hereof will be described in the next sections.) This mechanism consists of the creation of, and support for, ICT-based networks linking collaborating scientists and other partners in a research-oriented assault on global health problem areas. WHO's vision is that of a *network of networks*, including problem-, discipline, and training-oriented networks, all of which will have as their prime objective the identification, analysis and solution of global health problems. Some will have as their mandate the creation of a new research paradigm as the armament for science to cope with emergent problems of a global nature.

As described earlier, the proper utilization of the corporate memory of such working units (networks) and an adequate knowledge management within the goal-oriented collaboration are essential. In the following, INTERNET based knowledge management examples for some of the key knowledge management domains in health research planning of WHO are presented.

4.1 Access to the corporate memory

In the case of global planning the WWW offered the best attraction, because of its wide availability. A discussion database was also deemed to be essential and this was provided through special interactive pages developed for the WWW site. As a final design step, selected services are integrated into a platform (Schmidt and Rodden, 1992). This provides an interface that allow users to easily select services and move easily between them. Usually this means that, because of technical constraints on seamlessness, a minimum set of services is used.

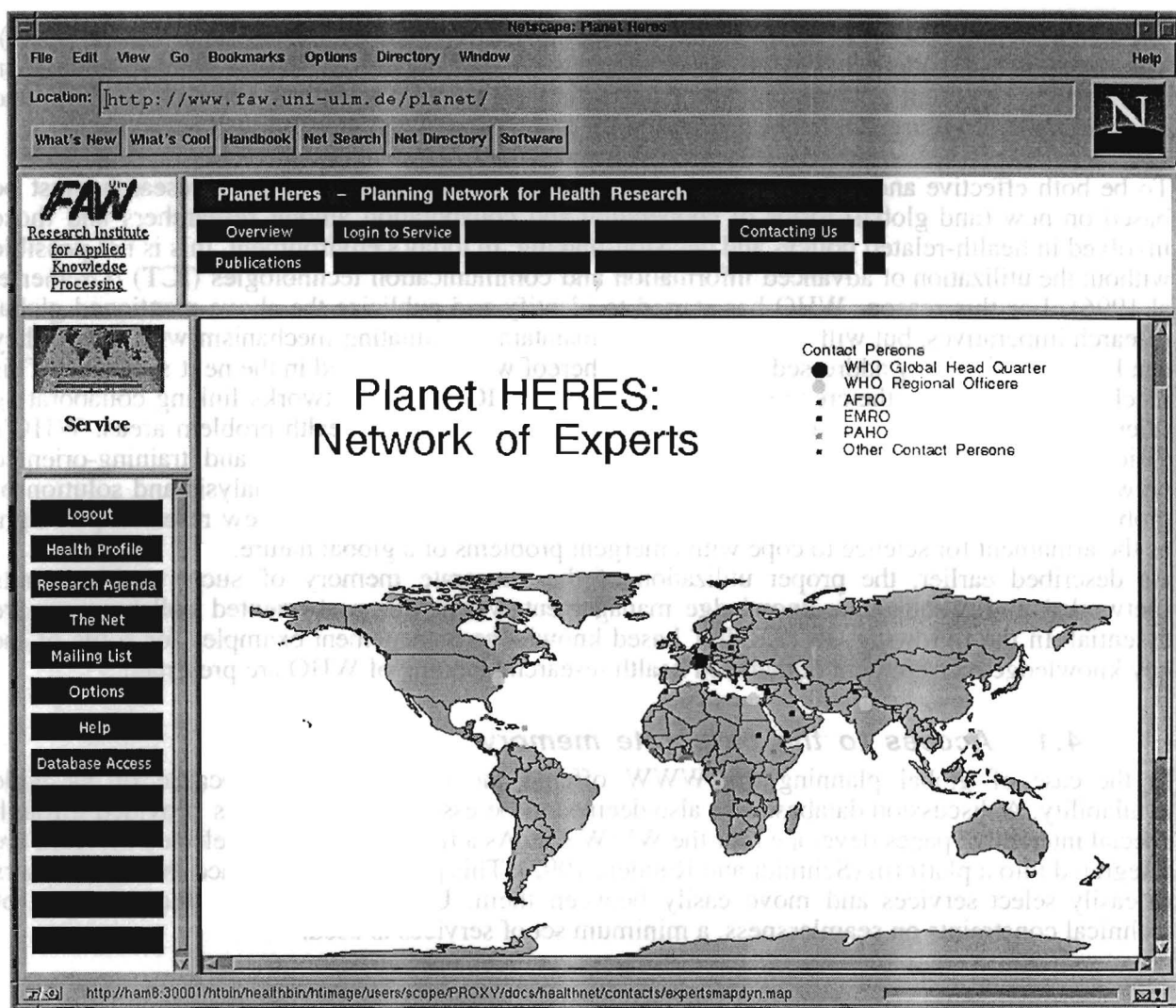


Figure 1: The Planet HERES home page, with a set of services allowing to utilize the corporate memory of WHO for global health research planning purposes on the left. In particular, this page shows the so-called contact map, which offers information concerning the experts involved in the research planning process (specific background, and expertise, contact information, etc.).

Our first step was to develop a demonstrator system based on the WWW to provide a platform for such services. This is illustrated in Figure 1 and is accessible via <http://www.faw.uni-ulm.de/planet>. The site provides both public services as well as those intended for restricted use only by members of the WHO. The top menu for example, includes contact points, access to publications as well as the overview of the operation of health planning. Selection of 'login to Service' brings up the menu shown on the left and is accessible only to people through password control.

In more detail, the WWW interface to Planet HERES (**Planning Network for Health Research**) provides support for basic services like, access to relevant databases, information of involved experts, a mailing list for general announcements. These services are designed to utilize the structural dimension of the corporate memory.

On the product or content dimension of the corporate memory, services for the analysis of the status quo of the global health situation (see 4.2) and for a moderated discussion forum to generate the research agenda (see 4.3) are provided.

4.2 The HEALTH PROFILE

A visual presentation of data describing or representing the health and well-being of a given population in an holistic manner must include disease associated information as well as information describing causal and contributing factors to disease conditions and health impairments, such as social, political, economic and environmental dimensions of a community. The chosen approach, the „Visual Health Information Profile“ (Beyrer, Greiner, et al, 1997 and <http://www.faw.uni-ulm.de/planet/healthnet/circle.html>) consists of a disc shape.

The indicators are grouped into five sectors called „domains“. Each domain is divided into various smaller sectors which represent specific indicators radiating out from the center. Each of these sectors are scaled linearly from 1 to 10 beginning at the periphery and progressing towards the center. Each indicator value is plotted onto „its own“ sector on the disc after receiving a relative rating between 1 and 10 using the decile rank ordering procedure described below. Indicators receiving low relative scores will be represented by sectors projecting out towards the periphery of the disc, while measurables with high relative scores will be located closer to the center of the disc. Projecting all of the indicators onto the disc yields a pattern, or profile, highlighting the relative differences between the applied indicators. This „strengths“ and „deficits“ are visually easily identified. For an example see figure 2.

A hierarchical approach is used in the Visual Health Information Profile which groups the health parameters into five domains. These include: 1) Disease Conditions and Health Impairments, 2) Health Care System, 3) Socio-Cultural Characteristics, 4) Environmental Determinants, and 5) Food and Nutrition. While the indicators represent the components generating the top level of the health profile, each indicator can be broken down further if sufficient detailed statistical or epidemiological data are available. Thus it is possible to always analyze a selected health indicator by disaggregating a given value down to its origin or to the actual problem level. This feature allows incorporating both aggregated and disaggregated data into the Visual Health Information Profile without losing essential information. The structure of the data base used in generating this health profile thus presents to the user with a very high degree of transparency.

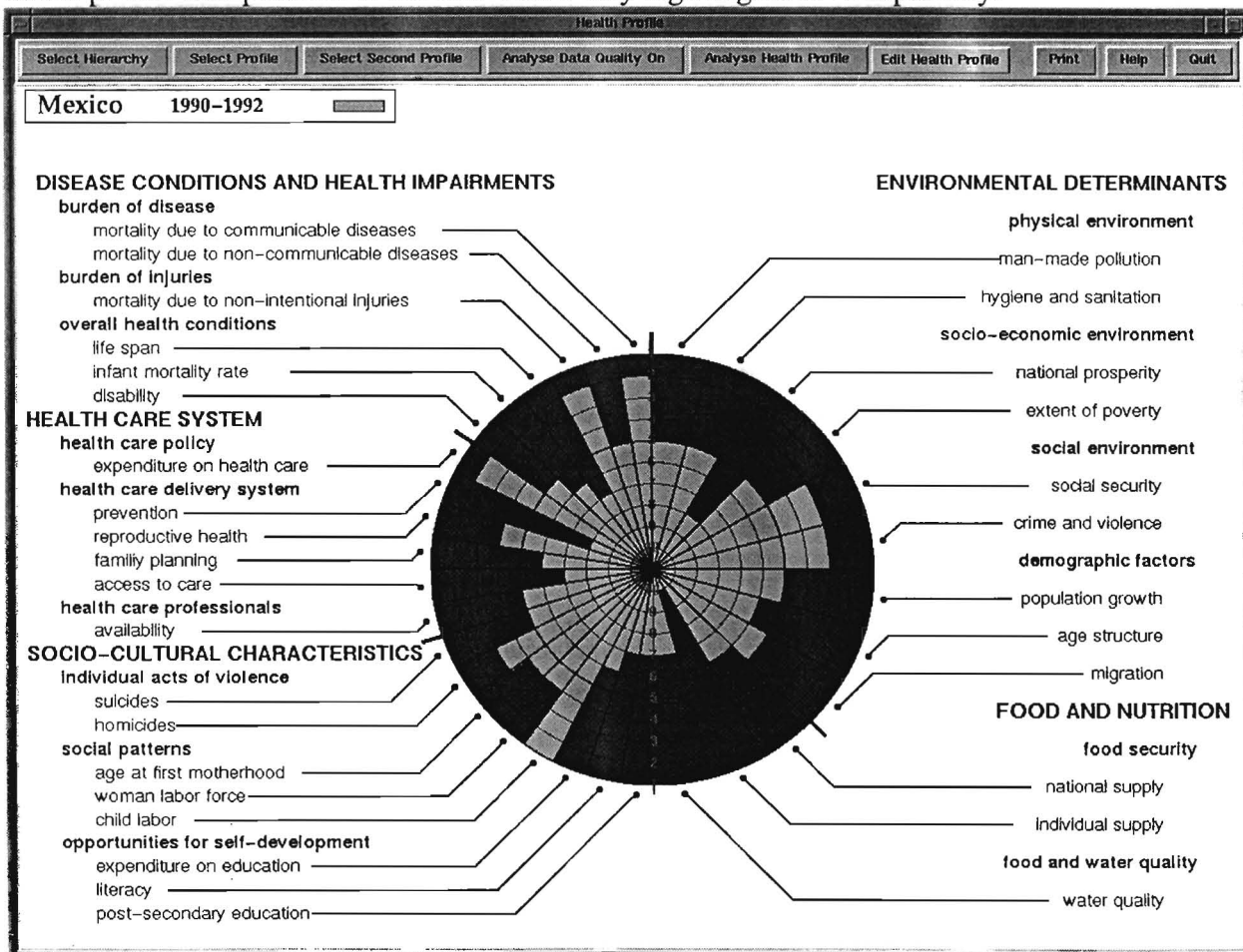


Figure 2: Visual Health Information Profile for Mexico. Five domains including 32 indicators are shown. The relative score of "1" is located at the periphery and the score of "10" is located at the center. "Deficits" are seen for indicators projecting out towards the periphery. Major deficits are seen in four domains.

4.3 The RESEARCH AGENDA

The Research Agenda outlines a framework of concepts and methods according to which individual members and representatives of the global science and technology community can be enlisted and coordinated in systematic and ongoing research initiatives targeted on making significant improvements in the global health situation.

The major planning support in order to develop the research agenda that can be selected as one of the login services is a moderated research agenda discussion forum. As shown in Figure 3 it provides the current listing of issues that have been raised in the agenda. The discussion forum shows the general outline (similar to the table of content of a book) of the research agenda, as developed in previous meetings, discussions and questionnaires, which represents the actual status of the on-going discussion.

Since each topic in this agenda should be further discussed an easy to use interface was developed. Users can make contributions by entering a particular issue (by clicking on it). Then user will receive an overview of the contributions of the other participants as well as it will give easy access to enter new recommendations or comments.

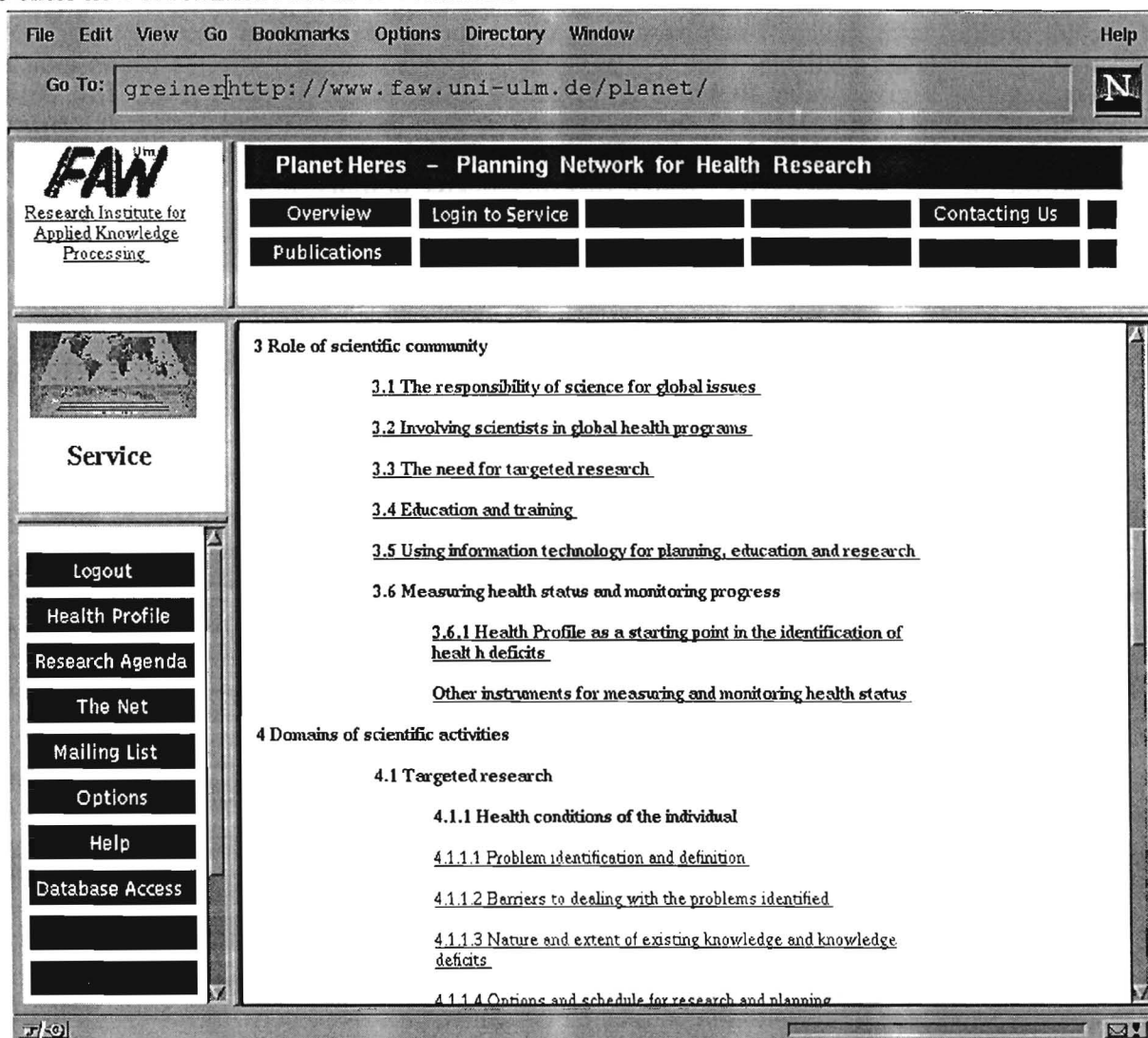


Figure 3: Moderated Research Agenda

5 Discussion and lessons learned

The development of an INTERNET based information system to support the various phases of research planning activities of WHO showed that a proper identification of existing experiences, competence (knowledge) is essential. The developed methods to deal with the involved knowledge domains have proven their validity. In particular, 1.) the health profile as presentation schema for statistical data, that can be combined with hierarchical structuring hereof and 2.) the research agenda as a example more explanatory information for the development of a strategic goal within a organization can be viewed as key elements for the management of knowledge concerning the content or product of a organization. Quality can not only be added to the product by the adequate support of those content oriented knowledge management technologies, but also be making full use of the existing structure and competence of the organization. In this respect, services like the described contact-map (easy access to structural knowledge of a organization) have their specific potential. Especially, due to the parallel design of the research planning process, development of content and structure oriented services, and finally, the identification of research imperatives, it was possible to monitor at different development stages the impact of the utilization of corporate memory on the „process“ and on the „product“. It came apparent, that knowledge management is one of the key elements for successful business processes. In this respect, as a example the adequate utilization of the human resources is mentioned.

The **application potential** of the presented approach lies in the various generic representation schemes of „knowledge“. This structural information can be used to develop powerful filters and brokers for any application domain be it in industry or administration. The planning process, a electronically supported, modified Delphi technique, can also be transferred to other situation, especially in the identification of strategic goals within organizations or enterprises.

In conclusion, the example of health research planning shows that the intensive communication needs can be supported by the utilization of currently available information and communication technology. An adequate knowledge representation of the corporate memory is essential. Thus, existing processes can be made to work more effectively by identifying deficits and transferring knowledge in shorter time frames and to deliver qualitatively, vastly improved products.

6 References

- Beyrer K., Brauer G.W., Fliedner T.M., Greiner C., Reischl U., Schabacker-Bock M.(1996): „VISUAL HEALTH INFORMATION PROFILE (VHIP): A Quantitative Approach for Analyzing the Health Status of a Population“, MIE97, Tessaioniki, Greece, 1997.
- Carrese, L.M. and. Baker C.G (1967): „The Convergence Technique - A method for the Planning and Programming of Research Efforts“, Management Science, Vol. 13, No. 8, USA
- Fliedner, T.M. et. al. (Eds.) (1995): „Strategic Planning in Health Research“, Report from the ACHR Workshop: „New Approaches for Health Research Strategy Planning“, WHO-Collaborating Center for Global Modeling of Health Perspectives, International Institute for Scientific Cooperation, Schloß Reissensburg, Germany, April 3-7, 1995.
- Fliedner, T.M.; et. al. (Eds.)(1996): „Strategic Planning in Health Research“, Report from the ACHR Workshop: „New Approaches for Health Research Strategy Planning“, WHO-Collaborating Center for Global Modeling of Health Perspectives, International Institute for Scientific Cooperation, Schloß Reissensburg, Germany, March 11-15, 1996.
- Gorry, G.M., Scott-Morton, M.S. (1971): „A Framework for Management Information Systems“, Sloan Management Review, 1971.
- Greiner, C., Hawryszkiewicz, I.T., Rose, T., Fliedner, T.M. (1996): „Supporting Health Research Strategy Planning Processes of WHO with Information Technology“, Proceedings of the GI-Workshop on „Telecooperation systems in decentralized organizations“, Berlin, March 22-23.
- National Institute of Health (USA), National Cancer Program, Systems Planning and Information Systems Overview, US Department of Health, Education and Welfare, NIH, 1973.
- Schmidt, K., Rodden, T. (1992): „Putting it Together: Requirements of a CSCW Platform“, Proceedings of 12th. Interdisciplinary Workshop on Informatics and Psychology, Shaerding, Austria, June, North-Holland.

Simon, H. (1977): „The New Science of Management Decisions“, Prentice-Hall, New Jersey, 1977

WHO (1986) Health Research Strategy for Health for All by the Year 2000, Report of a Subcommittee of the ACHR, WHO/RPD-ACHR(Ed.), Geneva.

WHO (1993a) Health for all targets - The health policy for Europe, WHO, Regional Office for Europe, Copenhagen.

WHO (1993b) Implementation of the Global Strategy for Health for All by the Year 2000, Volume 1: Global Review, WHO, Geneva.

A Framework for Knowledge Management Systems : A Proposal

Wai Keung Pun, Craig McDonald & John Weckert
School of Information Studies
Charles Sturt University
Locked Bag 675, Wagga Wagga, NSW 2678 Australia
{dpun | cmcdonald | jweckert}@csu.edu.au

Abstract

Traditionally expert systems have been built from knowledge elicited from domain experts. However, knowledge in applied science domains is grounded in published sources like research reports, text books, articles and the like. This corpus of knowledge is typically inconsistent, dated, dispersed, and so on. The project described in this paper aims to construct a putative Knowledge Management System. The core of the system is a knowledge server that represents each publication and expert as a separate knowledge base, and a meta-knowledge base to allow different kinds of access to the server. Different client systems can be connected to the knowledge server to meet different user needs, such as forecasting, advice, explanation, education, and training. The server can also be a resource for researchers and research managers, by allowing hypothesis testing and reviews of the literature. Knowledge re-engineering is not necessary, as the system simply embodies what is in the domain. The knowledge is being represented in conceptual graphs and the test domain is irrigation. The work is being supported by the Cooperative Research Center for Viticulture.

1. Introduction

Before the knowledge created by applied science research can form a normal part of industry practice, it must be published, presented at conferences and seminars, built into training and education courses, and slowly 'percolate' through the community. This process can take considerable time, and much detail is lost or misinterpreted along the way. The Cooperative Research Centre for Viticulture (CRCV) in Australia is investigating methods of building applied research results into a knowledge-based system *as a matter of course* so that new knowledge can be put to use in the grape growing industry. Such a system would provide a vehicle for quick and complete promulgation of research results. We envisage a future where knowledge created in the laboratory and in the field can be reported to a knowledge-based system and become immediately effective in viticultural practice.

The project described here aims to find ways of representing applied research papers and reports directly in a knowledge management system (KMS), and of establishing the "meta-knowledge" necessary to properly mobilise the knowledge embedded in the literature. Such a system will enable multiple kinds of access to the knowledge, by decision support systems or computer-aided education systems for example, which will use the knowledge in different ways, for advice, forecasting, education and training, explanation and so on. It will also be a resource for researchers in hypothesis testing and research management. A prototype KMS is being built in the irrigation of grapevines as a means of evaluating the KMS approach.

2. Problem Description

Human knowledge takes two forms: private and public (Kemp, 1976). Private knowledge is that held in and used by the minds of humans. In its public form, knowledge is published as periodical articles, research papers, conference proceedings, technical reports, textbooks and so on. The applied sciences create public knowledge through research and publication, but current methods of organising and mobilising this knowledge are flawed. Considered as a whole, the applied science literature is:

Dispersed: It is scattered across different kinds of literature such as books, periodicals, research papers, technical reports, proceedings, which are located all over the globe. It is possible that research is unwittingly being duplicated because the original was not found in the literature review.

Dated: Some knowledge created long ago has been superseded by more recent work, but still remains in the literature with a potential to mislead.

Under-utilised: Studies indicate that no more than 20 percent of the knowledge available in research institutes is really being used (Mühlemann, 1995). Therefore the full weight of current human knowledge is not brought to bear on problem solving.

Expanding rapidly: The quantity of knowledge is increasing at an exponential rate.

Variable in quality: The reliability of the public knowledge is complex. Bauer's knowledge filter theory (Rauscher, 1993) mentioned that "Textbook Science" is more reliable than primary (e.g., research papers) and secondary literature (e.g., review articles). Furthermore, knowledge that is reliable in one context may not be so reliable in another.

Inconsistent: Considerable contradictions have been found within the published knowledge and between the published knowledge and expert opinion (McDonald & Ellison, 1994).

Incomplete: There are considerable gaps between the published knowledge and expert opinion. For example, in the development of the AusVit module (McDonald & Ellison, 1994) to deal with the disease caused by *Botrytis cinerea* a number of questions arose which had a great bearing on advice being given by the system but for which there were no answers in the literature.

Slow to be published and applied: Publication in scientific journals can take 12 to 18 months after acceptance, which may have taken a year itself. This will lead to a delay factor in decision making. The path from applied science research to decision making in the field can be long and inefficient.

Clearly, there is a large knowledge management problem to be addressed here. Current approaches to the problem come from either information management technology (document indexing and bibliographic databases which store and deliver papers) or expert systems technology (advice giving systems built from consensus knowledge of domain experts). The former requires a person to interpret the information delivered while the latter is often pervaded by imprecision and/or uncertainty (Grabot 96).

The research project described here aims to employ knowledge based technology to deal more effectively with the knowledge management problem. The KMS under development will collect and consolidate knowledge in a form that is explicit and accessible, while still preserving the context of each research publication. By avoiding some of the problems in current knowledge

management, the KMS will be a powerful tool for technology transfer, allowing complete, unbiased and justifiable responses to industry problems and for research management. In the future, research results will be input to the KMS as though they were data. Of a parallel domain, forest science, McRoberts *et al.* (1991) say:

Computerised database management systems have been accepted as essential aides to the human mind for decades now. No one would dream of trying to manage a large forest inventory on paper or in the minds of humans any more. Computerised knowledge base management systems are making it equally wasteful to manage forest science knowledge in paper journals and books, or in the minds of human scientists. The volume is too large and, thanks to the advances in AI, the computer can now cheaply store and retrieve knowledge as easily as it can store and retrieve data. (p20)

The KMS will incorporate and integrate new knowledge that is being created in applied research projects around the world.

3. A KMS Architecture

A prototype KMS is being constructed with two components. The first is a set of knowledge bases each representing the knowledge in a particular research paper or report. In the KMS, each publication is treated like a small single and independent knowledge base. The second component is a meta-knowledge base that represents aspects of each research publication. These aspects influence the selection of which knowledge base is applicable in a particular instance.

The KMS will be used by a range of interface systems that will employ it in different ways. For example, a decision support system will use the KMS as a model of a domain to allow scenario processing. An expert system will give advice using the KMS as a knowledge base and justify the advice on the basis of the publications from which the KMS has been built. A Computer Aided Instruction (CAI) interface would allow the KMS to form the basis of courses in the domain. Researchers and research bodies can use the KMS as a source for literature reviews and hypothesis testing. Each of these interface systems will have specific systems components suitable to their purposes but will rely on the KMS as the source for their domain knowledge.

As each new research report becomes available it is represented as a new document-related knowledge base and so participates immediately in the various uses to which the system is being put. Figure 1 shows the KMS architecture.

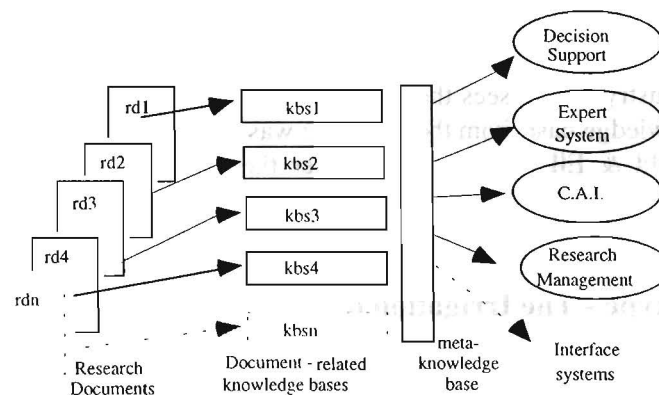


Figure 1: Knowledge Management Systems Architecture

The research involved in the construction of the KMS centers on the development of methods for knowledge extraction from literature, knowledge representation in conceptual graphs (Sowa, 1984), knowledge query, and access to KMS by the interface systems mentioned above.

4. Case Study - Viticulture

The Cooperative Research Centre for Viticulture (CRCV) in Australia carries out basic and applied science research on grape vines and their management. As part of its technology transfer program, the CRCV has developed an expert system, AusVit. The system provides advice to vineyard managers and grape growers about pest and disease risk in their vineyards and what appropriate action might be taken. The system also advises on irrigation, chemical use, and the like. The advice is based on vineyard profile, data from weather stations and user input from vineyard monitoring, all of which is interpreted by a series of disease simulators and a rule-based expert system. A chemical database provides details of the active components in agricultural chemical products, their application and registration information. The components of the system are shown in Figure 2.

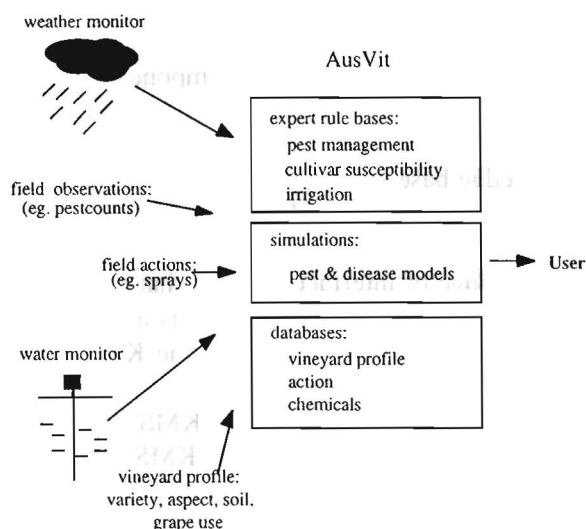


Figure 2. The inputs and components of AusVit

The rule base has been built using the traditional expert systems approach (Travis, 1992). The CRCV is interested in transforming AusVit from a traditional expert system to a KMS. An aim of the CRCV is to ensure that the results of its commissioned applied viticulture research are transferred to industry, and it sees the KMS as a vehicle to facilitate that transfer. A pilot study of building a knowledge base from the literature was conducted in the *Botrytis Cinerea* module of AusVit (McDonald & Ellison, 1994) and over the next two years the expert rule bases and simulations in one module of AusVit will be replaced by a KMS.

5. KMS Prototype - The Irrigation of Grapevines

Irrigation plays an important part in viticulture. It is a powerful technique for improving vine performance, because it allows an environmental factor (water) to be placed under managerial control. Grapes are grown in Australia in regions with annual rainfall as low as 250mm to as high as 1100mm. In those regions with low rainfall, irrigation is necessary, because without it

vineyards would be uneconomic due to water stress. Research into irrigation for grapevines has therefore increased significantly during recent years.

To show how a KMS for vineyard irrigation might work, the following section gives an example of literature being used as a source for a set of Conceptual Graph (CG) knowledge bases and Section 5.2 shows how these CGs might be used in different ways to meet different user needs.

5.1. Conceptual Graphs Knowledge Bases from Published Literature

As an example of developing conceptual graph knowledge bases from published literature, three papers, Goodwin (1995), McCarthy *et al.* (1993), and Williams and Grimes (1987), have been selected and CG representation of their content created. These CGs are intended as examples only.

A) McCarthy *et al.* (1993)

“Evaporation (ES): Water stored in the soil is lost by evaporation from the soil surface. The extent to which evaporation from the soil surface contributes to evapotranspiration depends on the frequency of wetting of soil, the area of soil surface wetted, and the proportion of the wetted soil surface that is shaded.
Air temperature, humidity, and wind speed at ground level also affect evaporation from the soil surface.
Transpiration (EF): Loss of water from vine foliage. Water vapour in the air spaces within leaves diffuses to the outside air through numerous valve-like pores (stomata) on the surface of the leaves.
Evapotranspiration: As both processes involve the use of radiant energy they are collectively called evapotranspiration ($ET = ES + EF$).
If all pores in the soil are filled water and no air then soil is saturated and is at the Drained Upper Limit (DUL).
Eventually a level of soil water is reached when plants can no longer extract enough water and they begin to wilt. When plants wilt by day and fail to recover at night, the soil is at the Lower Limit (LL).
Plant Available Water (PAW) is the amount of water held in the soil between DUL and LL and is the water that can be used by the plant. It can be expressed as millimetres of water per metre of soil (mm water / m soil). The amount of available water that a soil profile can store depends on its texture ranging from 33 to 208 mm per metre.
Actual vineyard water use (soil evaporation + plant use) is reported to be as low as about 250 mm to more than 800 mm.”

[TRANSPARATION] -

(OBJ)->[WATER: #]->(STORE)->[PLANT: #]
(LOSS)->[LEAF_SURFACE]->(ATTR)->[AREA: #]

[EVAPORATION] -

(OBJ)->[WATER_CONTENT_OF_THE_SOIL: 1/4 DUL]<-(OBJ)<-[STORE]->
(AGNT)->[SOIL_TEXTURE: Sandy Loam = *x]
(LOSS)->[SOIL_SURFACE]->(ATTR)->[AREA: #]

[EVAPOTRANSPIRATION] -

(LINK)->[EVAPORATION]->(OBJ)->[RADIANT_ENERGY]->(CHRC)->[USE: #]
(LINK)->[TRANSPARATION]->(OBJ)->[RADIANT_ENERGY]->(CHRC)->[USE: #]

[WATER_CONTENT_OF_THE_SOIL] -
 (CONTENT)->[DRAINED_UPPER_LIMIT]
 (CONTENT)->[LOWER_LIMIT]

[SOIL_TEXTURE: SAND] -
 (ATTR)->[SOIL_PROFILE: Moderately Coarse]
 (ATTR)->[TEXTURE_RANGE]->(MEASURE)->[MEASURE: 104mm]

[DRAINED_UPPER_LIMIT] -
 (OBJ)->[WATER_CONTENT_OF_THE_SOIL]->(MEASURE)->[MEASURE: 54mm]

[LOWER_LIMIT] -
 (OBJ)->[WATER_CONTENT_OF_THE_SOIL]->(MEASURE)->[MEASURE: 31mm]

[PLANT_AVAILABLE_WATER] -
 (OBJ)->[SOIL_TEXTURE: *x]<-(MEASURE)->[MEASURE: 1metre]

[ACTUAL_VINEYARD_WATER_USE] -
 (RSLT)<-[NUMBER: #]<-[SOIL_EVAPORATION]<-(ARG)<-[ADD]->
 (ARG)->[PLANT_USE]->(RSLT)->[NUMBER: #]

B) Goodwin (1995)

“Vineyard irrigation is best defined as the efficient application of water to maximise profit and minimise environmental degradation.

The aim of an irrigation is to replace the water used by the vineyard since the previous irrigation. The timing and the amount of irrigation will depend on the rate of water use and the quantity of available water held in the root zone. Knowledge of vineyard water use is therefore a critical component of irrigation scheduling.

Water stress is a physiological reaction of a vine to a limitation in supply of water. Some of the physiological responses of grapevines include: closing of leaf stomata, reduced photosynthesis, reduced cell division and loss of cell expansion.

The vines must use up the total storage of available moisture from rainfall before an irrigation is necessary. When to start irrigating is therefore a function of how much water is stored in the soil and the daily rate of water use by the vineyard.

When to start irrigating (days from bud burst) =
 Soil water storage (litres) / Daily vineyard water use (litres/vine/day)

[IRRIGATION_AIM] -
 [EVENT:
 [REPLACE]->(OBJ)->[WATER_USE]->(LOC)->[VINEYARD: #]] -
 (SUCCESSOR)->[EVENT:
 (PAST)->[PROPOSITION]: [VINEYARD: #]<-(LOC)<-[IRRIGATE]->(OBJ)->[WATER_USE]]

[WHEN_TO_START_IRRIGATING] -
 (RSLT)->[DIVIDE] -
 (RSLT)->[STORE] -
 (AGNT)->[SOIL: #]
 (OBJ)->[WATER: #]
 (MEASURE)->[MEASURE: # Litres]
 (RSLT)->[WATER_USE] -
 (LOC)->[VINEYARD: #]
 (OBJ)->[VINE: #]
 (FREQUENCY)->[FREQUENCY: Daily]
 (MEASURE)->[MEASURE: # Litres]

[IRRIGATION] -
 (FREQUENCY)->[TIMING: #]
 (QTY)->[AMOUNT: #] -
 [EVENT:-
 [DEPEND]->(OBJ)->[WATER_USE] -
 (MEASURE)->[RATE: #]
 (STATE)->[AVAILABLE]
 (SOURCE)->[PLANT: #]->(LOC)->[ROOT]]
 (NECESSARY)->[PROPOSITION: [WATER_USE] -
 (LOC)->[VINEYARD: #]
 (OBJ)->[KNOWLEDGE]
 (SUPPORT)->[SCHEDULING]->(INST)->[IRRIGATION]]

C) Williams and Grimes (1987)

“An important aspect of this study was to establish irrigation regimes that reflected best estimates of vineyard potential evapotranspiration and then apply water equivalent to that ET.

It was interesting to note that the relationship between applied water and growing degree days (GDDs) was linear. This would indicate that ET and vine growth were temperature dependent.

A constant level of soil moisture did not occur for the 0.4 ET treatment, however, until 1000 GDDs after budbreak at Kearney Agricultural Centre. The level of soil moisture that resulted from the 0.4 ET treatment throughout the growing season were sufficient to induce a water shortage for vines in this treatment.

The plant based measurements of vine water status indicated that grapevines receiving less than 1.0 ET in this study were under stress. These measurements have been used by many as a measure of the degree of stress experienced by the vine during a period of water deficits (Smart 1974, Hardie and Considine 1976, Kliewer, Freeman and Hossom 1983, Liu *et al.* 1978).

[VINE_WATER_STATUS] -
 (MEASURE)->[MEASUREMENT]->(METHOD)->[METHOD: PLANT BASED]
 (INDICATE)->[STATE:
 [VINE: GRAPEVINES = *x]->(RECEIVE)->[ET]->(LESS)->[NUMBER: 1.0]]->
 (CAUSE)->[STATE: [VINE: *x]->(ATTR)->[UNDER_STRESS]]

[MANY]->(QTY)->[PERSON: #]->(INST)->
 [EVENT:
 [MEASUREMENT] -
 (METHOD)->[METHOD: Plant Based]
 (OBJ)->[VINE: #]
 (CHRC)->[THE_DEGREE_OF_STRESS]
 (POINT-IN-TIME)->[DURING_A_PERIOD_OF_WATER_DEFICITS]]

(NEG)->[PROPOSITION:
 [OCCUR]-
 (TREATMENT)->[ET]->(QTY)->[NUMBER: 0.4]
 (OBJ)->[SOIL_MOISTURE]->(MEASURE)->[CONSTANT_LEVEL]]

5.2. Using the Conceptual Graphs Knowledge Bases

The KMS is designed to be used in different ways for meeting different users needs. Here we show how the KMS can be used as an expert system to simulate the problem-solving behaviour of an expert, can be employed as a decision support system to allow scenario processing, and can be also used as a computer aided instruction system for education and training.

I) KMS as an Expert System

In order to remain competitive, grape growers and vineyard managers often depend on agricultural advisers and experts to provide advice for their decision making. This advice is costly and expert assistance is not always available when the grape growers and vineyard managers need it. In this situation, the KMS can be used as an expert system. For example, a dialogue between KMS and user might be as follows:

User : *"Given the current state of the vineyard, should I irrigate ?"*
KMS : "Yes. Irrigate to field capacity."
User : *"How did you come to that advice?"*
KMS : "Your ET is less than .01 that implies water stress (Williams and Grimes, 1987). Your soil type is sand so you should irrigate to 31mm, the Drained Upper Limit (McCarthy *et al.* 1993)."

The KMS would be using the CGs both to come to a decision and to justify that decision specifically on the basis of the literature. As an expert system, the KMS combines CGs from many sources including the experiential knowledge and intuitive reasoning of many experts. The KMS would use a meta-knowledge base to select suitable knowledge bases for the response.

II) KMS as a Decision Support System

The grape growers or the vineyard managers could test scenarios through the decision support system (DSS) interface to the KMS, for example:

User : *"If the weather is hot over the next week, will I need to irrigate?"*
KMS : "Current water_content_of _soil is 40%. Hot weather implies high radiant energy and high evapotranspiration (McCarthy *et al.* 1993). Expected stress_level in one week is ?? (Goodwin,1995)."

This kind of 'what-if' processing reasons with the CG's by setting up the conditions that would apply in the scenario nominated by the user. Another possibility is the use of the knowledge bases to determine what scenarios would be necessary for a specified outcome to occur (Richards & McDonald, 1995). For example:

User : *"Under what conditions will I need to irrigate next week?"*
KMS : "Current water_content_of _soil is 40%. If there is Hot weather then there is high radiant energy and high evapotranspiration (McCarthy *et al.* 1993). Expected stress_level in one week is ?? If there is no rain you will need to irrigate (Goodwin, 1995)."

III) KMS as a Computer Aided Instruction System

The KMS can also be employed as a computer aided instruction (CAI) system to support education and training. For example, students enrolled in the irrigation module of a viticulture course might be presented with text followed by questions that they answer online:

KMS : “Vineyard irrigation is best defined as the efficient application of water to maximise profit and minimise environmental degradation.
The aim of an irrigation is to replace the water used by the vineyard since the previous irrigation. The timing and the amount of irrigation will depend on the rate of water use and the quantity of available water held in the root zone. Knowledge of vineyard water use is therefore a critical component of irrigation scheduling.
Water stress is a physiological reaction of a vine to a limitation in supply of water.

Question : List the factors that remove water from the soil.“

User : “*vine transpiration, evaporation, drainage*“

Their answers could be verified by reference to the CGs and student learning enhanced by other systems facilities such as information retrieval or simulations based on the CGs.

6. The Potential

The case study described above raises many interesting issues concerned as much with the nature of applied science itself as with the technicalities of constructing a CG based KMS. However, the system has the potential to become an effective vehicle for technology transfer and knowledge management. It will have:

Up to date knowledge: Because AusVit will contain the most recent research results as well as a full history of non-obsolete research it will be complete and up to date. As new research is entered the advice that the system gives will change.

Flexible knowledge application: To apply knowledge to a problem AusVit will weight the applicability of each of the various literature sources according to its match with the vineyard profile and prevailing conditions.

Explanation: Giving useful explanations of their advice has been a difficult issue for expert systems, in part because of the disassociation of the explanation facility from the actual reasoning in expert systems, and in part because experts can not explain how they know something. Explicitly basing both reasoning and explanation in the literature has the potential to add a new dimension to explanation.

Research implications: The pilot study of building a knowledge base from the literature revealed a number of questions that had a great bearing on advice being given by the system, but for which there were no answers in the literature. It also found contradictions between sources. Such gaps and contradictions in the literature can generate new research projects. The knowledge-based system will become a source of information for researchers, much like a data base (e.g., Managing the Global Climate Change Knowledge Base (Rauscher, 1993)), but one that holds active knowledge rather than passive information. It would, for example, allow hypothesis testing (Davis, 1991). This raises an issue for applied science funding bodies like the CRCV - given the bodies' strong industry orientation. For example, if the

results of one of its research project cannot be built into a KMS, or, if it is built in but has no impact on the advice given by the system, was it really applied science research?

Educational uses: The possibilities for using the system in education and training are clear, especially if the system captured complete literature sources and had a range of computer-based learning facilities (e.g., interfaces, programmed instructions, concept maps).

7. Conclusion

It is known that conceptual graphs are suitable for representing and processing knowledge due to their strong expressive ability and well-defined operations. Other reasons for using conceptual graphs, which are adopted as a knowledge representation language in this project, are that they allow advanced explanation, knowledge transportation and knowledge re-use and have potential to subsume a range of other forms of representation.

AusVit is a part of a growing trend to manage scientific knowledge using computer systems. Information technology has an extraordinary rate of change and its ability to deal with highly complex and voluminous data is increasing rapidly. It is already the primary vehicle for recording information and it will become the primary vehicle for mobilising knowledge. Systems builders of the future will have to come to grips with the issues of knowledge management rather than knowledge engineering.

References

- Davis, M. & Compton, P. (1991). Hypothesis Testing: A New Information Retrieval Task for Libraries. In *Libraries and Expert Systems*, eds. Craig McDonald & John Weckert. Taylor Graham: United Kingdom, pp. 142-148.
- Goodwin, I. (1995). *Irrigation of Vineyards: A Winegrape Grower's Guide to Irrigation Scheduling and Regulated Deficit Irrigation*. Institute of Sustainable Irrigated Agriculture: Tatura, Vic., Australia.
- Grabot, B. & Caillaud, E. (1996). Imprecise Knowledge in Expert Systems: A Simple Shell. *Expert Systems With Applications*, **10**, No. 1, pp. 99-112.
- Kemp, D. A., (1976). *The Nature of Knowledge: An Introduction for Librarians*. Clive Bingley: London & Linnet: Connecticut, USA.
- McCarthy, M. G., Jones, L. D. & Due, G. (1993). Irrigation - Principles and Practices. In *Viticulture*, **2**, eds. B. G. Coombe & P. R. Dry. Australian Industrial Pub: Adelaide, Australia, pp. 104-127.
- McDonald, C. & Ellison, P. (1994). Knowledge Acquisition from Domain Literature. In *AI and Natural Resource Management: A Workshop in Conjunction with AI'94*, University of New England, 22 November 1994, Armidale, Australia, pp. 83-91.
- McRoberts, R. E., Schmoldt, D. L. and Rauscher, H. M. (1991). Enhancing the Scientific Process with Artificial Intelligence: Forest Science Applications. *AI Applications*, **5**, No. 2.
- Mühlemann, S. (1995). Das verschüttete Kapital. *Spezial/Wissensmanagement, BILANZ*, **1**, pp. 76-80.
- Rauscher, H. M. & Dale, W. J. (1993). Managing the Global Climate Change Knowledge Base. *AI Applications*, **7**, No. 4.
- Richards, D. & McDonald, C. (1995). Bridging the Expert System - Decision Support System Gap: Adapting and Re-using an Existing Expert System. In *Proceedings of Sixth Australian Conference on Information Systems*, School of Information Systems, Curtin University of Technology, Perth, Western Australia, p219-229.

- Sowa, J. F. (1984). *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley: Reading, Massachusetts.
- Travis, J. W. *et al.* (1992). A Working Description of the Penn State Apple Orchard Consultant, An Expert System. *Plant Disease*, **76**, No. 6.
- Williams, L. E. & Grimes, D. W. (1987). Modelling Vine Growth - Development of a Data Set for a Water Balance Subroutine. In *Proceedings of the Sixth Australian Wine Industry Technical Conference*, Adelaide, South Australia, ed. T. Lee. Australian Industrial Publishers: Adelaide, Australia, pp. 169- 174.

Knowledge Integration for Building Organisational Memories

Ulrich Reimer

Swiss Life

Information Systems Research Group

CH-8022 Zürich, Switzerland

phone: +41-1-7114061, fax: +41-1-7115007, email: ulrich.reimer@swisslife.ch

Abstract

The paper starts with a discussion of the roles an organisational memory (OM) should play and what kind of knowledge should go into it. We then identify two kinds of integration problems. The first one is concerned with integrating the knowledge bases of different knowledge-based systems employed in an organisation into one physically or virtually unified knowledge base which is to be considered as part of the organisation's OM. The second problem concerns the integration of several representations of the same knowledge with different degrees of formalization, ranging from formally represented knowledge via semi-structured text to plain text. This is an issue because formally represented knowledge, e.g. company regulations, often also exists in textual form, and both representations are needed for different kinds of tasks. It is argued that the two integration problems mentioned can only be solved by making use of a high-level language whose representation constructs are on the conceptual level (in the sense of Brachman) and which covers all representational needs. We argue that such a language can be made easy to use despite its being extremely comprehensive if the representational ontology underlying its constructs is represented explicitly.

1 Introduction

It is increasingly acknowledged that knowledge is one of the most important assets of organisations. Especially in industrialised countries with expensive but highly educated employees, products and services must be outstanding in terms of innovation, flexibility, and creativity. A prerequisite for being able to face current and future challenges is the systematic management of the knowledge assets. An advanced knowledge management requires what is called an organisational or corporate memory. It is the central repository of all the knowledge relevant for an organisation. Building up such organisational memories (OM) and making them available to people and application systems with quite converging needs is a big challenge which can only be met by an integration of approaches from various fields of computer science.

There are two major roles an organisational memory can in principle play. In one role it has a more *passive* function and acts as a container of knowledge relevant for the organisation (including meta-knowledge like knowledge about knowledge sources). It can be queried by a user who has some specific information need.

The second role an OM can adopt is as an *active* system that disseminates knowledge to users wherever they need it for their work. This second functionality is not just mere luxury but of considerable importance as users often do not know that an OM may contain knowledge currently helpful to them. Furthermore, querying an OM whenever the user thinks it might be possible that the OM contains relevant knowledge is not practical because the user does not always think of querying the OM when it might actually be helpful and because it would be too time consuming (as it interrupts the users primary work and takes time for searching and browsing the OM).

For the OM to be able to actively provide the user with the appropriate knowledge it needs to know what the user is currently doing. Unfortunately, this is usually not the case. How can this be achieved at all in a realistic way? In our view, the only practical approach to achieve this is to give people systems which help them do their work wherever this makes sense. These systems (partly) know what the user is doing and what kind of information she needs and thus can provide her (often implicitly, cf. [Cole et al. 97]) with relevant knowledge she may not be aware of as existing. These systems may be knowledge-based, i.e., have their knowledge explicitly represented in a knowledge base, or are based on document management systems with a keyword-oriented and/or free-text retrieval component. Note that this scenario is not primarily motivated by the idea of having an OM but to provide people with as much support as possible and meaningful. When such systems get installed in an organisation their knowledge bases begin to form an OM - so to speak as a side effect.

A closer analysis of the scenario described above yields the following implications and conclusions:

1. All knowledge bases of the knowledge-based systems used in an organisation should be part of its OM.
2. Certain parts of the OM only come into existence via knowledge-based application systems in the organisation.
3. Other parts of the OM do not fall under the above category. They represent knowledge that may be relevant for some user at a future time and can be queried whenever needed. This knowledge is not related with any application system in use and is accessed only by human users via the query interface of the OM.
4. The knowledge in the OM falling into category 2 is the formalized knowledge whereas the knowledge belonging to the parts of the OM mentioned in 3 is not or only rudimentary formalized. This is because the enormous effort required for an extensive formalization of knowledge is only spent when it is clear that it will indeed be extensively used. This is usually only the case with an application system that exploits that knowledge. When the formalisation process will become cheaper in future times (e.g., due to employment of automatic text understanding) this situation may gradually change.
5. A user not only needs to be able to query the less formalized knowledge in an OM but also the formalized ones. The shift in the degree of formalisation should be transparent to the user who accesses all parts of the OM in the same way via a uniform query interface.

From the conclusions above it gets clear that one of the main research problems to be solved is how to integrate the various pieces of knowledge into a coherent OM, and how to ensure its extensibility. Especially the following integration problems arise:

Integration of distinct knowledge bases:

From point 2 above follows that the knowledge bases of the various application systems must in some way be integrated to become part of the OM. This can be done physically by making one big knowledge base out of them, or virtually by coupling them via an overall framework. To make things worse, the knowledge bases are typically not disjunct. At least with respect to the terminology there is an overlap, and possibly also with respect to the represented business rules, office tasks, organisational structure, etc. This means that even without the aim of fully integrating these knowledge bases to one OM they should at least be integrated to such an extent that knowledge is not represented repeatedly, avoiding problems with maintenance and consistency.

Integration of several representations of the same knowledge with different degrees of formalization:

It may very well be the case that part of the knowledge also resides in the OM in a less formalized state - typically as semi-structured (hyper)text. For example, company regulations may be given

in the OM as text but also in a formal representation, e.g., for use by an intelligent workflow system. It should be possible to link all the more or less formalized versions of the same knowledge together such that different kinds of queries become possible. The query system decides which query to evaluate on which representational form(s).

In this paper, we outline a solution to solving both kinds of integration problems and to gradually building a comprehensive OM. As we have realized a knowledge-based system for supporting office work, called EULE2, that already comprises knowledge which should be part of an OM, it is important to have an approach that ensures the integration of various knowledge sources into an OM. We believe that this kind of situation, where certain parts of (not necessarily formally represented) knowledge that should go into an OM already exist, is quite typical. Thus, in the subsequent section (Sec.2) we give a concise introduction to EULE2, while Section 3 motivates the usage of a high-level representation language to build up and maintain the knowledge in EULE2. This high-level representation language then serves as a starting point for solving the integration problems mentioned above for building an OM (Sec.4). While EULE2 and the basic constructs of the high-level language are implemented the approach to integration is currently in a conceptual phase. Section 5 concludes the paper.

2 EULE2: A Knowledge-Based System for Supporting Office Work

At Swiss Life, as in many other companies, office workers for customer support are no longer specialists dealing with certain kinds of office tasks only, but are becoming generalists who must deal with all kinds of tasks. The work of this new generation of office workers is quite demanding and calls for a better support. For this purpose, the Information Systems Research Group of Swiss Life has developed a knowledge-based system, called EULE2, that aims at providing a user with a maximal guidance in performing office tasks she may not be familiar with.

An office task can be visualised as a graph (cf. Fig.1). Its nodes stand for (a sequence of) actions the user can perform, while its links are associated with conditions that must be fulfilled for the subsequent action to be permitted. The conditions result from the law and the company regulations. An office worker starts work with EULE2 by selecting an office task and entering task-specific data as requested by the system (EULE2 takes most of the data needed from various data bases and does not request it from the user). As long as the office task is not completed each action has one or more possible subsequent actions. From the data given EULE2 decides which path to follow in the graph. However, nothing is done automatically. The control of what to do next stays with the user but she cannot go on to actions that are not permitted. Some of the actions (like generating letters) are performed by EULE2 (possibly delegating it to another application system), the others are done by the office worker, telling EULE2 when they have been completed. Subsequent actions may be illegal, permitted, or obligatory. The office worker may decide to initiate a permitted action, may inquire why an obligatory action must be executed, or may ask why an action is illegal. Finally, the office worker selects an action for execution, thus causing new instances to be created or existing instances to be modified. This leads to a new situation where again one or more alternative actions are possible until a terminal node in the graph is reached.

When we regard the knowledge EULE2 makes use of as being part of an OM then EULE2 makes that OM an active system (with respect to that knowledge) in the sense as it has been mentioned in Section 1: In guiding the user through an office task the system supplies her with exactly that knowledge that she needs at a certain moment (cf. the idea of an “electronic performance support system” as discussed in [Cole et al. 97]), namely what to do next and why (the latter only if she is interested to know). Since EULE2 is a system that provides people with knowledge they need and ensures that every user always gets up-to-date knowledge EULE2 serves the purpose of knowledge

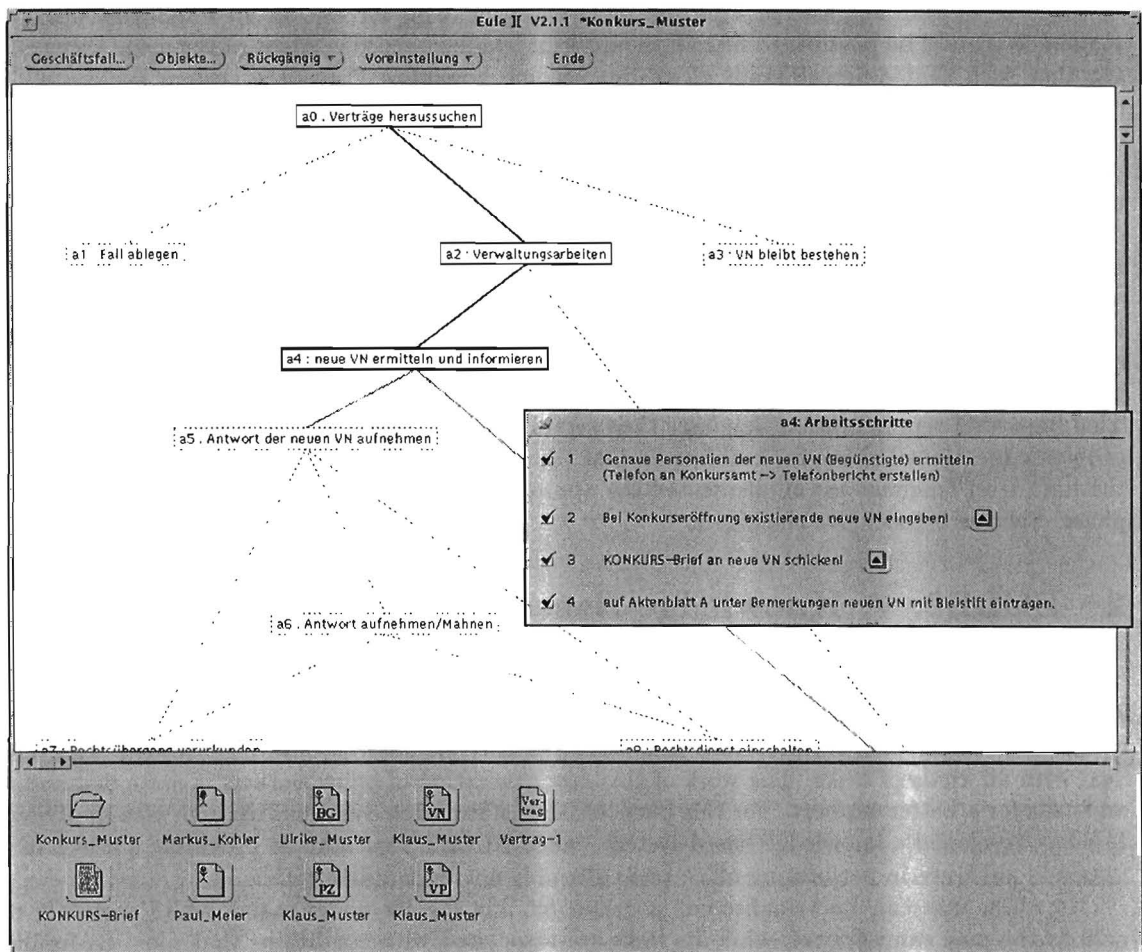


Figure 1: An Office Task Description as the User Sees it

management.

To achieve its functionality EULE2 requires the representation of

- the *office tasks*, mainly consisting of
 - a partially ordered set of actions,
 - for each of these actions the effects they have,
- the *instances* to be manipulated and the *concepts* they belong to,
- all the *laws and regulations* that must be obeyed by the office tasks.

Each of the three kinds of knowledge requires a representation formalism of its own. The knowledge about the office tasks is represented in a first-order language based on the situation calculus [McCarthy/Hayes 69], and knowledge about concepts and instances in a terminological logic [Woods/Schmolze 92, Reimer 85]. Knowledge about law and regulations is encoded in a syntactically restricted variant of first-order logic where we distinguish integrity constraints that must not be violated, and so-called auto-corrective integrity constraints which trigger corrective updates when they are violated. The latter are used like deduction rules with a complex condition

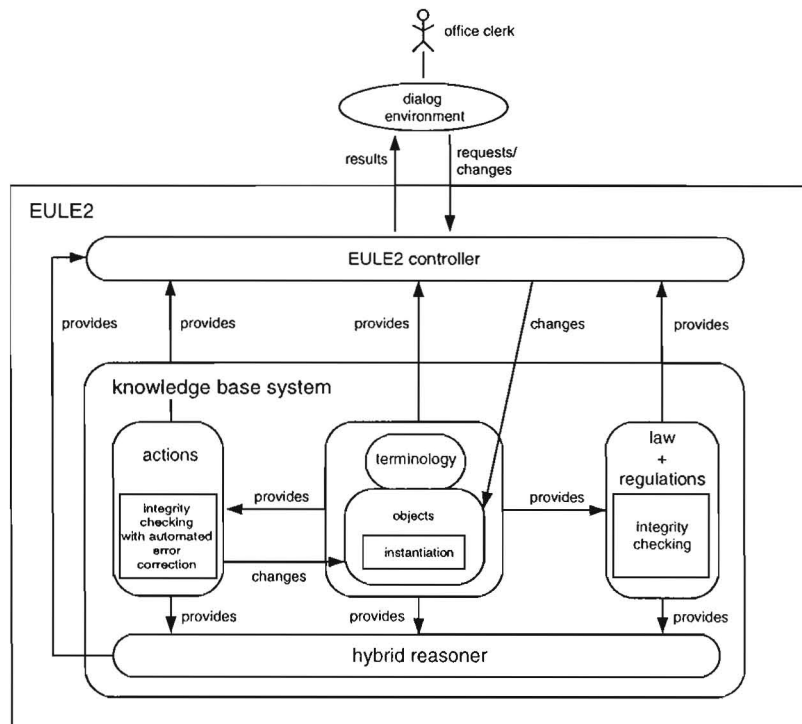


Figure 2: The Architecture of EULE2

part (cf. example in Figs. 3 and 4). Accordingly, the architecture of EULE2 (cf. Fig.2) provides for a knowledge base with three sub-components each of which offers its own inference services. As certain inferences needed for EULE2 require a combination of inferences of its sub-knowledge-bases they are integrated to a hybrid reasoning system.

EULE2 had to be integrated with several existing data bases where data needed for the office tasks resides. To this end we mapped the schemas of those data bases to a newly defined, integrated schema. Every relation schema belongs to a concept in EULE2's terminology while the relation tuples are seen by EULE2 as instances of the according concepts. Thus, for EULE2 it is completely transparent which concepts and instances come from one of the data bases and which reside in its terminology component only. The data bases are only read. Since no updates are made by EULE2 we avoid the problem of having long transactions with long locks. Data is changed by an office clerk in her usual way, namely through the application systems that already exist.

A further integration with workflow systems will probably become necessary in the future. We are currently investigating what kind of additional interface EULE2 would need for this. For more details on EULE2 see [Reimer et al. 97].

3 Employing a High-Level Representation Language for Modelling the Knowledge in EULE2

As can be seen, the EULE2 knowledge base captures quite some knowledge important to Swiss Life. Via EULE2 this knowledge is made available to an office worker in such a way that she gets always that knowledge offered which is relevant in the current situation. Besides for supporting office work, the knowledge EULE2 has available is also useful for other people and for other purposes, e.g.,

SchKG 232

1. *The Bankruptcy Office publicly announces the opening of bankruptcy proceedings as soon as it is established that due process of law must occur.*
2. *The announcement contains:*
 - ⋮
 4. *the call to those holding property of the bankrupt as pledgees or on other grounds to place this at the disposal of the Bankruptcy Office before the closing date, without any prejudice to their preferential rights, under threat of prosecution in the event of default and with the addition that preferential rights will lapse in the event of unjustified default:*
 - ⋮

Figure 3: Fragment of the Original Text of the Law SchKG 232

```

∀ t, s, b, bo, pb :
  instance-of(b, person, ⟨t, s⟩) ∧
  instance-of(bo, bankruptcy-office, ⟨t, s⟩) ∧
  instance-of(pb, publication-of-bankruptcy, ⟨t, s⟩) ∧
  related-to(pb, is-bankrupt, b, ⟨t, s⟩) ∧
  related-to(pb, by-bankruptcy-office, bo, ⟨t, s⟩)
⇒
∀ p, c :
  instance-of(p, property, ⟨t, s⟩) ∧
  instance-of(c, corporation, ⟨t, s⟩) ∧
  related-to(p, has-owner, b, ⟨t, s⟩) ∧
  related-to(p, hold-by, c, ⟨t, s⟩)
⇒
( ∃ o :
  instance-of(o, obligation-to-report-property, ⟨t, s⟩) ∧
  related-to(o, is-obliged, c, ⟨t, s⟩) ∧
  related-to(o, to-whom, bo, ⟨t, s⟩) ∧
  related-to(o, reason, pb, ⟨t, s⟩)
  AND
  related-to(o, concerned-property, p, ⟨t, s⟩)
)
```

Figure 4: Auto-Corrective Integrity Constraint Representing SchKG 232(2)(4)

for tutoring new employees, for inquiring about the effect of certain company regulations on office tasks, or for finding out about past instances of office tasks performed. Reusing the knowledge represented in the EULE2 knowledge base for other systems will be quite hard since the formalisms used have been selected to efficiently support the kind of reasoning that occurs in EULE2. Thus, they may not suit very well the purposes of another application. In order to facilitate reuse and to make building and maintaining the EULE2 knowledge base easier we are developing a high-level representation language (HLL) that abstracts away from the representation formalisms actually used in EULE2. In terms of the representational levels introduced in [Brachman 79] HLL is on the *conceptual level* while the EULE2 formalisms are on the logical level (only its terminological logic being on the epistemological level). Thus, the representation constructs offered by HLL already introduce certain fundamental concepts, like obligations, rights, regulations, and actions. They are especially tailored to the representational tasks encountered with developing EULE2. The move from the logical level of the EULE2 representation formalisms to the conceptual level of HLL introduces a representation ontology (like the Frame Ontology in [Gruber 92] – not to be confused with a domain ontology) which is reflected by the constructs of HLL. This ontology can and should be formally represented [Guarino et al. 94]¹. The knowledge modelled in HLL will be compiled down into the formalisms actually used in the EULE2 system. Due to its being on the conceptual level HLL offers the following advantages:

- There is no need to know the control flow of the inferencing in EULE2 as it is the case when modelling in the low-level representation formalisms of EULE2.
- HLL has an easy-to-understand, uniform syntax so that the knowledge engineer does not need to know the various representation formalisms used in the actual knowledge base.
- HLL abstracts away the state-space view so that the knowledge engineer does not have to deal with states and state transitions but with actions and their preconditions instead.
- A module construct supports reusing parts of an office task in other office tasks.
- There are syntactical constructs that mirror certain, very compact, natural-language formulations as they often occur in legal texts and in company regulations.

Thus, HLL, as it currently exists, is a domain-specific language because it only supports the kinds of knowledge needed for EULE2 and similar systems.

4 Making Use of the High-Level Language for Tackling the Integration Problem for an Organisational Memory

4.1 Integration of Distinct Knowledge Bases

As discussed in Section 1 there are two kinds of integration problems with respect to building an OM. One of them is the integration of the knowledge bases of several application systems into one physically or virtually integrated knowledge base which would form a part of the OM. The integration causes a considerable added value due to the following reasons:

- As parts of the knowledge in one system are often also needed for another system the integration avoids redundancy and all the follow-up problems of keeping redundantly represented knowledge consistent.

¹In fact, as [Guarino 95] suggests, the definition of the ontology underlying a representation language adds an additional, ontological level to the ones suggested by [Brachman 79]. It is situated between the epistemological and the conceptual level. In our case, the primitives used by HLL to make the representation of rights, obligations, etc. possible are formally defined on that ontological level while the result of their application, namely the fundamental concepts of rights, obligation, etc., are on the conceptual level.

- When part of the knowledge of an application system may also be of interest in future contexts that are not predefined it must be possible to access that knowledge via a general query interface where a user specifies her information needs. For that purpose the knowledge must not reside encapsulated in the knowledge base of some application system but must be generally available via the OM.

The integration of knowledge can only be achieved when it is represented either in the same language or in different languages that can be mapped to each other. Therefore, we intend to take HLL which has already been developed for EULE2 and extend it to a language we can use for representing the OM (the knowledge of EULE2 would then just be a small part in the OM). However, as the inferential requirements can be quite distinct for different application systems their knowledge can only then be uniformly described in one representation language if the language is on the conceptual level (rather than on the logical level), thus abstracting from the low-level representational views which reflect the measures taken for efficient inferences. This is the case with HLL. The extension of HLL to represent other kinds of knowledge as well pushes it more into the direction of a general-purpose language. Still, for a given application system only a certain subset is needed. By specifying the underlying representation ontology explicitly [Guarino et al. 94] the representational impacts of all constructs and their possible interrelationships stay clear (similar ideas underly the meta-modelling approach for customizing modelling languages as e.g. described in [Nissen et al. 96]). Due to such a formal ontology it is for example possible to have generalizations between constructs, like a general construct for representing actions with several specializations of it which serve the specific needs of representing actions in different application systems. The different constructs for representing actions may even be based on different conceptualizations of the world as long as the formal ontology keeps track of this so that a unified view can be generated (which is necessarily more general to capture all different conceptualizations). Different conceptualizations that are not on the level of constructs but affects how knowledge is actually represented can, of course, not be handled.

We think that the resulting language will not be bulky and monstrous because for a certain representation task only a subset is needed. The sum of all subsets which are properly put together via the underlying formal ontology make out the language. The development of such a language is still future work to be done.

4.2 Integration of Several Representations of the Same Knowledge with Different Degrees of Formalization

The second integration problem concerns the linking of representations of the same pieces of knowledge in notations that have a different degree of formalisation. We illustrate the need for doing this by the example of knowledge about company regulations which are represented in three different formalisations in the OM (cf. Fig.5):

- A regulation is represented in its original form as text, which is also used for its distribution (in Fig.5 the box labelled “regulation texts”).
- For each regulation there is a formal representation of what the regulation is about without representing its contents (in Fig.5 the box “content characterization”). This can be a set of manually selected index terms, or it may be a set of more sophisticated expressions in a terminological logic in which case it is typically the result of an automatic indexing (e.g., by using a text understanding system – cf. [Reimer/Hahn 97]). In any case, a fixed vocabulary (given by the underlying terminology of the domain) is used for characterizing the contents because otherwise a user would not be able to know what concepts to use for formulating a query.
- The contents itself of a regulation is formally represented (in Fig.5 the box “content representation of regulations”).

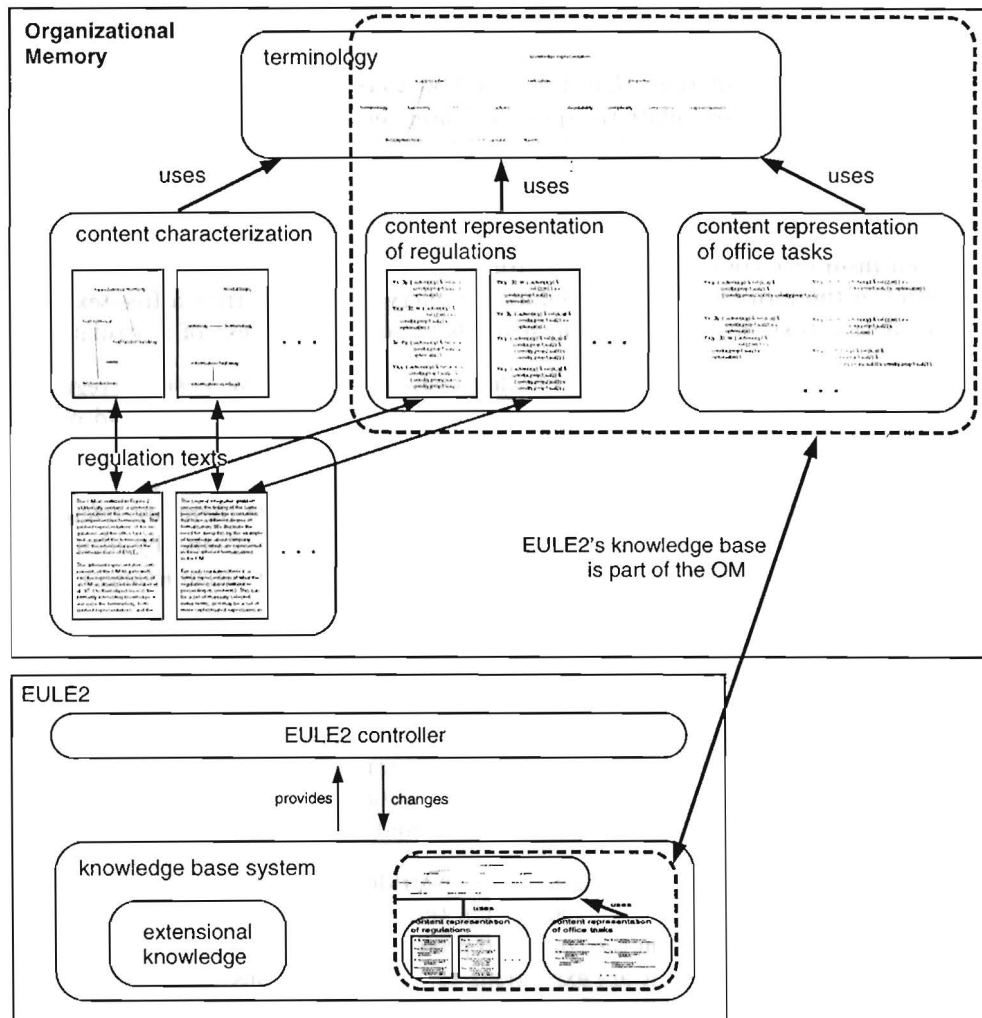


Figure 5: The Architecture of an Organizational Memory

The OM as outlined in Figure 5 additionally contains a content representation of the office tasks and a comprehensive terminology. The content representations of the regulations and the office tasks, as well as part of the terminology also forms the intensional part of the knowledge base of EULE2.

The different representation components of the OM fit quite well into the representational levels of an OM as discussed in [Abecker et al. 97]. On their object level is the primarily interesting knowledge, in our case the terminology, both content representations, and the regulation texts. The content characterization (of the regulation texts and thus, transitively, of the content representation of the regulations) belongs to their knowledge description level. The authors additionally suggest a relevance description level where the task-specific relevance of knowledge is represented so that it becomes possible to actively deliver exactly that knowledge which people need at a given time. This level has (currently) no direct correspondence in the OM architecture of Figure 5. However, the knowledge is implicitly present as part of the office task representations, but not independently on a meta-level.

A wide range of queries concerning company regulations can be posed to the OM. We give just a few examples (cf. with Fig.5):

1. The user looks for regulations that deal with how to react in the case of the bankruptcy of a Swiss Life client. To formulate the query the user selects concepts in the terminology given with the OM and sets relationships between them, resulting in a set of concept descriptions. The query is evaluated against the content characterization of the regulation texts.
2. The user looks for regulations that deal with certain underwriting issues (i.e., when to conclude an insurance contract, possibly with a risk supplement). No appropriate concept can be found in the terminology to formulate the query. Thus, she tries a free-text retrieval on the regulation texts by specifying which words to occur in the text of a regulation.
3. The user wants to know which kinds of office tasks are affected by a certain regulation. This query is evaluated against the content representation of the regulations and office tasks of the OM. This is a meta-inference on the content representations because the regulations are not used to find out if a given office task instance is to be executed in a certain way but instead the representations must be *inspected* to find out where there are references from a regulation representation to an office task. Such references are found by identifying which obligation the given regulation would deduce under the proper circumstances and to check which office tasks refer to this obligation in their precondition.

For some of the office tasks retrieved the user may then want an explanation in what aspects the regulation influences the way the task is to be performed. This request is satisfied with the help of the explanation component of EULE2.
4. The user wants to find out which regulations concern only one office task (maybe because she looks for possible ways to optimize the office work). This, too, requires a meta-inference on the content representations of the regulations and office tasks.
5. The user wishes to know which regulations override federal law (this happens in certain special cases where jurisdiction deviates from the literal interpretation of the law). Again, the query is to be evaluated on the content representation of the regulations. A similar query would ask for regulations that are exceptions to other regulations.
6. The user requests those office task instances of the last three years where a certain regulation was relevant for the way the office task was performed. This query is evaluated against the historized extensional knowledge base of EULE2 where the data of all formerly executed office tasks is kept.

The examples above illustrate that all three formalisations of company regulations are needed to evaluate all the possible queries. They show also that for the evaluation of one query more than one representation may be needed, for example, if a user specifies a regulation in terms that have to be evaluated against the regulation texts, and then, once the intended regulation is found, looks for regulations that are exceptions of it, which requires evaluation against the content representation. To the user it must remain completely hidden against which representation a query is evaluated so that she does not need to know to which representational form to pose the query nor to know all the query languages required. Instead, she always makes use of one and the same query interface. Consequently, there must be links between all the representational forms of regulations, as indicated in Figure 5.

We suggest to establish these links by exploiting a certain feature currently under development for HLL: With respect to law and company regulations it maintains a one-to-one correspondence between the knowledge represented and its original natural language formulation. This correspondence is on the level of subsections when law is concerned, and on the paragraph level for regulation texts. As HLL provides special constructs for certain kinds of complex natural language phrases

(like "... pursuant to...", "... except of...", "... analogously to...") the one-to-one correspondence is in these cases on the sentence level or even below. Having a one-to-one correspondence on the sentence level or below is in general not possible as a certain piece of knowledge may be described in more than just one sentence.

The one-to-one correspondence not only helps the knowledge engineer but is a necessary prerequisite for generating explanations that use phrases of the original text so that the user can more readily see the correspondence of a restriction she encounters in the office task with a certain law or regulation. An ordinary, first-order representation of law and regulations does not allow such kinds of explanations because a first-order representation usually atomizes the statements to be represented so that no correspondence can be seen any more.

HLL's property to link a regulation representation with its textual counterpart can be exploited to keep track of the dependencies between a content representation and a natural language text which describes (part of) the same knowledge. In this way, later changes to the text or the content representation cannot be done by a knowledge engineer without his taking notice of the dependency. Of course, it can by no means be ensured that the text and the formal representation are consistent with each other because that would require a degree of text understanding abilities currently far from being feasible.

5 Conclusions

We have outlined an approach to creating an OM by integrating the knowledge bases of existing application systems as well as those to be built in the future. To support the integration we advocate to employ a high-level representation language HLL which is used to represent the knowledge in all the knowledge bases. HLL has to be on the conceptual level (according to [Brachman 79]) because only in this way it can abstract from the lower-level inferential commitments made to achieve efficiency. Although HLL would be quite a comprehensive language, for a certain application only certain constructs are needed. The relationship of the various constructs is to be given by a formal ontology so that a mapping between constructs is possible (where meaningful). In this way, we can even support different conceptualizations of the world in different knowledge bases while still using the same representation language.

An HLL representation is compiled into the actual representation formalisms used in the knowledge bases which are usually quite different for the sake of efficiency. The compilation can be different for the various knowledge-based systems.

We have also addressed the need of having the same pieces of knowledge in more than just one representation in the OM. The representations differ in the degree of formalisation, ranging from natural language text to deep, first-order representations. These representations are needed to answer the various kinds of queries that may occur. A query interface to the OM must hide from the user what kind of query is evaluated on what kind of representation. To enable the query system to pick out those representation which is the proper one for the current query the textual and more formalized representations must be linked to each other. We suggest that this is done with support by an extension of HLL, too, which maintains a one-to-one correspondence between pieces of a natural language text and a formal representation of that text.

The first version of HLL is currently being implemented for a system to support office work we call EULE2. Further extensions, especially concerning the support of certain complex natural language phrases, are planned. The ideas of using HLL for building an OM are still preliminary and need further elaboration. This will be done in the context of a new project which has the aim to integrate quite different knowledge sources so that they can be queried via a single user interface. This integration effort is intended to become the starting point for Swiss Life's OM.

Acknowledgements:

I am grateful to my colleagues Jörg-Uwe Kietz and Martin Staudt for their helpful comments

on an earlier version of this paper. I also got a lot of comments from the reviewers which, too, helped considerably to improve the paper.

References

- [Abecker et al. 97] A. Abecker, A. Bernardi, K. Hinkelmann, O. Kühn, M. Sintek: Towards a Well-Founded Technology for Organizational Memories. In B.R. Gaines, R. Uthurusamy (eds): Artificial Intelligence in Knowledge Management. Papers from the 1997 AAAI Spring Symposium. Menlo Park: AAAI Press, 1997.
- [Brachman 79] R. Brachman: On the Epistemological Status of Semantic Networks. In: N.V. Findler (ed): Associative Networks. Academic Press, 1979, pp.3–50.
- [Cole et al. 97] K. Cole, O. Fischer, P. Saltzman: Just-in-Time Knowledge Delivery. In: Communications of the ACM, Vol.40, No.7, 1997, pp.49–53.
- [Guarino et al. 94] N. Guarino, M. Carrara, P. Giaretta: Formalizing Ontological Commitments. In: Proc. 12th National Conf. on Artificial Intelligence, 1994, pp.560–567.
- [Guarino 95] N. Guarino: Formal Ontology, Conceptual Analysis and Knowledge Representation. In: Int. Journal of Human and Computer Studies, Vol.43, No.5/6, 1995. (Special Issue on the Role of Formal Ontology in the Information Technology)
- [Gruber 92] T.R. Gruber: A Translation Approach to Portable Ontology Specifications. In: Knowledge Acquisition, Vol.5, 1992, pp.199–220.
- [McCarthy/Hayes 69] J. McCarthy, P.J. Hayes: Some Philosophical Problems from the Standpoint of Artificial Intelligence. In: B. Meltzer, D. Michie (eds): Machine Intelligence 4, 1969, pp.463–502.
- [Nissen et al. 96] H.W. Nissen, M.A. Jeusfeld, M. Jarke, G.V. Zemanek, H. Huber: Managing Multiple Requirements Perspectives with Metamodels. In: IEEE Software, March 1996, pp.37–48.
- [Reimer 85] U. Reimer: A Representation Construct for Roles. In: Data & Knowledge Engineering, Vol.1, No.3, 1985, pp.233–251.
- [Reimer/Hahn 97] U. Reimer, U. Hahn: Text Summarization Based on Condensation Operators of a Terminological Logic. In: Proc. ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization, Madrid, July 7–12, 1997.
- [Reimer et al. 97] U. Reimer, A. Margelisch, B. Novotny: Making Knowledge-Based Systems more Manageable: A Hybrid Integration Approach to Knowledge about Actions and their Legality. In: R. Pareschi, B. Fronhöfer (eds): Representing and Managing Dynamic Knowledge. Kluwer, 1997. (to appear).
- [Woods/Schmolze 92] W.A. Woods, J.G. Schmolze: The KL-ONE Family. In: Computers and Mathematics with Applications, Vol.23, Nos.2–5, 1992, pp.133–177.

Processes of Knowledge Preservation: Away from a Technology Dominated Approach *

by: Kai Romhardt

Input-Paper for the 21st Annual Meeting of the German Society for Artificial Intelligence

Workshop 01:

"Knowledge Based Systems for Knowledge Management in Companies"

9-12 September '97 in Freiburg/Breisgau

Kai Romhardt (lic.oec.HSG)
University of Geneva/geneva knowledge group
Hadlaubstrasse 110
8006 Zürich
Switzerland

Tel./Fax.: ++41-1-361 45 63
E-Mail: romhardt@msn.com

**This article is a building block of a knowledge management concept the author developed with his partners at the University of Geneva. Compare Probst/Romhardt (1997), Romhardt/Probst (1997) or Probst/Raub/Romhardt (1997).*

1. Processes of Knowledge Preservation: Away from a Technology Dominated Approach

voices of
managers

*"In our R&D center we have a small group of absolute product experts. The most skilled and most respected retired some days ago. We are sure that a good part of our product expertise will be lost but we do not know how to save his experiences for the future."
(R&D manager of a food-company)*

*"Some months ago I realized what it means to work for a company that knows how to protect valuable knowledge. I participated in a presentation of a younger colleague and saw him flipping charts that I had produced some time ago. The charts had transformed into company knowledge and the speaker had no idea of their origin."
(Partner of a consulting company)*

*"In our company a multitude of project groups on different levels are trying to create an electronic memory for their area of responsibility. But an integrated solution for the whole organization is missing, which will cause interface problems in the future. I fear that if we have reached the end, there will again be only a fraction of our experience at the fingertips of our employees".
(IT manager of a big service company)*

The special importance of an organizational memory has been stressed by many management thinkers in the last years, but in most management concepts the conscious handling of the organization's past plays secondary role. In general, we can describe memory as a system of knowledge and capabilities that preserves and stores perceptions, actions and experiences over time and secures the possibility of recall for the future.¹ The organizational memory is the crucial point of reference for new experiences: without memory, learning is impossible.

underestimation
of experience

The preservation of knowledge is an important building block within the concept of knowledge management. The value of the organizational memory is today particularly underestimated in the process of reorganization. The following statements of managers seem to be typical: "We have to become leaner." "We have to build up a younger staff." These arguments often prepare the way for radical outsourcing or downsizing activities. "Ballast" is discharged, but do we talk about well thought out or negligent separation from our own (out-dated?) experience?²

retrievable
issues

The dismissal of employees unwilling to change may break a blockade but always at the price of losing personal know-how and expertise. Many companies made the bitter experience that through *lean management* and the unavoidable discharges and outsourcing activities valuable know-how left the company and had to be bought back by hiring expensive external consultants. Certain company-specific skills, e.g. the architecture of old industry plants, were lost forever. The loss of certain critical information may reduce the performance of entire company departments.³

the memory of
the company

Let's take the case of Andy Miller. Andy has been working for the last thirty years in the sales department of a big American trading company. Everybody knew him as the 'good soul' of the hundred employees in the department. Most of his time was spent in informal talks and in this way he talked daily to nearly every salesman. A change in management induced a detailed analysis of the individual sales performance for which an external consultant was hired. Andy Miller, who never had sold very much and was aged over fifty, got his dismissal at the end of the quarter. The personal comment of the consultant: "Miller could seldom be found in his place and spends most of his time in non-sales-related chats." After Andy's dismissal problems never experienced before started. The coordination of sub-unions did not function as well as before. Responsibilities that seemed to be clearly defined became unclear and the number of customer complaints started to rise. The general mood changed. There were various complaints. Anniversaries and birthdays were forgotten, new employees felt poorly supported and broke unwritten rules of the company. Day by day it became clear that with Andy's dismissal the "memory of the company" had also been dismissed. He was someone who had built up detailed knowledge about people and processes within the organization and he had shared that with others during his 'unproductive' chats.

This case underlines that without intentionally keeping experience, unexpected losses may occur. Man is a creature deeply rooted in his evolutionary history and he creates his own identity by referring to his past experiences and using his unique ability to learn. Indeed, many companies complain that they have lost a part of their corporate memory in the process of reorganization. This *collective amnesia* is often the result of the non-reflected destruction of informal networks which steer important but seldom observed processes. Consultants have already named this disease *collective Alzheimer syndrome*, a phenomena which can be found particularly in shrinking companies.

*'unlearning
versus protecting*

Management theory discusses the tension between the destruction and protection of old skills, capabilities, and information mainly under the topic of *unlearning*. Hedberg defines unlearning as the process in which learners discard their old knowledge⁴. This definition requires a rigid separation from past knowledge that burdens the carrier. This separation is necessary for a new start. Organizational unlearning has to start if the actual patterns of interpretation and reaction or if the organizational theory in use no longer fit the changing challenges of the organizational environment. The problem is in the selection process between knowledge that is no longer needed and knowledge that is absolutely necessary for the future. In this logic, unlearning means to question one's own routines and to let go of things we are used to.

*xperience as a
starting point for
improvement*

Should all of our existing customer data be deleted because our marketing did not function well in the past and has to be reorganized? Absolutely not. Should successful teams be disbanded, because they worked on the wrong questions? Absolutely not. Should we dismiss all our employees who have reached a certain age because they can be categorized as too inflexible for future changes and because their early retirement will be subsidized by public institutions? Absolutely not. Purposeful protection of critical data and information is of enormous significance for every organization. The evolution of the organizational knowledge base is only possible in reference to existing knowledge. Individual psychologists think that old experiences can not be overwritten by new knowledge and therefore cannot be deleted. Instead, old rules are marked 'obsolete' and do not become operational under changed circumstances. But, they still exist as an optional path of action that widens the scope of action of the organization in a turbulent environment⁵.

*rocesses of
nowledge
reservation*

Organizations that would intentionally manage their experiences in order to have them on call for the future have to master three basic processes of knowledge management. They have to select out of the large number of organizational events, persons or experts and processes only those that are *worth being preserved*. They should be able to *store* their experience in a suitable form and, as the last step, ensure the *actualization* of the organizational memory.

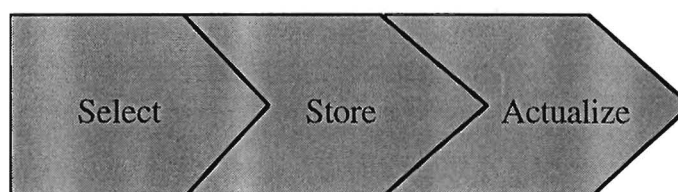


Chart 1: The main processes of knowledge preservation

1.1 Selection of valuable knowledge and information

inciple: selection

Every organization gains new experience day by day that may be useful in the future and therefore should be protected. Project reports, meeting minutes, letters or presentations emerge from a variety of places. Every day customers come up with complaints and problems but also with ideas and praise. It is impossible to keep track of all these organizational events. Let's take the example of a salesman who - the same as his colleagues - often presents his products to business partners. This salesman has produced a sales presentation which visualizes the product advantages much better than present sale tools. That his colleagues do not know anything about his presentation may be a problem of insufficient knowledge identification processes within the company or it may have its roots in inadequate communication. Perhaps no one ever offered any sharing-incentives to our presentation professional. The perspective of knowledge protection asks what is going to

happen if the salesman leaves the organization tomorrow? Who will find his central documents or presentations on his better or worse organized hard disk? Are the most important processes and contact persons documented? In many cases, the unexpected departure of an employee leaves a painful hole because of insufficient documentation during the time of employment. As documentation means work and money, an investment that rarely pays in the short run and rarely brings a reward to the writer, we need selection rules. It makes no sense to document everything. We cannot and should not keep everything. The challenge lies in the selection between protection-worthy and not-protection-worthy knowledge entities. We have to transfer valuable data, information and skills into organizational systems in which they can be used by the whole company. An ARTHUR ANDERSEN system is a good example:

ARTHUR ANDERSEN

Systematic knowledge protection and selection by ARTHUR ANDERSEN ONLINE

The internal information system of ARTHUR ANDERSEN is the backbone of this knowledge-sensitive organization. Electronic discussion groups exist in all areas that are important for their consulting know-how. The quality of contributions to these news-groups is extremely volatile and many quotations are out-dated very quickly and become worthless.

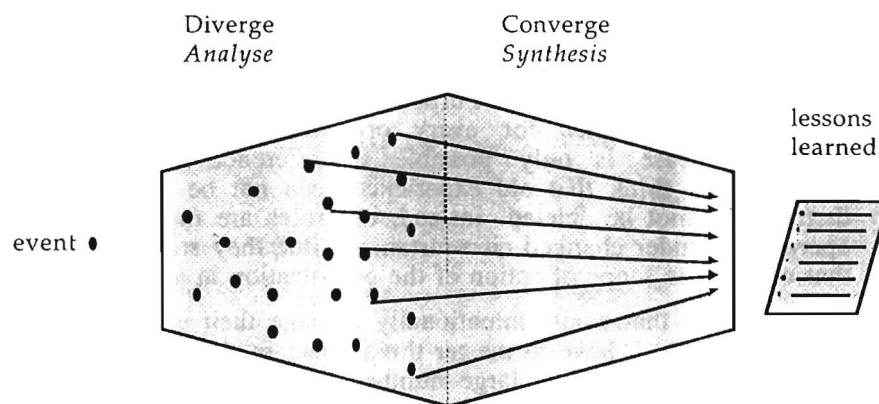


Chart 2: From a Divergent System towards a Convergent System (source: Arthur Andersen)

The challenge lies in analyzing the single information bit of this divergent system. ARTHUR ANDERSEN defined clear responsibilities for this analysis and selection process. Every organizational center of competence (e.g. total quality management) has a manager or a team that is in charge of this process. Supported by an interactive knowledge creation process, single contributions and documents are condensed into central documents. For example, twelve experience reports concerning the introduction of Lotus-Notes are summarized in one master-document, which sums up the central *lessons learned*. This synthesis enables the reduction of the information load and ensures that only relevant topics or reports are in the system. The user does not have to read all the single reports and saves time. Typical products of these converging processes are *best practices*, *best firms*, *presentations*, *process definitions*, *studies*, *articles* and *impact-analyses*. These are stored in a structured way by ARTHUR ANDERSEN ONLINE and are accessible for all members of the organization. The knowledge managers are responsible for the clean up in their center of competence. Without the deletion of irrelevant or out-dated information, the world-wide databases would grow at a rate of three percent daily and would soon become dysfunctional.

parallels with the
human brain

Organizational selection and storage activities are comparable with processes within the human brain. To make an impression on our long-term memory an information has to pass the filters of *ultra-short-term* and *short-term memory*⁶. These gatekeepers of long-term memory divide relevant from irrelevant perceptions and in this way protect the brain from permanent overstimulation. Unfortunately, the conscious part of our psyche can influence this process only in a very limited way. The consequences are various learning tricks and techniques to outwit the gatekeepers and to filter valuable bits out of the passing stream of information.

Organizational routines of knowledge protection	On the organization level comparable problems exist. Not all selection mechanisms are planned in a systematic way. Organizational routines - such as the filing of a certain document type - take care of the automatic execution and conservation of certain processes. Nearly every office has its file cemetery or dusty archives that represent the wrong way to deal with knowledge and information preservation. In these areas, routines are anchored very strongly and employees who are in charge of the concerned systems will not change their behavior without being pushed.
Knowledge documents	Therefore, organizations will never be able to manage all the processes of knowledge selection. That also would not make sense. But for core areas of the organizational knowledge base (e.g. knowledge about customers) efforts of purposeful selection and documentation should be taken. The materialization of this knowledge (the part that can be made explicit) in knowledge documents as <i>knowledge maps</i> or <i>lessons learned</i> separates experience from the individual and secures its possession by the organization for the future. ⁷ Crucial for this process is the concentration of knowledge on certain core points and strictly relating it to observed problems. Only knowledge that will be usable for a third party in the future are valued to be protected and saved for the future. The rest only steals time and trust in the quality of the documentation system. Sometimes less is more. Established documentation systems should be seen through this perspective and be tested on their right to exist. Nevertheless we have to take into consideration that only a small part of our future information needs can be estimated, which demands a not-too-rigid selection approach.
Documentation of success stories	Another chance to gain knowledge about an organization's past is to illustrate leading ideas as <i>leading principles</i> , <i>models</i> , <i>stories</i> or other <i>symbolic forms</i> . These storage media are very helpful because of their potential to gain quick access to thinking patterns within the organization. ⁸ For example a Swiss retailer engaged a consultant to document the process of a very successful introduction of a new product line. By interviewing all the important actors within the strategic process and the formulation of specific key factors of success, the internal success story was reconstructed. The findings of this study were summarized as a case study and can today be used for training. The case shows that a positive and successful event in a company's history can be used for motivation and knowledge transfer.
Identify key-employees	New technologies like <i>workflow-management</i> or <i>document-management-systems</i> open a new dimension for the protection (and surrender) of organizational knowledge. Despite these technological options, it's still people who make good or bad selections. Employees like Andy Miller cannot be replaced by machines or computer systems. Their experience is the key for a useful organization of the company's past. These key employees have to be identified and their continued presence in the company assured. This is the most secure preventative against collective amnesia.

10.2 Storage of knowledge

After separating the protection-worthy part of organizational knowledge from the less important entities, we have to store it in the organizational knowledge base in a suitable form. We distinguish three forms of storage: *individual*, *collective* and *electronic storage of organizational knowledge*. As these three storage processes follow different logic we will discuss each separately.

10.2.1 Individual storage or "Who still knows about...?"

Unfixed assets	Whether caused by lay-off and termination, retirement or death, organizations permanently lose valuable knowledge kept by these individuals. The <i>Economist</i> named these outstanding employees with a ironic wink, <i>unfixed assets</i> . This is the exact problem of a company's relationship to its <i>knowledge workers</i> . Knowledge that is anchored only in the head of the employees is of volatile quality. Once losses occur, they can only be replaced with high investment and incalculable side-effects. The easiest way to save intellectual capital is to create an atmosphere that does not stimulate thoughts of changing companies.
----------------	---

incentive systems
and barriers to
exit

If top-performers like their social environment, they will be less open to lucrative offers from competitors. But, top-performers have quite different motivational structures. Knowledge about these motivational structures that may offer the right incentives can be gained by careful listening. If we think that an excellent working atmosphere added to an average income, is sufficient for the long-term commitment of an employee, we will most probably lose some of our best experts. *Exit barriers* may be created by *social* or *material incentive systems*. In order to be effective, they have to take the personal needs of the employee into consideration.

flexible holding-
mechanisms

In many cases it will not be possible to create sufficient exit barriers. Many qualified top-performers who have their own ideas, cannot stand big companies permanently and risk the jump into independent work. The establishment of *flexible cooperation* with these alumni is a rewarding option to preserve the access to their know-how after the termination of their contract. Alternative forms of cooperation are operations as trainers, consultants and selective cooperation in difficult talks with old customers and more. The guiding principle is the creation of *win-win-situations*. Consulting firms with annual fluctuation rates near thirty percent use the permanent exodus of skilled employees for the establishment of a strong network. They have 'their people' in nearly all important companies and they secure exclusive access to important information. The clever management of relations with retirees is a neglected field. One might get the impression that the 'elders' have no value at all. Their exit of the organization is often abrupt and communication stops. An investment in a relationship with retirees enables access to information and customer contacts that would, without them, be lost to the organization forever.

ABB CONSULTING AG

Storage of experience from 'gray managers'

Before founding ABB CONSULTING SA, the Swiss management of ABB was facing a dilemma. On one hand, early retirement was a necessity to make room for younger promising talents. On the other hand, ABB did not want to lose their highly experienced and well connected managers. In addition, older managers wanted to influence when and how they retired in a more flexible way. The founding of ABB-CONSULTING SA facilitated this retreat in installments. This company employs today only former top-managers of ABB Switzerland that want to start a new career as a consultant at the age of sixty. These *gray consultants* act mainly in daughter companies of ABB and profit from their world-wide network as well as their industrial experience.

The consultants of ABB CONSULTING work in various fields of consultancy such as short-term management, public relations or cooperative projects with state or other official parties. The knowledge of the old foxes can be applied to many areas. Their experience provides ghost-writing for top-management, temporary leadership of a production plant they know from their active corporate life, or as an experienced process coach in complex projects. While regular and sometimes forced retirement normally cuts the knowledge transfer between employer and employee, ABB found a flexible solution that works for both sides. The company has extended access to the know-how of their former employees while the gray consultants get personal and financial recognition of their expertise. By the way, the fees for expensive external *outplacement* consultants can be saved.

systematic
transfer of
capabilities

Another option to preserve critical capabilities is the continuous training of the 'successor'. The successor should be introduced step by step into his future tasks to guarantee knowledge transfer in critical areas. In many European companies current position holders try to defend their knowledge monopoly till the last day in order to strengthen their power position. The Japanese have different management principles.

sempai-kohai

One principle is called *sempai-kohai* and expresses the close relationship between a male pair that is built of an older, teaching *sempai* and a younger *kohai* who has to be instructed.¹⁰ Every 'rookie' is assigned to an older mentor (who is sometimes only a few years older) and who transfers the necessary tricks to the younger generation. The relationship between *sempai* and *kohai* is strengthened by shared leisure activities in a systematic way so that trust can be built for the free-flow of information of all kinds. The transfer of implicit knowledge is also supported by this personal relationship. In 1993 when the Japanese steel industry discharged 25 percent of its 150,000 working force, many observers feared a radical decline of the average qualification. But the decline in quality did not happen, as *sempai-kohai* ensured the quick transfer of knowledge within the shrunken work force.

purposeful
explicitation

However, the damage caused by the loss of experts can be reduced with simple tools. One of the easiest ways is the implementation of *structured exit-interviews* that must be conducted by trained experts. In these interviews, the critical knowledge of the person (special documents, contacts, project engagements) is made explicit and documented. These talks should be coordinated with the successor. Led in an open, positive atmosphere, these interviews may capture much valuable information and learning references about the organization. One gets to know why the expert decided to leave the company and may get ideas for the adaptation of actual exit barriers. SANDIA NATIONAL LABORATORIES in Albuquerque, New Mexico, documents these interviews on video.¹¹ This is their method of capturing the wisdom and know-how of resigning scientists and of keeping their expertise in the memory of their organization.

10.2.2 Storage in the collective memory

The human memory is transient and not static. Psychologists claim that we change or rewrite our own past with every act of remembering it. The problem with remembering is that 'wrong' memories feel exactly like 'right' memories.¹² In order to avoid a complete confusion between self-constructed reality¹³ and his 'real' past, man needs feedback of third parties who confirm his own picture or force him to adapt it to their perceived version. That's how groups or other collective entities become the regulators of their own experience. Collectives store shared experiences differently than individuals. For example, psychologists can identify the borders between differently involved regions of the 30-years-war by conducting "depth-interviews". The horror of these times is burnt deep into the collective memory and after three hundred years still influences the daily behavior of the descendants.¹⁴ What is the link to knowledge management and knowledge protection? It is the proof that historical experience is deeply rooted in organizations, although it is often missed by a quick observer.

collective wins
over individuals

Collective storage can be seen as historical ballast, but it may also be very productive. This can be illustrated by a laboratory experiment in which single persons and a closed group were trained in the construction of transistor radios.¹⁵ One week after the training, the people were reunited and asked to remember and execute the construction steps they learned. Individually trained people were put into small groups while the 'group-trained' people stayed together and tried to remember collectively. The results were analyzed and showed that the group-trained could recall more details and constructed better radios. Detailed video analysis showed that during the initial training a series of social and cognitive relations were established which the researchers called *transactive memory system*¹⁶. This collective memory was superior to that of individuals.

extension of the
memory

Another phenomena of collective storage was gained through the observation of pairs¹⁷. People use certain other people with whom they interact intensely as extended memories. This is a way to increase their own memory capacity. They develop a feeling for which details of a shared social situation will be remembered very good (e.g. names). This division of memory has the funny effect that two partners can only remember a social situation exactly if they are together.

documentation
of important
processes

Many team success stories cannot be explained by these observations. Group processes or collective problem solving processes have their own self-dynamic and are difficult to understand for external observers, or even the members of the group. Despite these difficulties, much can be done to capture important processes for the future and to gain starting-points for improvement. Employees of the Japanese-American joint-venture FUJI-XEROX documented every important step and detail of the cooperation process.¹⁸ Through this, future employees of the company should have the opportunity to learn from and about the past.

taking minutes

The most traditional form of documenting a meeting are the *minutes*. But good minutes and the people who can produce them are rare, and, this activity has often been seen as a tiresome duty. The result of this low regard are too short or too long, redundant, badly-written, poorly structured, full of gaps, late-arriving annoyances in paper form. But for organizations in which most employees work in fast-changing project teams, minutes become a central source of collective storage of the actual project status. Japanese companies train their moderators specifically in documentation techniques. They want to

guarantee that experience and decisions do not fade away or are lost. By studying the existing minutes, new group members can learn about the actual status of group discussions.

importance of
shared language

More powerful than *writing* is the *spoken word*. It offers the biggest option to build and anchor collective experience. The spoken word is closer to us than the written word. Over time, companies create their own vocabulary, which newcomers have to learn. This vocabulary is much more than a collection of the usual efficiency-creating *abbreviations*. Normal words like *quality*, *change* or *safety* are defined in a firm-specific way and reflect the company's past. These terms are often linked to strong emotions. An industrial company had suffered for a long time under a highly-paid consultant team that had conducted a failed reorganization project called „horizons“. Years later, the simple mention of the word "horizons" easily killed any new attempt to bring consultants into the organization. Every organization knows terms like this, terms that are permanently embedded in daily actions. Knowledge management has to try to anchor its central ideas in the organizational language and use this power for its purpose.

collective
formation of
concepts

One possibility to anchor and store central ideas is the process of *collective formation of concepts*¹⁹. By conscious questioning of central terms at the beginning of a group process 'clear terms' can be picked out as a central theme. The collective deconstruction and redefinition of terms reduces the danger of future misunderstandings. Especially buzzwords, like *total quality management* or *process reengineering*, often stay misunderstood. A clear vocabulary avoids confusion.

shared
experience

The integration of ideas is a complicated issue in decentralized or heterogeneous companies. Investments in shared language or shared experiences can become enormous. At SWISS TELECOM, 20,000 employees participated in a so called "mind change workshop" as part of the reorganizing process of the whole company. In this workshop, the participants can experience 'change' in small groups and may relate this change experience to their personal situation. All employees see the same video clip about insufficient customer orientation and discuss this problem in their working groups. Shared experience emerges that may be transferred to the working place and change daily behavior. The "Change TELECOM" project that is shaping all organizational structures and strategies can be 'felt' and anchored in the organizational memory.

10.2.3 Storage in electronic memory

unlimited storage
capacities

IT-revolutions have multiplied electronic storage capabilities. If this development continues we will soon have virtually unlimited electronic storage capacities that cost very little.

digitalizing

Nearly all traditional storage media can be digitalized. Video cassettes can be replaced by CD-ROM's, documents can be scanned and digital cameras are already in the market. The average future computer user will soon have access to all forms of storage media under one surface. The qualitative difference of digitalized storage media comes from their easy editing and reuse, as well as their cheap and fast distribution over networks. With rising digitalization the base of the global digital memory of mankind is growing. The Internet gives more and more users access to these data masses. The Internet and intranets of today are just the beginning of a development that is unpredictable even for experts. If libraries, magazines, sound, movie, and text archives grow together and widely accepted standards for the organization and structuration of the digital raw materials emerge, the Internet will become a meta-archive for everything.

consequences for
the
organizational
memory

This development has dramatic consequences for companies which act in knowledge sensitive environments. First, they have to anticipate that their competitors have principal access to the worldwide datapool and use it for its purposes. Second, they have to look at their own electronic knowledge base. In knowledge-intensive companies, an important part of their know-how is stored in digitalized documents like presentations, forms, blue-prints or reports. Their systematic filing and re-use becomes a growing competition factor.

memory loss

There are many reasons why access to the electronic memory may shatter. If for example the knowledge documents of a personal computer were not fed into the system, they are not available for the electronic memory of the company and its employees. If the documents were codified incorrectly or filed in the wrong place, they may never be found again and are lost (perhaps for ever). If a codification can not be interpreted or if a network or single

computer is not connected to central databases, the organization will not remember that this document exists.

Only a few companies organize their electronic memory consistently and are able to master these difficulties. Most organizations fight against their own information systems and data structures that hinder the establishment of efficient and user-friendly *corporate memories*.

'ontrolled
ocabulary

Another problem is the introduction of a *controlled vocabulary* that codifies documents with a well-defined set of key-words. This corporate language may be built by a selection and definition of central terms within the company and permits a later classification of documents to central action or knowledge fields of the organization. The disadvantage of this solution lies in the high investment for the establishment and maintenance of the vocabulary.

'onnections

The intelligent connection of documents is the key factor for success in the establishment of a powerful electronic memory. Similar to the neuronal structure of the brain that captures memories in connection structures, a good IT system may lead the way to experience that has nearly disappeared. The chaotic connection structure of Internet and intranets is both an advantage and a problem for the systematic storage of organizational experiences. More and more hypertext-links lead to nowhere because their reference side does not exist anymore or has been changed.

10.3 Actualize and remember

importance of
ctualization

The conscious and structured storage of organizational knowledge is not the end of the knowledge preservation process. The organizational memory is operational only if desired information can be remembered in acceptable quality. Beside sufficient selection and data-sensitive storage, we have to pay attention to actualization processes. Companies may lose money if they base investment decisions on out-dated or incorrect knowledge. At ARTHUR ANDERSEN the leaders of competence centers are responsible for the up-date of their databases, documents and discussion groups. So the uncontrolled growth of databases can be avoided and users do not have to suffer under old or irrelevant information.

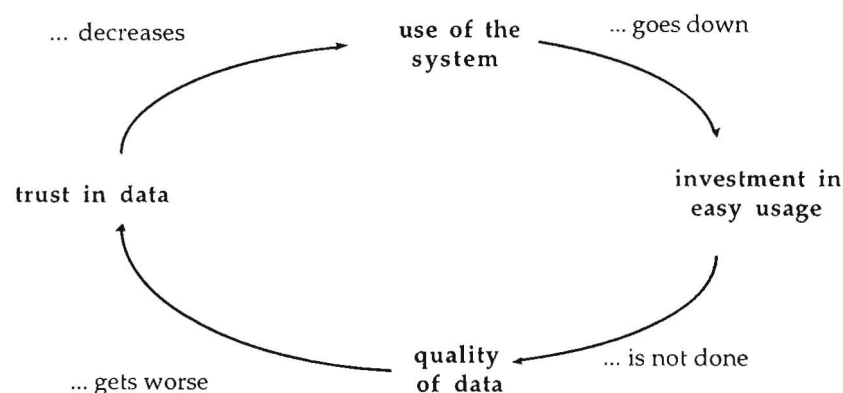


Chart 3: The Death Spiral of an Electronic Knowledge Base²⁰

'eath spiral

If actualization processes cannot be controlled, knowledge systems may easily get into a death spiral. In managing their memory, companies have to solve mainly *trust* and *access* problems. If trust in data quality exists and easy access to the system can be guaranteed, systems will be fed and will be used in ways that increase data quality. If the actual database is already full of mistakes, trust cannot be built and no-one will invest much energy into the system. Data quality decreases further and the system dies. In times of short half-life death may arrive very quickly.

'rganizational
nesia

The biggest problem for the organization memory is corporate amnesia. Typical statements of managers are: "We once knew about it, but now we seem to have forgotten it." There are many ways that organizations lose their memories. Employees walk away, excellent teams split, data-bases become infected by a virus or whole functions are outsourced. These

actions reduce collective memory. Sometimes access to the memory is simply blocked, sometimes forever. Examples are the permanent or limited overload of experts or the low motivation to share experiences with others. Equivalent memory barriers exist on the electronic and collective level.

mode \ type		individual	collective	electronic
memory has been deleted		<ul style="list-style-type: none"> • termination • death • amnesia • early retirement 	<ul style="list-style-type: none"> • dissolving of adjusted teams • reengineering • outsourcing of competencies 	<ul style="list-style-type: none"> • irreversible data loss through: <ul style="list-style-type: none"> • viruses • hardware mistakes • system crashes • insufficient back-up • hackers
access not possible	limited	<ul style="list-style-type: none"> • overload/limited • transfers • illness/vacation • inadequate training • service according to regulations 	<ul style="list-style-type: none"> • making taboo of routines • collective sabotage 	<ul style="list-style-type: none"> • reversible data loss • overload/limited • interface problems
	permanent	<ul style="list-style-type: none"> • overload/permanent • no awareness of the importance of own knowledge • inner termination 	<ul style="list-style-type: none"> • sale of company divisions • migration of teams • cover-up 	<ul style="list-style-type: none"> • permanent incompatibility of systems • overload/permanent • wrong codification

Chart 4: Types of Organizational 'Unlearning'

raining

This table illustrates that the protection of experience and know-how can be described as a permanent battle against the natural process of forgetting things. Forgetting is a well-known phenomena. A foreign language we once knew but do not speak is lost word by word. Untrained muscles shrink. Learning psychology speaks about *conserving repetition*, well illustrated by the pain of learning vocabulary. Many trainings fail because fresh knowledge cannot be translated and applied in daily activities. Much knowledge fades away over time and when we need it may be already gone. This is the reason for action training in many companies. They train their employees in their typical working situations and hope in this way to conserve the new know-how for a longer time.

eeeping the
roduction line
arm

Another case shows that sometimes high investment is necessary in order to avoid collective amnesia. Over many years, the US government ordered overpriced submarines without any actual demand in order to avoid the loss of skilled workers in this specialized field of construction. These workers were the only ones with the special welding skills and without their knowledge, the US defense department would have faced serious problems in the future. To secure the availability of this know-how for the next twenty years and to establish a pool of future trainers, the US-government decided to keep the production line warm and kept on building submarines.

¹ Following Oberschulte (1996). Oberschulte constructs connections between organizational learning and organizational memory.

² Compare Davenport (1996: 35).

³ So-called redimensioning activities that should reduce costs have led to the loss of capabilities in many companies. Compare Mitroff (1995: 27).

⁴ Compare Hedberg (1981: 18).

⁵ Similar Cohen/Levinthal (1990).

⁶ For different forms of memory, compare Vester (1978: 43ff.). For the importance of the actual memory, compare Wessells (1994: 107ff.).

⁷ The necessity of the materialization of knowledge in knowledge documents is stressed by Schüppel (1996: 256f).

⁸ Compare Probst/Büchel (1994: 21).

⁹ Compare Bonoma/Slevin, (1978: 205).

¹⁰ Compare Economist (20.04.96: 58).

¹¹ Compare Economist (20.04.96: 58).

¹² In this process, the brain instinctively mixes experienced and 'heard' stories. This phenomena is named cryptomnesia by psychologists. Compare Kotre (1996).

¹³ The different approaches to how reality is constructed is not part of this paper. Interested readers can find ideas in the schools of radical constructivism, knowledge sociology and different psychiatric schools. Compare Watzlawick (1986, 1988); Berger/Luckmann (1994) and Sacks (1995).

¹⁴ These observations are compatible with the Freudian idea of a *collective Super-Ego*. For modern forms of collective memory research compare Hejl (1991).

¹⁵ Compare Liang/ Moreland/Argote (1992).

¹⁶ Compare Cohen/Bacdayan (1994).

¹⁷ Compare Wegner (1996:189ff).

¹⁸ Compare Economist (20.04.96: 58).

¹⁹ Schüppel (1996) classifies this method as a process to manage implicit knowledge potentials. Compare Schüppel (1996: 264f).

²⁰ Following Manago/Auriol (1996: 28).

Berger, P.L./Luckmann, T. (1994): *Die soziale Konstruktion der Wirklichkeit: Eine Theorie der Wissenssoziologie*, Frankfurt (Main): Fischer.

Bonoma, T.V./Slevin, D.P. (1978): Management and Type II Error, in: *Business Horizons*, 4, S. 61ff.

Cohen, M.D./Bacdayan, P. (1994): Organizational Routines Are Stored as Procedural Memory: Evidence from a Laboratory Study, in: *Organization Science*, 5: 4, S. 554-568.

Cohen, W.M./Levinthal, D.A. (1990): Absorptive Capacity: A New Perspective on Learning and Innovation, in: *Administrative Science Quarterly*, 35: 1, S. 128-152.

Davenport, T.H. (1996): Some Principles of Knowledge Management, in *Strategy&Business*, 1: 2, S. 34-40.

Economist (20.04.96): Fire and forget?, 20.04.1996, S. 57-58.

Hedberg, B. (1981): How Organizations Learn and Unlearn, in Nystrom, P.C./Starbuck, W. (Hrsg.): *Handbook of Organizational Design*, S. 3-27, New York: Oxford University Press.

Hejl, Peter M. (1991): Wie Gesellschaften Erfahrungen machen oder: Was Gesellschaftstheorie zum Verständnis des Gedächtnisproblems beitragen kann, in: Schmidt, S.J. (Hrsg.): *Gedächtnis - Probleme und Perspektiven der interdisziplinären Gedächtnisforschung*, 2. Auflage, S. 293-336, Frankfurt (Main): Suhrkamp.

Kotre, J. (1996): Weiße Handschuhe. Wie das Gedächtnis Lebensgeschichten schreibt, München: Carl Hanser.

Liang, D.W./Moreland, R./Argote, L. (1992): *Group versus Individual Training and Group Performance: The Mediating Role of Transactive Memory*. paper presented at the 1992 meetings of the Operational Research Society of America/The Institute of Management Science in San Francisco and at Carnegie Mellon University.

Manago, M./Auriol, E. (1996): Mining for Or, in: *OR/MS Today*, Februar: S. 28-32.

Mitroff, I.I. (1995): Warum schaffen wir es nicht? in: *io Management Zeitschrift*, 3, S. 27-31.

Oberschulte, H. (1996): Organisatorische Intelligenz - ein Vorschlag zur Konzeptdifferenzierung, in: Schreyögg, G./Conrad, P. (Hrsg.): *Managementforschung 6: Wissensmanagement*, S. 41-81, de Gruyter.

Probst, G.J.B./Büchel, B.S.T. (1994): *Organisationales Lernen: Wettbewerbsvorteil der Zukunft*, Wiesbaden: Gabler.

Probst, G./Raub, S./Romhardt, K. (1997): *Wissen managen: Wie Unternehmen ihre wertvollste Ressource optimal nutzen*, Frankfurt (Main)/Wiesbaden: FAZ/Gabler.

Probst, G.J.B./Romhardt, K. (1997): Bausteine des Wissensmanagements - ein praxisorientierter Ansatz, in: Dr.Wieselhuber & Partner Unternehmensberatung, (Hrsg.): *Lernende Organisation*, S. 129-143, Wiesbaden: Gabler.

Romhardt, K./Probst, G. (1997): Building blocks of knowledge management - A practical approach. Input-Paper for the seminar: "Knowledge Management and the European Union-Towards a European Knowledge Union", 12-14 May 1997 at Utrecht, Kennis Centrum/Dutch Ministry for Education.

Sacks, O. (1995): *An Anthropologist on Mars*, London: Picador.

Schüppel, J. (1996): *Wissensmanagement-Organisatorisches Lernen im Spannungsfeld von Wissens- und Lernbarrieren*. Dissertation, Hochschule St.Gallen.

Watzlawick, P. (1986): *Wie wirklich ist die Wirklichkeit?* 14. Auflage, München. Zürich: Piper.

Watzlawick, P. (1988): *Die erfundene Wirklichkeit - Wie wissen wir, was wir zu wissen glauben? Beiträge zum Konstruktivismus*, München/Zürich: Piper.

Wegner, D.M. (1996): Transactive Memory, in Mullen, B./Goethals, G.R. (Hrsg.): *Theories of Group Behavior*, S. 185-208, New York:

Wessells, M.G. (1994): *Kognitive Psychologie*, München/Basel: UTB.

The Acquisition of Novel Knowledge by Creative Re-Organizations

Franz Schmalhofer and James Stuart Aitken

German Research Center for Art. Intelligence
(DFKI) Postfach 2080
D-67608 Kaiserslautern, Germany
e-mail: schmalho@dfki.uni-kl.de

Department of Computing Science
University of Glasgow
Glasgow G128QQ, Scotland
stuart@dcs.gla.ac.uk

Position Statement for Workshop 1 at the 21st German Conference for Artificial
Intelligence, 1997, 9.-12. September 97. Freiburg i Br.

For launching innovative products into successful markets, knowledge, its organization and its timely utilization are becoming the most decisive business factors in the currently emerging knowledge society. As Nonaka & Takeuchi (1995) have convincingly pointed out, the companies of the future will live in an environment, where markets are continuously shifting, technology proliferating, competitors multiplying and products will become obsolete overnight. Under such circumstances, where uncertainty is the only certainty, that remains, any successful enterprise must continuously acquire and utilize new knowledge.

However, the knowledge-based system technologies which have been developed in the mainstream of Artificial Intelligence research, such as model-based approaches (Wielinga, Schreiber & Breuker, 1992) and formal ontologies (Gruber, 1993) are not yet sufficiently armed for creatively producing new knowledge that can subsequently be reified into novel products. In the current position paper, it is therefore suggested that human learning and human comprehension, as it has been studied in cognitive science (Schmalhofer, in press), can also serve as a model for the constructive and creative learning processes that may occur in organizations and businesses. In particular, we thereby emphasize creative learning processes (see: Boden, 1991).

Learning by *creative inferencing* does not only (deductively or inductively) explicate some piece of information that is already implicitly contained in the available information, but instead creates some unexpectedly novel (and possibly quite abstract) statement by merging information from different conceptual spaces. A creative inference thus requires the immersion of information from another conceptual space which is then used (in combination with other information) to construct a novel *creative initiative*.

Given this presupposition, we describe the EKI system, (EKI = Evolution of Kreative Initiatives). With the EKI system (see Schmalhofer, Franken & Schwerdtner, 1997), the scientists and practitioners of a company can independently advertise within a company their most significant or most favorite competence (as well as cooperation possibilities with colleagues and other departments) in knowledge-bases. The knowledge-bases may

for example contain personal skill profiles as well as the actual or desired activities in which a practitioner participates (or wants to participate) in. These knowledge-bases thus contain knowledge-level descriptions of autonomous agents (Aitken et al., 1994) upon which creative inferencing can be performed.

A *knowledge manager* may then apply the methods of the EKI-tool for analyzing the company's *knowledge assets* with respect to possible business initiatives. More specifically, by a user-programmable marker-passing process (which is to be programmed by the knowledge manager), possibilities for promising innovations are first identified. Via a compilation process, knowledge maps for enterprise-specific successes can then be generated. These processes should be performed in a way so that there is a maximum compatibility to existing enterprise modeling approaches. Ideally the EKI-tool should be integrated with current workflow-modeling technology.

With this analysis, creative initiatives towards promising innovations can subsequently be proposed. Unlike the current wisdom in the field of Artificial Intelligence, solutions to knowledge management problems are thus no longer seen as developing information processing machines (see Gardner, 1987), but instead as providing useful and useable tools to knowledge workers (see: Kidd, 1994) and their creative thoughts so that they will be better able to anticipate the scope of the possible future market successes for their company. In our position presentation, we will show a worked-out example of how the EKI-tool is applied for these purposes. It will thus become clear how the EKI-tool can be applied for achieving a cooperative knowledge-evolution process (see Schmalhofer & Tschaischian, 1995) and how this process can be guided by future market needs.

References

- Aitken, J. S., Schmalhofer, F. & Shadbolt, N. A knowledge level characterisation of multi-agent systems. In: Wooldridge, M.J. & Jennings, N. R. (Eds) Intelligent Agents: Agent Theories, Architectures and Languages, Berlin: Springer-Verlag, (Lecture Notes in Artificial Intelligence 890) pp. 179-190, 1994.
- Boden, M. A. (1991) The creative mind: Myths and mechanisms. New York: Basic Books, Inc., Publishers.
- Gardner, H. (1987). The mind's new science: A history of the cognitive revolution. New York: Basic Books, Inc., Publishers
- Gruber, Th. R. (1993) Towards principles for the design of ontologies used for knowledge sharing. In N. Guarino and R. Poli (Eds) Formal Ontologies in Conceptual Analysis and Knowledge Representation. Kluwer Academic Publishers, in press.
- Kidd, A. (1994). The marks are on the Knowledge Worker. Proceedings of CHI-94: Human Factors in Computing Systems, Boston, Mass (24-28 April'94), 186-191. ACM Press: New York.
- Nonaka, I., & Takeuchi, H., (1995). The knowledge-Creating Company. Oxford: University Press.
- Schmalhofer, F. (in press) Constructive Knowledge Acquisition: A Computational Model and Experimental Evaluation. Hillsdale: Lawrence Erlbaum Associates.

Schmalhofer, F., Franken, L., & Schwerdtner, J. (1997) A computer tool for constructing and integrating inferences into text representations. Behavior Research Methods, Instruments, & Computers.

Schmalhofer, F. & Tschaitichian, B. (1995) Cooperative knowledge evolution for complex domains. In G. Tecuci & Y. Kodratoff (eds.) Machine Learning and Knowledge Acquisition: Integrated Approaches, London: Academic Press, 145-166.

Wielinga, B. J., Schreiber, A. T., & Breuker, J.A. (1992) KADS: A modeling approach to knowledge engineering. Knowledge Acquisition, 4 (1), 5-53.

Making the Tacit Explicit: The Intangible Assets Monitor in Software

Dr. Charles A. Snyder
Professor
Department of Management
Auburn University
Auburn, AL 36849 –5241 U.S.A.
*e-mail: snyder@business.auburn.edu

and

Larry T. Wilson
President and Chief Science Officer
LearnerFirst, Inc.
The Genesis Center, Suite 201
Birmingham, AL, 35211-6908 U.S.A.
ltw@learnerfirst.com

* Corresponding author

Abstract

One of the core principles of the Knowledge Management movement is that of capturing expertise and making it accessible to those in the organization who need it (see, e.g., Amidon, 1997; Stewart, 1997). The process of harvesting the tacit knowledge of the expert and converting it into a form that is available and useful presents some formidable obstacles. We explore the needs and the means to perform the needed activities that will result in a computer-based resource that fulfils those needs.

An example of the process is provided by describing the creation of the Intangible Assets Monitor as conceived by Karl-Erik Sveiby, the expert in this instance and one of the founders of the knowledge management community of practice. The result is a software product developed by LearnerFirst, a company that specializes in harvesting the knowledge of experts and incorporating it in computer-based learning resources. These learning resources encapsulate the tacit knowledge of an expert, placing it in explicit form. The software supports procedural knowledge-based tasks by providing the learner with expert guidance on an as needed basis.

An example of the software generated from the Know-How Harvesting process will be demonstrated at the conference.

The Knowledge Harvesting Process

Know-how Harvesting™ is a term applied to the process of eliciting the tacit insights and intuitive knowledge of experts or top performers and converting it into specific, actionable know-how that is easily accessed and used by others. The developer firm, LearnerFirst, formed in 1992, has specialized in this process since its inception. The process is akin to the more familiar Knowledge Engineering activity in the Expert Systems field (see, e.g., Keller, 1987; Wiig, 1990). However, the process differs in its standardized approach and the outcome. Learning support software is the primary output from the process. Also, the intent is different from the expert systems as the software is used to supplement, not replace a decision-maker.

The harvesting process can be applied to virtually any kind of human knowledge that is procedurally actionable. This means that the process is as relevant to the highest level of knowledge as well as to the routine. The resulting software from the process is designed so that an individual can simultaneously understand, learn, perform, and record the performance in a single action. In this sense, it can be classified as an Electronic Performance Support System (EPSS)(See, e.g., Gery, 1991). An EPSS is defined as „...the electronic infrastructure that captures, stores, and distributes individual and corporate knowledge assets throughout an organization, to enable individuals to achieve required levels of performance in the fastest possible time and with a minimum of support from other people“ (Raybold, 1995, p.11). harvesting is described in more detail in a following section.

The Firm and Products

LearnerFirst, Inc. was founded with the aim of producing computer-based products that could augment human intelligence. The conceptual foundation of the company had as its core, the thesis research of the founder. The software is thoroughly grounded in learning theory because it is designed to assist the user (learner) learn from the captured knowledge of an expert while in performance of a process.

Several software products have been developed and marketed by this company. These products have served to verify the know-how harvesting system. There have been dozens of experts whose knowledge has been made accessible through applications that are in daily use in over 4,000 organizations around the world. Thus, the concept has had verification in the marketplace.

What is harvesting?

The harvesting process is proprietary, however it can be described in general terms. During the knowledge engineering process, the expert and the knowledge engineer walk through the process and separate the relevant

procedural knowledge from the trivial. The focus is on the procedural or process domain and the organization and sequencing is based on the expertise of the domain expert.

Harvesting is a set of methods for: 1) finding valuable know-how, 2) getting inside the mind of the expert performers to uncover the processes involved, 3) optimizing and deploying the know-how to individuals and teams as software applications, and 4) evaluating and improving applications.

After a top performer or process expert has been identified, LearnerFirst utilizes methods for eliciting and re the expert's intuitive and explicit knowledge. Then, the expert's thinking processes are broken down into their elementary components – mental operations and knowledge units – which may be viewed as psychological „atoms“ and „molecules.“ Researchers at LearnerFirst have discovered that performing any intellectual task or solving any problem requires a certain set of elementary knowledge units and operations. These operations and units are generic, domain-neutral, irreducible thinking processes that were brought to light after an extended study of over 300 cases in which the LearnerFirst harvesting process was applied

Information about the thinking process is optimized as specific, actionable know-how that is easily accessed, understood, and applied by target learners/performers. Specifically, the information is communicated to users as guidance and support information via a software application. Guidance is information about how to actually think through the process task. Support information is information to enhance understanding. The software application's user interface varies in relation to the task that needs to be performed and the user's individual preferences. Beneath the surface of the user interface, there is a database that successfully holds guidance and support information according to three primary levels of abstraction.

After repeated use of a performance support application, cycles/iterations are evaluated so that improvements can be made. Thus the application is designed so that it can continue to evolve or learn.

Benefits

There are several major benefits of the harvesting system. First, successful thinking is made visible, manageable, useful, and accessible. This means that expertise is available to anyone who needs it, anywhere in the world, when they need it. All that is required is computer access to the software. This can be either in a stand-alone version or via a network such as the Internet. Organizations may wish to make their internal expertise captured by the harvesting process available via Intranets.

Second, decisions can be made faster at lower levels because the expert's knowledge is available at lower levels. In essence, we can have just-in-time access to the needed expertise.

A third benefit is that work can be properly performed with less supervision and intervention. There are also some important impacts on the individual learner. As the learner progresses, he or she gains in proficiency, confidence, and autonomy. Thus, we are able to improve on the organization's human capital through increasing competency. This aspect is of great importance because of the leverage effect on the firm's performance (see Sveiby, 1997 for a discussion of the leverage effect).

Fourth, the organization can use the process to preserve know-how as its most important asset. The nature of the system also allows the individual learner to record experience and learning so that the individual has a continuously updated resource with perfect memory.

Last, the learning resources improve performance in a cost-effective manner. Since the harvesting process is thorough, the captured expertise is usually more complete than that available if the expert were on site. Our experience has revealed that the process of eliciting the tacit knowledge of an expert helps the expert to formalize his/her concepts with a great deal of clarity so that they can be made explicit.

An Example

As an example of the harvesting process output, the first portion of the resource developed with Karl-Erik Sveiby is presented. Sveiby was the first to write about the „Knowledge Company“ in 1986 and has been a leader in the Swedish community of practice surrounding knowledge management and intellectual capital management (Amidon, 1997; Sveiby, 1997). His books and articles are widely referenced and quoted. His concept and taxonomy of Intangible Assets with the components of Customer Capital (External Structure), Human Capital (People's Competence), and Structural Capital (Internal Structure) are widely known and used by many organizations.

The learning resource under development with Sveiby is the „Intangible Assets Monitor.“ The entire Intangible Assets Monitor includes many features that will be beyond the scope of this presentation. The Invisible Balance Sheet and a simplified version of the monitor will be presented as an example of how the harvesting process is used to generate a learning resource. The software provides individualized guidance for creating and evolving measures of intangible, knowledge-based assets of an organization.

Conclusion

Every firm that undertakes the task of capturing the knowledge of its best performers or that wishes to make expert advice on performance of its key processes available to performers is faced with the problem of harvesting the know-how. The captured knowledge needs to be made available to the individual performer on an as needed basis. The methodology described allows learning resources to be produced that provide the individual with performance oriented expertise. By deploying such software, developed with the aid of the harvesting methodology, a firm can manage its knowledge transfer process as well as remove a significant obstacle to knowledge transfer. The learning resources generated from this process become an important part of a firm's knowledge management strategy.

References

- Amidon, D. M. (1997). The Ken Awakening: Management Strategies for Knowledge Innovation, Boston: Butterworth-Heinemann.
- Gery, G. (1991). Electronic Performance Support Systems: How and Why to Remake the Workplace Through the Strategic Application of Technology, Boston: Weingarten.
- Keller, R. (1987) Expert System Technology: Development & Application, Englewood Cliffs, NJ: Yourdon Press.
- Raybold, B. (1995). Performance Support Engineering: An Emerging Development Methodology for Enabling Organizational Learning. Performance Improvement Quarterly, (8:1), 7-22.
- Snyder, C. A. & Wilson, L.T. (1997). Individual Learning: The Foundation of Knowledge Management, Proceedings, The Austin Knowledge Management Dialog, Austin, TX.
- Stewart, T. A. (1997). Brain power: Who Owns it...How They Profit From It, Fortune, (135:5), 104-110.
- Sveiby, K.E. (1997). The New Organizational Wealth: Managing and Measuring Knowledge-Based Assets, San Francisco: Berrett-Koehler.
- Wiig, K. (1990). Expert Systems: A Manager's Guide, Schema.

Knowledge Management for Electronic Insurance Catalogs

Dr. Markus Stolze
IBM Zurich Research Laboratory
Saumerstr. 4 CH-8803 Rueschlikon Switzerland
phone: +41/1/724 8263 fax : +41/1/724 8953
email: mrs@zurich.ibm.com
<http://www.zurich.ibm.com/~mrs>

Talk Abstract:

Knowledge Management for Electronic Insurance Catalogs

Electronic Product Catalogs enable users to find, compose, compare, and buy products electronically. Catalogs for insurance products have to account for the fact that these products have a rich and customisable structure. This is why existing cataloging tools do not work very well for insurance products. In my talk I will illustrate how the structure of insurance product knowledge can be used to improve catalog searching and catalog maintenance. I will also point to two important challenges which I see as related to the issue of knowledge management: the creation and management of insurance ontologies for open catalogs and the cataloging of unstructured product information.

Relevant Background:

For a number of years I have been performing research dealing with the problem of how to develop knowledge based systems in such a way that it is easy to "get knowledge into them and out again".

I first got interested in this topic when I developed an "expert maintainable" knowledge based system (Bunke, Stolze (1989) INDEX - A Diagnostic Expert System for a mail inserter machine, Microcomputer Applications 8(1), 27-33). Later my interest shifted to include also the usage aspects of knowledge based systems and how to enable users to make best use of the knowledge contained in a such a system. (Stolze (1994) From system requirements to appropriate knowledge representations: a case study. Proceedings AI for Applications, 156-162).

While being at the University of Colorado I worked in a research group that investigated evolutionary approaches to the problem of knowledge based system development and how such systems can be used as resources for communities of practitioners. (Fischer, G., Lindstaedt, S., Ostwald, J., Stolze, M., Sumner, T., Zimmermann, B. (1995). From Domain Modelling to Collaborative Domain Construction. Proceedings DIS'95.)

Here at IBM research in Zurich I am now involved in a project where we create technology components for flexible product catalogs.

Toward a Method for Providing Database Structures Derived from an Ontological Specification Process: the Example of Knowledge Management

by Gary Templeton and Charles Snyder
Department of Management
Auburn University, 36849-5241, U.S.A.
Contact: snyder@business.auburn.edu

Abstract

The paper describes the operation of a methodology used for ontologically specifying the key concepts in the field of Knowledge Management. Ontological specification is of particular interest to knowledge managers because associated methods support top management in the processing of tacit knowledge into a more explicit form. The paper is a part of a larger project designed to utilize ontological processes for the building of a 'best practices database' in the KM field. Implications of such a database involve greater speed in which dynamic and unstructured fields such as KM can develop into a more explicit and transferable form of knowledge.

Introduction

Since the advent of computing technology, researchers in information systems and particularly in artificial intelligence (AI) have made several attempts at the formalization of a process which can be termed *ontological specification*. An ontology is the explicit specification of conceptual meaning in a topic area such as Knowledge Management (KM) in this project. The ontology includes a complete vocabulary and logical statements about the terminology, how the terms are defined and how they are related (or not related). The process involves the derivation of more specific meaning from more general concepts. This is one method for providing meaning in a knowledge domain. Its applicability in AI is such that methods can be constructed which attempt to automate the human information processing tasks associated with deducing logical structures, commonalities and relationships in conceptual representations found in textual data. The goal of such methods is the reduction in need for humans to be inundated with the belaboring task of extracting meaning from volumes of text. The need for such systems derives from the proliferation of web-based media, which has caused a dramatic increase in organizational access to business intelligence in textual form. The application of such structuring processes could help greatly in the growing area of KM, an organizational function where knowledge is considered to be the strategic focal point for competitive advantage.

A problem that organizations are faced with is that of increasing access to masses of strategically critical contextual and conceptual information. This coincides with limited means of automated support in the form of computer-based information technology for the capturing and organization of important knowledge. This paper provides some insights on the idea of ontological specification (or *concept structuring*) by depicting the application of one such method in the area of KM. Thus, we describe a tool that can greatly aid in the process of KM in its own field.

Although the field of KM has just recently started to take off, the prospect of gathering all of the information published in the area is formidable. This is especially true given the enormous growth in the amount of information published each month on the topic area. The immaturity, yet dynamic development of the field of KM means that the field has achieved little structure in terms of frameworks, commonly held beliefs, taxonomies, and terminology. This has made the field an excellent target for the application of a method for its structuring.

Previous Work in Structuring Tacit Knowledge

Much work has been done with the goal of providing algorithmic approaches to extracting knowledge from empirical data (whether qualitative or quantitative). Past work in the structuring of numeric data is vast, including the entire field of statistics, graphing methods, and obscure methods, such as the symbolic-oriented FACES concept (Bruckner, 1979). The latter involves an analysis resulting in characterizations about numeric data portrayed in the form of illustrated human "faces".

Attempts at structuring of qualitative text data has a long tradition, beginning with the study of language structures (parts of speech, grammar, etc.), and more recently, proprietary text processing methods made possible from the use of computer-based information technology.

A rudimentary form of IT-enabled proprietary methods might include the counting of each different word appearing in a text. The development of such a database enables the objective development of an argot by using the most commonly used terms found in the analysis. The argot can then be used subjectively to decide which themes (i.e., technological or managerial terms) are emphasized and what themes are important in a body of text. For example, a researcher can determine if a CEO is more concerned with technical or managerial issues by analyzing which type of words he/she emphasizes in the annual statement. In this way, such a method can serve as a utility function used in the preliminary analysis of a body of text.

There are various forms in which AI programs work to transform English statements into computer-based representations for processing. These methods generally take the form of transferring statements into a objects, actions and actor representations for the output of logical inferences (Schank, 1984). While serving its purpose in uncovering inferences, this method does not have practical use in processing a mass of text.

Perhaps the most advanced attempt at using language structures for management, other than the implications of ontological specification, has evolved in the form of Zachman's (1987) ISA Framework. Zachman used three (later developed into six) possible questions asked about a business system with each segregated into six roles of system analysts. The original framework considered every system to be composed of data ("what"), process ("how"), and network ("where"). Sowa and Zachman (1992) later extended the three question categories to include "who, when, and why" components of systems, meaning that all questions available in the English language are reflected in the model. Implications were that information related to all possible questions about the real-world business system could be represented in the computer.

A convenient idea in information systems literature concerns the applicability of normalized database relations in relating to themes of meaning. The subject nature in which normalized database relations are evaluated relates strongly to judgements concerning the appropriateness of ontology structure. In fact, the two processes are very similar. Both are attempts at capturing meaning using standardized representations of meaning (concepts and subconcepts for ontological specification and relations and elements for normalization). It can be said that both processes are attempts at organizing knowledge in a more straight-forward manner for an end user (or agent) of the knowledge acquired.

The Ontological Concept-Specification Process

Definition of an Ontology

An ontology represents a specifying scheme of concepts which holistically describes some topic. For our purposes, it can yield declarative knowledge about the structure and processes related to the KM concept. It can serve as a formal vocabulary for researchers,

instructors, students, and practitioners in the KM community of practice and includes logical descriptions of the items, relationships between items, and how items cannot be related (Gruber, "What is an ontology?").

The utility of specific terminology is a greater stimulation in more refined concepts in the field, which can result in further formalization of the topic area and structuring of decision-making processes in the field. Further, the KM ontology can be used by researchers to uncover what subconcepts are and are not related when such relationships are of interest. This process of greater formalization inevitably results in prescriptive associations in academic fields, which means that methods can be prescribed for various contingencies found in practice. Ontologies can be viewed as a knowledge-based communications technology in that a greater ability to represent and communicate knowledge about a concept is possible as more terms and relationships are uncovered in the ontological specification process. With this definition of ontological specification at hand, we can see that the process itself can serve as a subconcept in the KM ontology (see table), a scenario which adds to the implications of this research.

The Utility of an Ontology

In viewing the ontological specification process as a knowledge-based technology, researchers at Stanford University are at the forefront. There, a web-based ontology building application resides, allowing users to build ontologies of any subject matter. The system, called Ontolingua, serves as a knowledge acquisition laboratory for the AI faculty and supports the standardization of knowledge structures transferable to intelligent software modules. The system has been used to build ontologies mainly in the field of AI, but some in areas which can be directly linked to KM: Bibliographic-Data, Documents, Job-Assignment-Task, User, Design, and Domain.

The many-to-many relationship in the ability of the application to receive ontology-building requests and to be able to transfer to varying software environments means that the ontological specification process has received ontological attention. The ontology engineering perspective has yielded a meta-ontology which can be used across any knowledge domain, much of which can be found in the work of Gruber (1992).

The Ontological Specification Process

Levels of specification during the ontological specification process are delineated as shown in Figure 1. While not all ontological specification methodologies are the same, all can be said to share common steps. These steps are 1) *selection* of the topic area, 2) *delineation* of concepts, 3) *transfer* to a usable medium, and 4) *use* of concepts. Operationalization of the steps depends on the purpose of the ontology. For example, the purpose of the Stanford ontology procedure involves capturing the knowledge for transfer into rigid data structures, which requires further analysis of the definitions of knowledge elements. The current study uses the ontological process as a way of marking literature for future literature reviews of the KM field. The current methodology was structured to accommodate this purpose, in the form of 1) *selection* of the KM topic, 2) *delineation* of concepts (as illustrated in Figure 1) by literature review, 3) denotation of concepts in the literature, 4) *transfer* to database format, and 5) *use* of database in research.

The Ontological Specification of KM Illustrated

Selection of the KM Topic

Selection of the topic of KM for ontological analysis was done due to its place on the research continuum. A review of the literature showed that an overwhelming amount of the knowledge available about KM was descriptive in nature. This meant that most effort had been justifyingly aimed toward the definition and uncovering of key concepts in the field. Very little academic empirical work had been conducted except for conceptual "blue sky" (normative) works using examples of KM utilization in the field (as in Sanchez and Mahoney, 1996). Thus, the field was a prime target for formal structuring methods such as ontological and taxonomic specification.

Delineation of Concepts by Literature Review

Operation of ontological specification should be seen as an iterative and subjective behavior of the agent operator and heavily dependent on operator learning. The classic iterative control process is used throughout ontology development, following the steps of 1) setting an ideal, 2) setting standards, 3) evaluation of feedback data, 4) changing operations or the ideal in perpetual cycles (Newman, 1975). Deciding on an exact ontology cannot be accomplished in such a new and volatile field as KM, where expected discoveries in the field will relate to a redefinition of the ontology. Defining and placing relevant subconcepts in an ontology is done based on some purpose, such as 1) to define and refine a researcher's interests, 2) to capture common themes in a body of literature and 3) to organize experiential knowledge about a research topic.

Thus, the delineation of the KM concept does not mean that KM will be the most significant or important field in the analysis. It means that as we decompose the concept, other related concepts should be included in the study of KM (see Figure 1). For example, executive information systems, telecommunications and database are more mature and may be more important in the eyes of managers, but they may relate in a strong way to KM. The *relationship* between these important concepts and KM is of pertinent concern to users of the ontology and thus justifies its existence in the scheme.

Figure 1: Delineation of Levels in the Ontological Specification Process

Level of Description		Example from Current Study
<div style="display: flex; align-items: center;"> <div style="writing-mode: vertical-rl; transform: rotate(180deg);">Specificity</div> <div style="margin-left: 10px;"> ↓ ↓ </div> </div>	Concept (KM) → Specification	Knowledge Management
	Concept (KM) → Specification	III. KM Operation
	Concept (KM) → Specification	D. How KM impacts the organization (OI)
	Concept (KM) → Specification	1. Organizational change and KM ()
	Concept (KM) → Specification	a. How technology can support KM (TECH)
	Concept (KM) → Specification	i. Learning Systems (LS)
	.	ii. Best Practices databases (BEST)
	.	iii. Organizational Memory Info. Sys. (OMIS)
	.	iv. Networking and KM (NET)

An initial review of the KM literature revealed three categories of knowledge in the literature associated with the KM concept. These categories are Resource Meta-Data, KM Description and KM Operation (see Table 1).

Meta-Knowledge involves the entomological view, which includes knowledge about sources, states, structures, processes, histories, and evaluations of knowledge about KM, including the current paper. In most fields, Meta-Knowledge is important for researchers but of little value to practitioners. However, the understanding of Meta-Knowledge is the goal of one who studies KM: This category of knowledge is where the objectives of practice and academia coincide.

The *KM Description* concept involves descriptive knowledge about the field. This type of knowledge is emphasized early in the development of academic fields and is concerned with the defining of key issues, terms, and the history of the field. Writers of KM literature have emphasized *KM Description* knowledge for better understanding of the meanings of the concept and related terminology.

KM Operation involves the structure and processes associated with the topic. We use this type of knowledge to depict for management the way KM should be aligned and operate. This type of knowledge has been relatively rare in KM literature because the field has concentrated on describing the field (as in *KM Description*). Work in the area of operationalizing KM will lead to prescriptive knowledge about what causes, effects and contexts are important in operating the organizational KM effort. Thus, this type of knowledge is the goal of academic research and study about the field and is the direction KM is currently heading.

It is important to understand the three categories of KM knowledge posed in the framework. The nature of Meta-Knowledge is descriptive, as in the *KM Description* category. However, the former is used in describing KM knowledge while the latter is used to describe KM. The *KM Operation* category is where we learn to competently manage knowledge and its processes.

Table 1 shows how KM can be delineated into more specifying subconcepts. For example, the early development stage of the field results in the need for researchers and practitioners to have agreed-upon definitions and a foundation for terminology. Hence, the consideration of the *Definition of KM* idea has become important and was placed as a direct subconcept of the *KM Description* concept in the ontology. The concept was derived by the consistent efforts by several 'guru's' offering varying definitions in the KM literature.

Table 1: The KM Ontology

I. Resource Meta-Data	
A.	Source type (SO)*
B.	Study Type (ST)
C.	Academic base (AB)
D.	Empirical Support (EM)
II. KM Description	
F.	History of KM (HIS)
A.	Definition of KM (DEF)
B.	KM characteristics (CH)
C.	How to determine the presence of KM (PRES)
D.	Examples of KM and its Absence (EX)
E.	KM Architecture (ARCH)
III. KM Operation	
A.	Processes of KM (ACT)
1.	Determining info requirements during KM (IR)
2.	Knowledge Acquisition (KA)
3.	Data Management in KM (DM)
4.	Processing/Transforming Knowledge (PROC)
	<i>Ontological Specification</i>
5.	KM and GST (GST)
6.	Organizational Learning (OL)
7.	Organizational Memory (OM)
B.	Why KM is needed (NEED)
	Control Theory and KM (CT)
C.	Knowledge as Intangible Asset (IA)
1.	Knowledge capital theories (KT)
2.	Knowledge creation (KC)
3.	Intellectual Capital Management (ICM)
D.	How KM impacts the organization (OI)
1.	Organizational change as related to KM (♦)
a.	How <i>technology</i> can support KM (TECH)
i.	Learning Systems (LS)
ii.	Best Practices Databases (BEST)
iii.	Organizational Memory Info. Sys. (OMIS)
iv.	Networking and KM (NET)
b.	KM <i>culture</i> (CULT)
2.	Organizational performance (OP)
E.	Organizational use of KM (OUSE)
F.	Benefits of KM (BEN)
G.	Factors effecting quality KM effort (FEQ)
1.	Implementation of KM (IM)
2.	Evaluation of KM (EV)
3.	Characteristics of the Knowledge Manager or Group (CHAR)
* parentheses indicate subconcept tag used in text denotation process	

The ontology in Table 1 also shows the decomposition of the *KM Operation* subconcept of KM. *KM Operation* literature was found to be described by seven subconcepts relating to the issues associated with KM practice. One intriguing facet of *KM Operation* is the *Processes of KM* subconcept, relating to the activities associated with KM practice. These activities may be classified as operational or managerial activities, but were explicitly mentioned in the literature as one of the seven categorizations shown. Further decomposition leads to the *Processing/Transforming Knowledge* subconcept, which includes practices such as *Ontological*

Specification and the textual processing methods mentioned previously. This ontology shows the congruence in purpose between academics and practitioners in the KM field. The previous description of the method of concept delineation highlights several problematic issues in the derivation and selection of subconcepts. First, the description points out that a methodology operator must subjectively select from competing ideas the most appropriate and pertinent subconcepts to be placed in the ontological structure. An important principle is that the operator must have knowledge in the area of interest due to the potential for researcher bias in selection. In his engineering approach to ontological specification, Gruber (1993b) refers to this problem as "encoding bias" and uses it as a measure of ontology quality.

Denotation of Concepts in the Literature

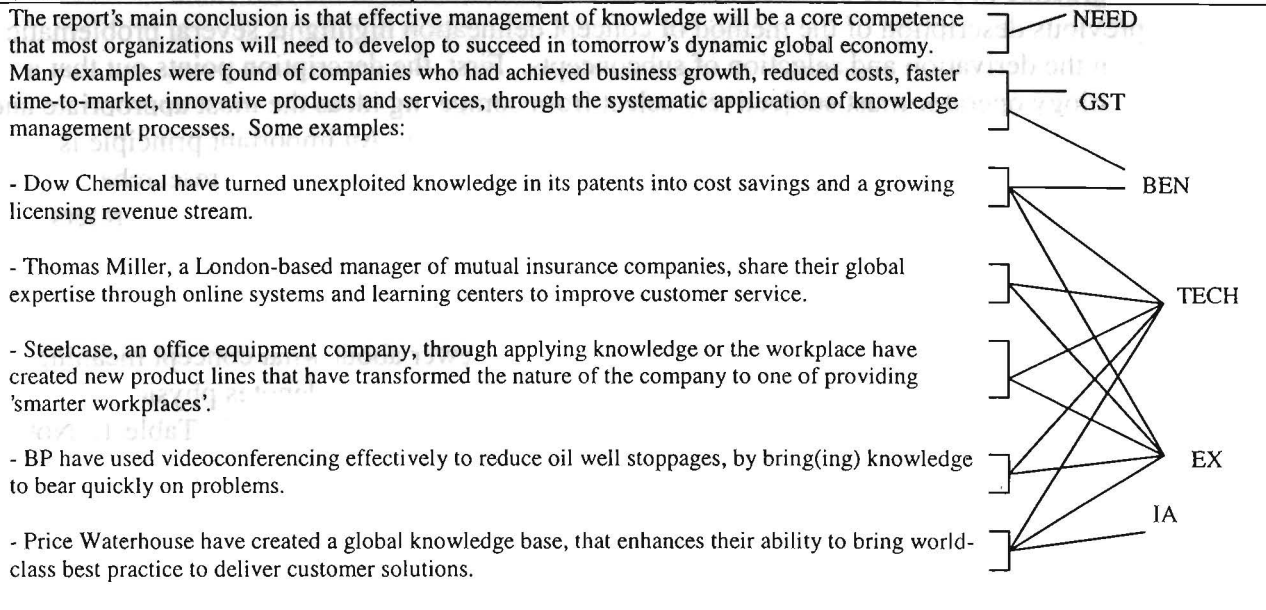
For immediate communication to a literature reviewer about what concept meaning is associated with a given set of text, the specification operator simply denotes physically in the literature using concept-associated tags such as those shown in parentheses in Table 1. Notation tag creation is done with two purposes in mind: 1) to support the learning curve parameters of the denoting specialist and 2) to communicate to a notation user about meanings in the text. Denoting text can become extremely tedious without the utilization of communicative tags. For instance, the current methodology was initially employed using numeral tags until a more descriptive variable name approach evolved.

Denotation criteria, the standard by which each reviewed text set is evaluated, is an important consideration and should be documented for all denoting specialists employed on a specification project. Well communicated and implemented evaluation criteria can result in less risk of subjective bias and other problems in the ontology. For example, the researcher denoting the current project on KM would document explicit knowledge ("the statements must contain two of the three words in related statement or set of statements") about the criteria by which *Knowledge Capital Theories* are tagged with the KCT.

The relationship between denotation tags and textual data items is many-to-many. This means one instance of text can have many tags and one tag can have many instances of text in a body of dialogue. Figure 2 shows specific examples of each, and depicts the complexities which can arise in extracting meaning from textually represented knowledge.

Figure 2: Examples of Denotation in KM Literature

(text extracted from: Amidon and Skyrme, 1996)



Transfer to (Relational) Database Format

Use of denoted data, stored as shown in Figure 2, can be tedious and time-consuming in knowledge work like literature reviews. For this reason, and due to advances in relational database technologies, it can be said that conversion to a relational format is desired (and is coincidentally a natural process). Figure 3 shows the result of this conversion, a relation logically depicting which literature work is associated with a given concept.

Figure 3: Relational Representation of Denotation Instances

	SO	ST	AB	EM	HIS	DEF	...
Title1	Book	Field	Academic	Conceptual	0	1	...
Title2	Dissertation	Field	Practitioner	Empirical	1	0	...
Title3	Book	Case Study	Academic	Empirical	1	1	...
Title4	Periodical	Blue Sky	Academic	Conceptual	3	1	...
...

Conversion can be done with or without further analysis, which presents the opportunity for a KM specialist to convert tagging instances to either of the four data types (nominal, ordinal, interval, or ratio). For example, Figure 3 shows that the specialist may input ratio data which may represent the number of times *History* (HIS) is tagged in a specific work. It is shown that *Definition of KM* either does (DEF = 1) or does not (DEF = 0) exist in a given title and is therefore represented nominally. Qualitative data in this case is more appropriate for *Definition of KM* since multiple definitions of KM usually do not usually appear in text sources. The definition of the data type of a concept can be important, because qualitatively defined concepts have implications for the range of capabilities associated with hypertext links between concepts and text instances.

More elaborate examples of concepts where a qualitative design is appropriate are those of *Study Type* (ST) and *Empirical Support* (EM). *Study Type* may be coded as either a field, case, or "blue sky" study upon conversion from denotation for example. These examples show

the richness by which denotation can be captured in relational format but also the inevitable reliance the intuitive feel of knowledge workers operating the methodology depicted.

Use of Database in Research.

The storing of denotation in relational form implies that normalization decomposition procedures can be performed on the data in the production of a KM production database. The goal of such a database is to relate a knowledge manager to literature corresponding to good and bad practice in the field, a KM *Best Practices Database* (KM-BPDB). A data-driven approach to using the KM-BPDB would involve sorting values to find where knowledge about a particular concept exists in the literature. A researcher on KM would use this query to build one ontological specification of the field or to discover associations and terms much faster than with traditional research approaches. For example, the user may wish to review all *Definitions of KM* (shown in Table 2) for further ontological refining.

Table 2: Some *Definitions of KM*

(source of definitions: The Knowledge Management Forum, 1997)

KM 'Guru' Source	Definition of KM
Thomas Bertels	the management of the organization towards the continuous renewal of the organizational knowledge base - this means e.g. creation of supportive organizational structures, facilitation of organizational members, putting IT-instruments with emphasis on teamwork and diffusion of knowledge (as e.g. groupware) into place.
Denham Grey	an audit of "intellectual assets" that highlights unique sources, critical functions and potential bottlenecks which hinder knowledge flows to the point of use. It protects intellectual assets from decay, seeks opportunities to enhance decisions, services and products through adding intelligence, increasing value and providing flexibility.
Brian Newman	Knowledge Management is the collection of processes that govern the creation, dissemination, and utilization of knowledge
Karl-Eric Sveiby	the art of creating value from an organization's intangible assets.
Karl Wiig	focusing on determining, organizing, directing, facilitating, and monitoring knowledge-related practices and activities required to achieve the desired business strategies and objectives.

A goal-driven approach may include powerful drill-down capabilities built into the database in the form of querying languages, embedded scripts and external applications. For example, we may wish to review all articles containing both *KM Characteristics* (CH) and *How Technology Can Support KM* (TECH) using SQL capabilities. Management can use the ontological schema to segment work activities in a knowledge model, so that practices and technologies can be researched for a given work setting (much like how software support personnel work currently). In the face of enormous growth in online textual data, these examples show how the process of transforming ontologically specified elements to relational format can support the conceptual decision-making activities of strategic management. Last, systems can be built with capabilities to perform ontological specification, denotation, and transfer to standard relational data structures, and processing of electronically held textual knowledge.

Conclusion

Although a majority of the KM ontology depicted is concerned with prescriptive concepts in the KM field, a majority of the available literature remains descriptive (estimated at about approximately 90%). Appropriately, most literature on KM at this point is available from non-

academic sources, as we know very little about variables and constructs in the field. Where no cause-effect relationships have been found, most literature is of the "blue-sky" nature where we assume the normative case and the field's position on the research continuum is justifiable in this early stage of development.

The point of this study was to show that ontological specification can aid researchers in providing structure in its own field, Knowledge Management. The prospect of transferring ontologically derived concepts to databases is very appealing and feasible. It has long been held that databases are much easier and efficiently created, operated and maintained when the data is held in logical structures. The process of normalizing a relation has this purpose and is relevant when the data is held in relational format. Thus, research is needed in the area of proper normalization of such an ontologically derived relation. It is the contention of this research that normalization will follow a very similar decomposition process as that found in the ontology presented and new elements of the field can be uncovered in that process. Further, the notion that an ontological notation scheme should consist of unique values suggests that a given ontology lends itself to the application of a production database scheme, where concepts are records and properties can be generated that describe important dimensions of the field.

Knowledge about KM and ontological process can speed the specification - automation process which has enormous implications for supporting work associated with conceptual knowledge (in unstructured domains of problem solving). The process of considering and culling ideas for positioning on a KM ontology is in itself an important area of research in Knowledge Management. This leads to the postulate that ontological specification is a very intricate part of KM and that its study is vital in understanding KM and organizational learning.

References

Amidon, Debra M. and Skyme, D.J. 1996. *Business Intelligence* (newsletter). Published by Business Intelligence, Ltd., London.

Bruckner, Lawrence A. 1979. The interactive use of computer drawn faces to study multidimensional data, Department of Energy, Los Alamos, N.M.

Gruber, T.R. Accessed 5/12/97. *What is an ontology?* Available from the WWW at: <http://www-ksl.stanford.edu/kst/what-is-an-ontology.html>.

Gruber, T.R. 1992. *Toward principles for the design of ontologies used for knowledge sharing*. In Nicola Guarino, Ed., International Workshop on Formal Ontology, Padova, Italy.

Gruber, T.R. 1993a. *A Translation Approach to Portable Ontology Specifications*. Knowledge Acquisition, 5(2):199-220.

Gruber, T. R. 1993b. *Toward principles for the design of ontologies used for knowledge sharing*. Originally in N. Guarino & R. Poli, (Eds.), International Workshop on Formal Ontology, Padova, Italy. Revised August 1993. Published in International Journal of Human-Computer Studies, special issue on Formal Ontology in Conceptual Analysis and Knowledge Representation

Gruber, T. R., & Olsen, G. R. 1994. *An ontology for engineering mathematics*. In J. Doyle, P. Torasso, & E. Sandewall (Eds.), Fourth International Conference on Principles of Knowledge Representation and Reasoning, Gustav Stresemann Institut, Bonn, Germany, Morgan Kaufmann.

Newman, William H. 1975. Constructive Control; Design and Use of Control Systems, Prentice Hall, Englewood Cliffs, N.J., pp. 174.

Schank, Roger. 1984. The Cognitive Computer, Addison-Wesley, Reading, Massachusetts, pp. 268.

Sowa, J.F. and J.A. Zachman. 1992. *Extending and Formalizing the Framework for Information Systems Architecture*. IBM Systems Journal 31 (3): 590-616.

Sanchez and Mahoney. 1996. *Modularity, Flexibility, and Knowledge Management in Product and Organization Design*. Strategic Management Journal 17 (Winter Special Issue): 63-76.

The Knowledge Management Forum. Accessed 5/11/97. *What is Knowledge Management?* Available on the WWW at: http://revolution.3-cities.com/~bonewman/what_is.htm.

Zachman, J.A. 1987. *A Framework for Information Systems Architecture*. IBM Systems Journal 26 (March): 276-92.

WorkBrain: Merging Organizational Memory and Workflow Management Systems

Christoph Wargitsch
Bavarian Research Center for Knowledge-Based Systems (FORWISS)
Am Weichselgarten 7, D-91058 Erlangen-Tennenlohe

Thorsten Wewers, Felix Theisinger
University of Erlangen-Nuremberg
Computer Science Research Group B
Martensstr. 3, D-91058 Erlangen

Abstract. Despite the enthusiasm, the workflow management idea faces currently, some problems occur when setting up large workflow applications for complex business processes. To solve some of these problems, a combination of workflow management concepts and the notion of "organizational memory information systems" is suggested. The basic idea is to create an evolutionary workflow management system using an organizational memory storage component consisting of a workflow case base to save the workflow lessons learned and a storage for the general domain knowledge of an enterprise. The concept and a prototypical implementation of the system are presented. The example workflow we use to illustrate the system functions is the inquiry/proposal process of a roller bearing manufacturer.

1 Introduction

Organizational Memory Information Systems (OMIS) can be seen as a tool to support the management of knowledge in an enterprise. One of the distinct characteristics of an OMIS is to support a temporal integration of the knowledge an enterprise generates [StZw95]. Workflow Management Systems (WMS) are an instrument to support the execution of business processes. They provide the possibility to save knowledge along the time axis since workflows have an intrinsic time-logic structure and furthermore mirror the present business context of a certain period of time. Therefore, it seems to be a promising approach to connect the concepts of Organizational Memory (OM) and Workflow Management and merge their technical implementations – the OMIS and the WMS.

Since business environments are obviously dynamic systems, OMIS and WMS must be flexible enough for allowing frequent changes. They have to be able to evolve and improve with time. In this paper, we introduce an approach for an evolutionary OM-based WMS. It is supposed to help enterprises to get error-free, effective and efficient, IT (information technology)-enabled business processes. Key elements of the WMS are a process-oriented OM storage layer and OMIS functions which have the purpose to retain know-how that is generated permanently while running the WMS and ensures progress along a process learning curve. The OM is kept almost automatically updated since it is permanently fed with information from currently performed business processes. The tasks of the OM-based WMS are divided into three parts: increasing mastery of a company's workflows, continuous im-

provement of the workflow quality, and retention of professional knowledge. These tasks are interdependent. The stepwise mastery of workflows, for example, contributes to an efficiency improvement. In turn, the weakness analysis in the course of the continuous improvement process could lead to a reduction of workflow variety - simpler workflow structures result. The retention of professional and organizational knowledge is the foundation to accomplish the other tasks since it saves "the lessons learned" and ensures proceeding on the learning curve.

2 Workflow Management

The Workflow Management Coalition (WfMC), an international association of WMS producers, consultants and WMS users, defines workflow management as: "The management of processes through the execution of software whose order of execution is controlled by a computerized representation of the process" [WfMC94].

The life cycle of a workflow can be divided in different phases that usually are summarized into two blocks [MoRW96]: The workflow design phase, also called build time, and the workflow execution phase (run time). The build time contains business process analysis, business process modeling, and workflow modeling. The run time consists of workflow planning, workflow start, workflow execution and control, and workflow archiving. WMS are considered to be one of the most expanding software markets. Analysts predict an annual 47 percent growth rate for the North American and European market until 1998 [NN95]. Especially in Germany the market is booming, particularly for document-oriented systems [Yaki96]. Though the

enthusiasm in IT divisions of enterprises is given, workflow management faces currently some problems in "business reality":

Workflow Modeling Problems and Complexity. Many workflow management projects revealed that it is difficult to obtain a detailed picture of the real processes including all procedures, official and implicit rules [Herr95; Iten96; MüSc96; p. 7; Scot94]. Workflow users often state they do not recognize "their" workflow although they got involved in the modeling process by interviews. Different interpretation of terms and gaps between real acting and the information about acting are some of the reasons why organizational analyses only provide an incomplete picture of things [Gapp93]. Additionally, and according to this, business processes are often more complex than assumed [Jost96].

Lack of Flexibility. When managing business processes with the help of WMS the danger to freeze them is acute [AuBo96, p. 9]. The high costs to implement a WMS additionally hampers necessary changes. Complementary to the long-term flexibility, it is important to be able to react properly to exceptions and disturbances within a single business case. Only very few commercial systems are able to do so [Schn96]. Inflexibility and clumsiness often result from the fact that workflows are not designed modularly but consist of large, cumbersome units.

Loss of Know-how. For technology-based companies the retention and quick dissemination of know-how increasingly is an important production factor and critical success factor. However, it turned out to be problematic to collect knowledge, store it appropriately, and retrieve it to solve actual problems within workflows. Additionally, changes of all kind, especially the breaking up of grown organizational structures and the reduction of management levels in the course of business process reengineering measures may cause a loss of know-how. Retrograde steps on the learning curve result. Particularly the middle management turned out to be a key player: technical and organizational knowledge is often highly concentrated here [Fais96, p. 5].

A combination of WMS and OMIS ideas and the creation of an evolutionary, OM-based WMS should help to remedy at least some of these problems.

3 Organizational Learning and Organizational Memory

Learning organizations are characterized by actively supporting learning processes of their members and continuous self-development [Pedl89]. The objective of the learning organization is to recognize and perform necessary change processes itself. Certainly, this institutionalized learning and adaptation culture assumes a "learning culture" which fosters innovations and creativity [BuSc96].

3.1 OL

"Learning organizations ... purposefully ... enhance organizational learning" [Dodg93]. "Organizational Learning (OL)" means the learning of an organization that leads to performance improvements. OL refers to gaining know-how as well as to "lessons learned" in its negative and positive sense. Klimecki states that "problem solving networks" that are aligned crosswise to hierarchies and divisions are the most important carriers of organizational learning processes. We will show that the semantics of workflows could be taken for such networks.

Argyris and Schon define OL briefly as "the detection and correction of errors" [ArSc78]. Basically, they distinguish two types of OL: "single-loop learning" and "double-loop learning". Single-loop learning is reactive and means an organization is able to fix errors. Double-loop learning exceeds this by the ability to question procedures, rules, norms, and strategies. Paper and Johnson as well as Nevis et al. emphasize that OL includes personal learning of single members of an organization but goes beyond it [NeDG95; PaJo96]. Therefore, OL is more than the sum of individual learning results [Bala97; Dodg93]. Nevertheless the question arises where learning organizations actually retain their knowledge. "... for organizational learning occur, learning agents' discoveries, inventions, and evaluations must be embedded in *organizational memory*" [ArSc78].

3.2 OM

The term "Organizational Memory" is not coherently used in the literature ("there are as many perspectives on OM as authors") [Acke96]. Argyris and Schon take OM merely as a metaphor that is an aid to explain the behavior of organizations ("organizations do not literally remember"), whereas Sandelands and Stablein grant organizations cognitive capacities and consequently take OM for independent [WaUn91]. One of the most cited definition was made by Walsh and Ungson: "... organizational memory refers to stored information from an organization's history that can be brought to bear on present decisions" [WaUn91]. This basic definition is extended by Stein: OM lead to a higher effectiveness of an organization, under some circumstances also to a lower one [Ste95].

3.3 OMIS

Whereas OM is a conceptual term, OMIS try to support this concept with information technology. It is difficult to mark off OMIS from conventional information retrieval systems, databases, and general resources like organizational handbooks [AkSt96]. The ambition to create with OMIS shared knowledge spaces that span the entire knowledge of an organization is not realistic [ScBa92]. The main problem is to interpret the stored data correctly if the knowledge domain is large. Instead, tools that support an "OM in the small" have

turned out to be useful in order to perform certain tasks in an organization more effectively [AcMa95].

4 Basic Concepts

4.1 Framework: Double-loop Learning with a WMS

WMS are usually introduced in a company in the course of a reorganization project. They are often the most important information system to put organizational innovation into action. Among others, the main task of the suggested evolutionary WMS is to serve as an instrument for supporting the redesign of business processes. Reorganization can be divided into two basic forms: a radical reengineering (revolutionary) [Dave93; HaCh93] and a continuous improvement process (CIP, evolutionary) [Harr91; Imai94]. In order to support the evolutionary reorganization approaches, a WMS must be able to grow and mature rather than to be a self-learning system.

The double-loop learning approach of Argyris and Schon [ArSc78] is taken to be the basic concept of the evolutionary overall system "WMS + organization". The WMS is an instrument for a learning process within the domain "design, configuration, and execution of business processes". Two learning cycles can be elaborated (see figure 1): an inner learning cycle where a *learning by doing* is performed, and an outer learning cycle which implies reflexive, observed learning - *learning by supervision*. Both cycles are supported by WMS functionalities. Core element is an OM storage layer with two categories of contents: In the first place, it contains historical workflows that are stored with their audit data in form of cases, secondly domain-specific knowledge about the organization, products, and technologies, e.g. an organization database that contains the organizational structure, target measures, business rules, responsibilities etc.

Primary goal of the inner learning cycle is to increasingly master the planning and execution of workflows. This works as follows: Upon their introduction, workflows are not modeled completely and in detail. Merely the rough structure and workflow building blocks are pre-designed. For each business case within this rough structure the workflow model is configured out of these building blocks and can be modified during execution (see also sections 4.2 and 6.4). This way we obtain workflows that are stored in a case base after their execution and serve as templates for new business cases.

By *learning by supervision*, the outer learning cycle ensures a continuous improvement of workflow plan-

ning and execution and contributes to an improvement of the basic conditions. Prerequisite is to recognize weaknesses and deficiencies. When they are identified, personnel that is responsible for the workflows or the participating employees themselves can correct the system e.g. by changing target values, business rules, and the activity network. This guarantees a closed control cycle. Additionally, the outer cycle has the function to reach strategic goals with the help of the WMS. The major part of this paper refers to the inner cycle.

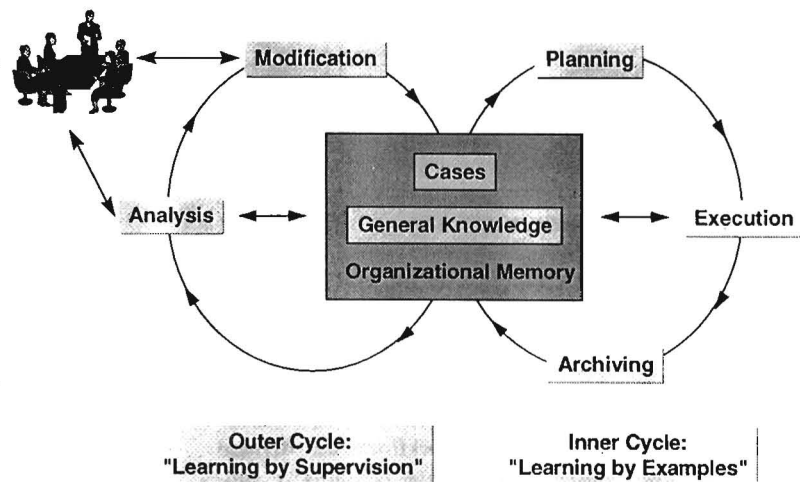


Fig. 1: Double-loop learning with an evolutionary WMS

4.2 Workflow Building Blocks and Case-oriented Workflow Configuration

For complex workflow-enabled business processes the authors introduce a new approach that substitutes the conventional workflow phase model that strictly distinguishes between build and run time [MoRW96]. Complex, partly-structured workflows can barely be mapped to a few basic workflow models. On the other side, it is unrealistic to supply a complete workflow model for each special case. Firstly, not each exception can be modeled a priori, secondly, a huge number of variants would result. Therefore, there exists a notable gap between the WMS requirements "easy handling" and "standardization" on the one hand and the need to adequately map the complex business reality on the other hand. A solution for this type of problem is a modularization. We do not think of a componentware approach and take the WMS software to pieces rather than modularize the workflows itself.

For this, workflows are dismantled horizontally and vertically. Since also users themselves should modify workflows, it is necessary to use a plain meta model. Theoretical terms like frames, patterns, class hierarchies etc. appear deterrent and should not be presented to the user. A pragmatic, clear structure helps to divide workflows into "comprehensive chunks".

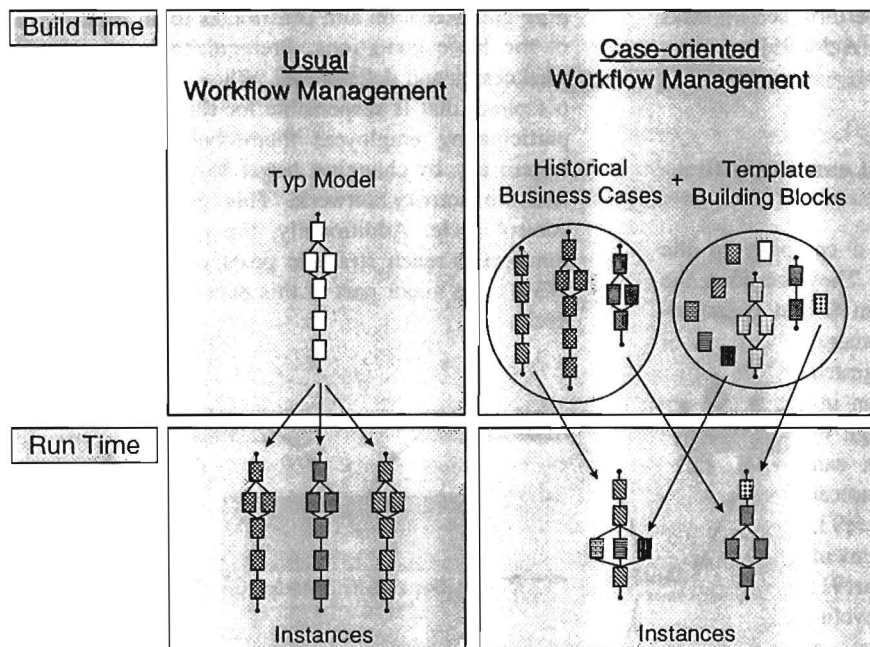


Fig. 2: Case-oriented workflow configuration of workflow building blocks

The following structure is suggested (see figure 3): Workflows consist of building blocks of various granularity. At the top level, workflows are divided into workflow phases. Each phase contains a sub-workflow, consisting of a network of activities. The term "phase" indicates that workflow phases should mainly be structured as a sequence. At the finest level, activities consist of elementary actions. For each type of building block there is a catalog, the items of the catalogs are either instances of building blocks used in historical workflows or generic templates. The various

building blocks have to be configured suitably for each business case (see figure 4). The *case-oriented workflow configuration* reduces the complexity of workflow models drastically compared to closed workflow type models. The concept of case-oriented workflow configuration is supported by a prototypical workflow engine we implemented, "FLEXWARE" [WaWe97]. The basic idea for the engine was to strictly separate the "bare workflow engine" and the workflow specification data. The configuration is supported by special search and retrieval functions and a case-based reasoning component.

The suggested approach results in a modified workflow phase model that differs from the

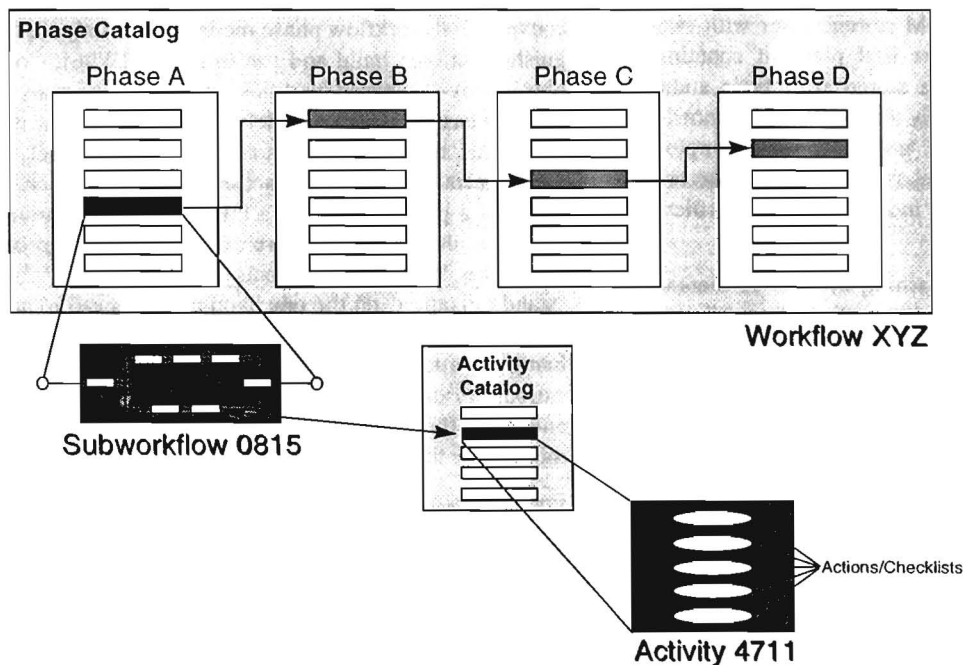


Fig. 3: FLEXWARE model

conventional one described in section 2. Business process analysis and business process modeling in their existing forms are substituted by a single analysis and modeling of the rough workflow structure and the necessary workflow building blocks. The exact design of a workflow is created when it is started – exactly specified according to the requirements of the current business case. At this point of time it can be tailored to the requirements of the current business case. Thus, the life cycle phases "workflow specification" and "workflow planning" merge to the phase "workflow configuration". The conventional separation of workflow type and single workflow instance fades away. For this reason, a modification cycle is shortened drastically and is executed at the implementation level, not at an abstract business model level. Furthermore problems, typically occurring when transforming business process models into workflow models, vanish. Nevertheless it is not possible to drop the analysis and modeling entirely, since at least "germs" for the maturing of workflows have to be present. Starting point are workflow building blocks which are held in

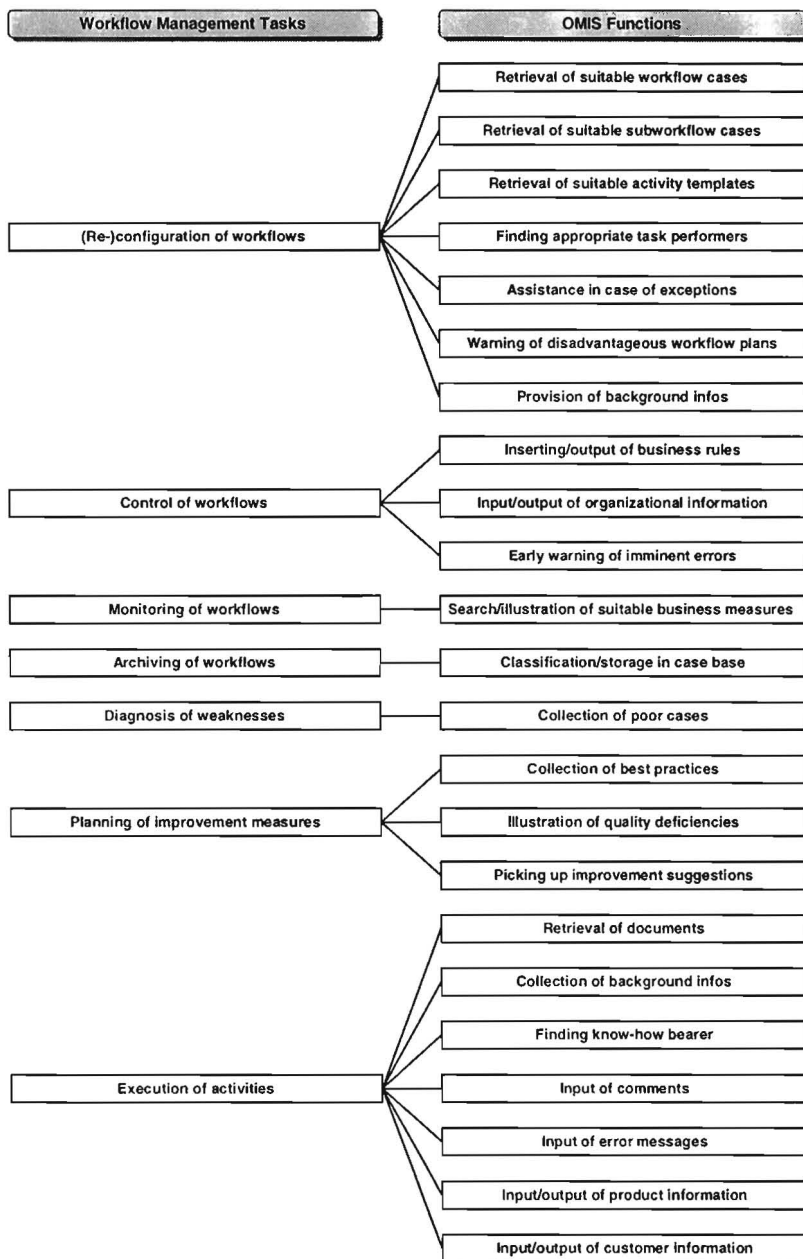


Fig. 4: OMIS functions for workflow management tasks

libraries and which can be configured freely (see figure 2). Currently, we are performing case studies with customers of our industry partner COI in order to set up "starting catalogs" which reflect typical building blocks for certain types of processes and industries.

The concepts of workflow modularization and *case-oriented workflow configuration* are important prerequisites for an OM-based WMS that is flexible enough to cope with the problems mentioned in section 2.

4.3 OMIS Functions to Support Workflow Management Tasks

To effectively support the tasks of an evolutionary workflow management with OMIS functions it is nec-

essary to design the structure and contents of the OM storage layer in a process-oriented way.

One aspect of process orientation is to support forward and backward coupling of know-how, i.e. to transport know-how along a workflow in and against the flow direction [Ste93, p. 32]. Intelligent information systems that enable BPR have to support a "reallocation of knowledge" in most early business process stages [Hart96]. For the INA inquiry/proposal process - which we will present in section 6.1 - this means e.g. that an engineer (design phase) might have information from the scheduling department (cost estimation phase) about how to design a bearing in order to produce it economically.

In turn, know-how can be transported in forward direction with a workflow, e.g. it could be beneficial to send hints about technical problems along with the business case folder although it might not contribute to a workflow output directly but perhaps has an influence on the product quality. We implemented a couple of functions to

support the forward and backward coupling of know-how (see section 6.4)

Besides these special know-how-coupling functions, there have to be a lot more OMIS functions which foster the processing of workflow knowledge and, in turn, assist the workflow management itself.

The tasks of workflow management can be divided into two groups:

1. Professional tasks related to single activities of a workflow
2. Workflow configuration and control tasks

Both categories have to be assigned to OMIS functions. In figure 4 possibilities to support workflow management tasks with OMIS functions are shown. We implemented most of these functions prototypically (for a description of some of these see section 6). Functions where users have to fill in information actively in order to obtain a rich and up-to-date OM are critical. At consulting companies like McKinsey & Company [Kiel93; Pete92] and the audit firm Arthur Andersen & Company [Quin92], incentive systems have been implemented in order to encourage the employees to contribute to the OMIS' success. A second possibility to overcome this obstacle is to integrate functions which provide information and functions which require the user's input.

5 Implementation

5.1 Technical Architecture

The basic system we use is the document and workflow management system BusinessFlow 3.3 of our industry partner COI GmbH. It is coded in a C++-like language: OEL (Object-oriented Extensible Language), a proprietary language designed by COI [COI96]. We made two severe modifications of the system's architecture: First of all, we exchanged the workflow engine FLOWARE against FLEXWARE. It is based on graphical description files and a database-oriented control mechanism. Secondly, we implemented a web-based user front-end, called WAX (Web-AXessed Workflow Management). For the latter we added a special HTTP server to the existing server architecture, also implemented in OEL. The server is tightly coupled with the DMS/WMS functions and is able to use the whole functionality of the basic system and FLEXWARE. Its functioning can be described like this: Each URL request to the server is transformed into a method call of the DMS/WMS. The system outputs the desired information – folders, documents, workflow status reports etc. – which the server transforms to HTML code, sometimes enriched with Java applets, and returns it to the browser. Therefore, the major part of the original client functionality of BusinessFlow and all of the OMIS functions could be mapped to a WWW browser. Each WMS user receives a start URL leading him to the login page. According to his permissions, e.g. the right to change workflow models, specific functionality is offered to him.

For displaying certain graphical information like activity networks and business charts we implemented Java applets. These and other graphics are provided by a second server, the freeware HTTP server Apache.

5.2 Logical Architecture

In principle, our system consists of three logical layers: the storage layer, the service layer, and the user interface. Each component within the layers is described in the following sections. The architecture is quite similar to most of the client/server-type information systems.

5.2.1 Storage Layer

The storage layer includes several databases and file systems which can be clustered into three categories. The *workflow storage* contains a workflow case base where completed workflows are stored. Each case consists of extended feature vectors which are used for a similarity search of workflows, stories and remarks about the workflow. Additionally links to a description file that contains the graphical representation of the workflow and its process logic is provided. Furthermore, audit data like cycle times, processing times, and costs which are logged are part of the workflow memory. Beyond this episodic knowledge, the workflow memory holds the building block catalogs (workflows, sub-workflows, activities, checklists, application system calls) and a "general knowledge database" where the product groups of INA, the different kinds of engineering applications, the market areas, and the responsibilities are stored. A rule base is used to decide for what combination of these attributes of a business case which organizational unit should perform which activity within workflows. The *organizational storage* mainly holds the organization database (roles, rights, units, and positions) and a bunch of "organizational documents" like ISO 9000 handbooks. The *technical storage* consists of databases that belong to KODAS (a mechanical design database), MEDIAS (an electronic product catalog), and TADDY (Technical Application Documentation and Description System) as well as all kinds of technical documents like drawings, norms, and calculation sheets that are stored in the DMS.

5.2.2 Service layer

The service layer components are either directly related to storage layer components like KODAS and MEDIAS or serve for different storage elements like the HTTP server. Some types of application systems are clearly INA-specific or at least typical for an engineering environment. The service layer can be divided into three basic sections: The control/basic services, the information services, and the communication services. Document server, update server, the database server, the short message, and electronic mail mechanism are components which are part of BusinessFlow. KODAS and MEDIAS are systems we integrated into our prototype. All the other components have been implemented in our project and added to the prototype. ExperienceFlow is a case-based reasoning (CBR) application that is used for workflow planning. TADDY is a know-how database for engineering solutions. Each TADDY document folder contains problem descriptions, problem solutions, and the related drawings. The discussion forum WIBIS (Workflow Issue-based Information System) serves as a communication platform to discuss technical and organizational problems.

The user interface is explained in the next section with the help of an example workflow.

6 Usage of WorkBrain in the INA-Inquiry/Proposal Process

The OMIS user interface components are part of WAX. With WAX we made almost all "regular" workflow client functions available in a WWW browser. We are able to: model and configure workflows, monitor workflows, provide to-do lists, show workflow document folders, edit and upload documents, and offer application systems that should be used for certain activities.

It seems to be natural to use a WWW browser as the user interface for all OMIS functions, too. It integrates all kinds of application systems inside and outside a company homogeneously, elegantly, and at little costs. This closeness and homogeneity of information and the possibility to connect informational chunks to stories enforce the imagination of the user to navigate through a unified knowledge space. Some authors speak of the WWW as global brain [MaBa94]. It is transparent for the user where the information comes from.

Our system has – of course – a smaller focus. However, since the multiple relationships and links between all the information categories resemble kind of a system of neurons and synapses we called the OMIS part of our prototype "WorkBrain". We tried to visualize the information network in form of a "knowledge map" which is clickable. Each icon symbolizes an information category. The notion of a "workflow" defines the semantics of the network.

One element of the user interface is the "control panel" of WorkBrain. It provides access to all parts of the storage layer components except the technical application systems and databases. Extended search and retrieval functions are given.

6.1 Process Example

In order to illustrate the concept and the system functionality we consider the inquiry/proposal process for special bearings of our industry partner, INA Wälzlager Schaeffler KG in Herzogenaurach. INA produces roller bearings, motor elements, and linear-guidance systems for car manufacturers and the ma-

chine tool industry. Standardized catalog bearings as well as customer-specific bearings are produced. The set up of a proposal for a customer-specific bearing is organized in the process phases depicted in figure 5 details of the process are described in [MoRW96].

The inquiry/proposal process has a long cycle time and passes through several hierarchy levels of INA. It consists of well- and poorly- structured process parts. This spread is challenging for a WMS designer. The processing of inquiries needs a lot of know-how. The involved employees are mostly highly-qualified specialists with long-term experience. The high specialization results in a strongly functional orientation and a ramified organizational structure. The process costs for completing a proposal are about US-\$ 3,000. According to habits in the machine tool industry, these costs are not billed. The process is important because special bearings often become catalog bearings and special system solutions that are generated within the process are used as competitive weapons. The process leads to innovation, and the developments launched by the customer ensure that INA stays close to market needs. Even though the catalog business has a higher revenue, the special bearings are of importance because a lot of customers want to get all bearings they need from a single source. The complexity of the process can simply be shown by some figures: About 700 people and 300 organizational units can be involved in inquiry/proposal processes. There are more than 100 elementary actions that have to be performed. The cycle time is 55 days on average whereas one business case can include up to one hundred documents. 2.000 inquiries have to be processed per year.

The next section describes the WMS functions of WorkBrain. We take the modified workflow life cycle phases to structure the description.

6.2 Design of Workflow Building Blocks

The first phase in the modified workflow life cycle is the design of building blocks. Reference process models or reference function models which mirror the characteristics of an industry seem to be a promising starting point for setting up building blocks [MoRJ94]. The first one – the process reference model – faces the

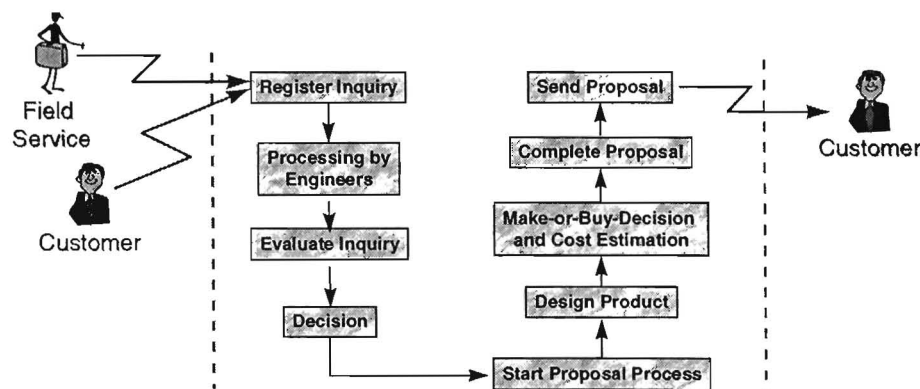


Fig. 5: Phases of the INA inquiry/proposal process

Bezeichnung:	Inzept_AT_D4_schnell	Name
Rolle:	angestellt_ott	Role
Beschreibung:	Grobkonzept schnell ers	Description
Applikationen:	Word, AutoCAD	Applications
Bedingung:	keine	Condition
Checkliste:	HTMLATGrobK	Checklist
Typ:	aktuell	Status

Speichern

Fig. 6: Creation of activity variants

problem to be either too abstract (then the information needed to design workflows is too poor) or it is too specific (then what is left over after an adaptation to the own situation tends to zero). The second one – the function reference model – does not provide any process logic. The authors suggest the already explained approach of *reference building blocks* to overcome these problems: Process logic can be provided at the desired level (workflow or phase level) and the building blocks can be continuously shifted between the abstract and the concrete level. The building blocks stem from three sources:

- a sample library containing real case studies of our workflow project of our industry partner COI,
- a template library which contains generalized elements from the case studies,
- and an enterprise-specific database which consists of parts of the sample database and the case study databases, both modified according to the INA process requirements.

The only library that is released to be modified is the enterprise-specific library. It contains building block templates for the INA inquiry/proposal process and instances (the workflow case base). This library is evolving with time and can be adjusted continuously. New building block variants can easily be created by copying and changing existing ones. Figure 6 shows how a variant of an activity is created. The father-son-relationships of building blocks are stored in order to be able to trace its evolution. Each building block can have the status "active", "obsolete", or "planned". Due to the outer learning cycle presented in section 4.1 it is necessary to align the building blocks with higher level business tactics and strategies. It is a future task of our work to systematize the realization of strategic goals at the operative workflow level while simultaneously reacting to the recognized weaknesses of the current workflows and external influences.

6.3 Workflow Configuration

The start phase of a workflow instance is split up into three parts: the workflow triggering, the automated workflow pre-configuration, and the manual fine configuration. An inquiry/proposal process is triggered by an inquiry a customer or a field service employee posts. In WAX both groups have dedicated access to special forms where they can fill in data like the customer name, the desired product category, the application of the product etc. By sending this HTML form to the WAX server a workflow folder is created, automatically pre-configured, and the workflow is started and routed to the first person handling the workflow. The pre-configuration already uses know-how saved in the OMIS storage: Depending on a set of parameters, a default template workflow is combined of building blocks according to business rules found in a pre-configuration rule table. The desired proposal date, e.g., is decisive for the "length" of the workflow: If this date is closer than ten days to the present date then in the workflow phase "cost estimation" a special sub-workflow is assigned (a shortened estimation procedure that drops the detailed handling of the expected manufacturing costs at the involved plants). Additionally to the regular role mechanisms, we implemented case-dependent roles that may override default roles subject to certain parameter constellations. Parameters are the customer size, the customer region, the industry, the product category, and the application category. Figure 7 shows the role "abt_leiter_ant" which is chosen if the customer is large, located in South/West Germany, the product type is "Axial-Nadelkranz", the application is "Fahrwerk", and the industry is "Personenfahrzeuge". If this combination is valid for a workflow, the role is taken for every activity having the default role "angestellt_ott".

The first employee dealing with the workflow is the "workflow planner", a person who can be seen as the counterpart of the classical "work scheduler" for manufacturing operations. He takes a look at the specification of the pre-configured workflow and makes changes if necessary. The first thing he has to do is to search workflow cases which might be logical predecessors of the current case. For instance, an order by a customer is related to an offer he received from INA several months ago. At INA people think very much in "business cases". They even sometimes remember the arbitrarily chosen project numbers of past inquiries, the so-called "F-number". Old business cases and their related engineering solutions are often been taken to solve actual problems. To use existing process know-how for his changes, the workflow planner has two possibilities: He can regularly search the workflow case base or use a CBR component. The result of the case search is the complete specification of (one or more) historical workflows. Suitable building blocks of this specification can be reused by copying, pasting, and adapting it to the current business case. Figure 8 shows the search mask for historical workflows.

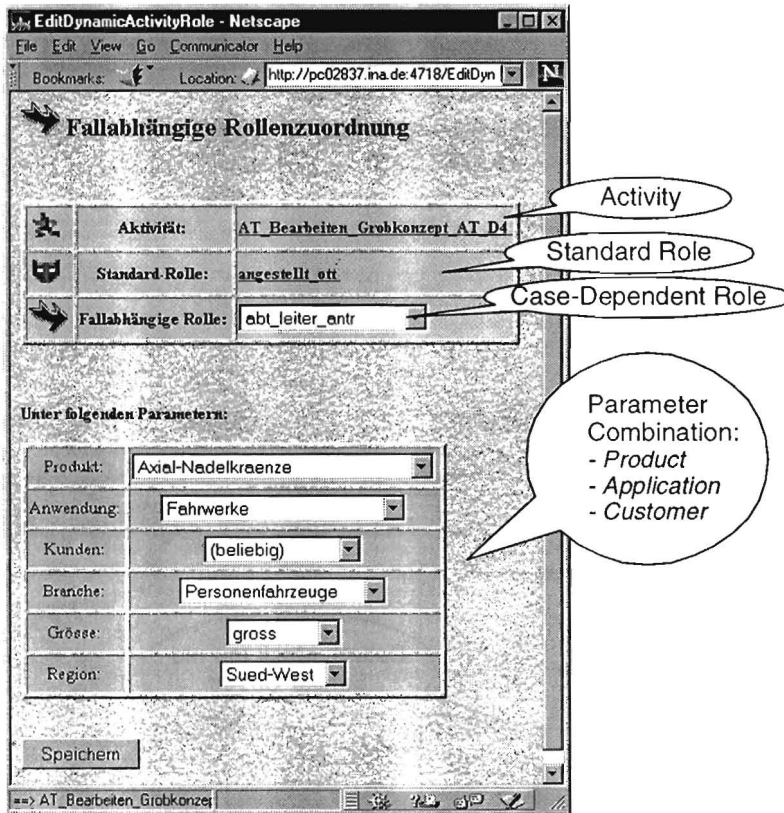


Fig. 7: Modification of case-dependent roles

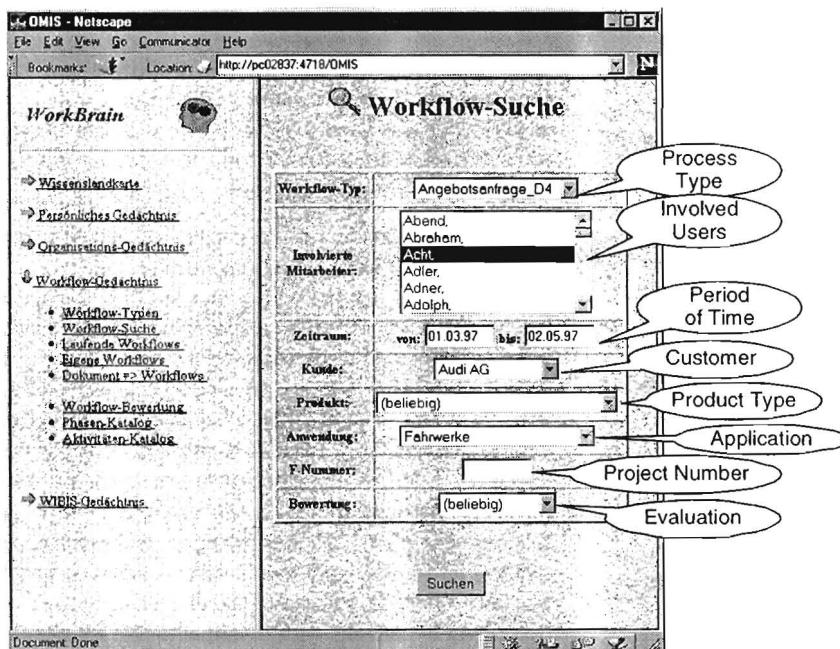


Fig. 8: Workflow search mask

Much of the organizational knowledge is implied in the activities, that are performed in a company, and their

relationship to each other (the workflow phases). In WorkBrain, all workflow activities can be searched and browsed. Since it is important to get a visual impression of a workflow we implemented a Java applet that illustrates the activity network. We use different layers to visualize the workflow structure. At the top level, each rectangle represents a workflow phase that contains a sub-workflow. With a mouse click on the phases, the underlying sub-workflow pops up in a different window.

The part *organization memory* of the WorkBrain control panel provides access to an organizational database. The database can be browsed entirely. Each relation between information categories is represented by hyperlinks. For a role, e.g., the corresponding users, activities, and rights are hyperlinked. For the workflow planner it is easy to get an overview of the multiple relationships in the organization and manually change activity-role-assignments if necessary.

The second possibility for the workflow planner to access workflow knowledge is the CBR component that is coupled via DDE (dynamic data exchange). It uses flat feature vectors to find workflow cases. Different similarity measures can be chosen. Results of a CBR search are past workflows. The process logic, the graphical description file, the documents, and the plan data can be taken to configure a new workflow or to just take a look at the solutions and copy the drawings, business data etc. Accordingly, sub-workflows and activities can be found and reused as building blocks in a similar manner. After the termination of a workflow, it is completely stored with all related information. Figure 9 shows the CBR search window of the workflow phase case base and the result list. A detailed description of the similarity measures and the implementation of the CBR component can be found in [Oed97].

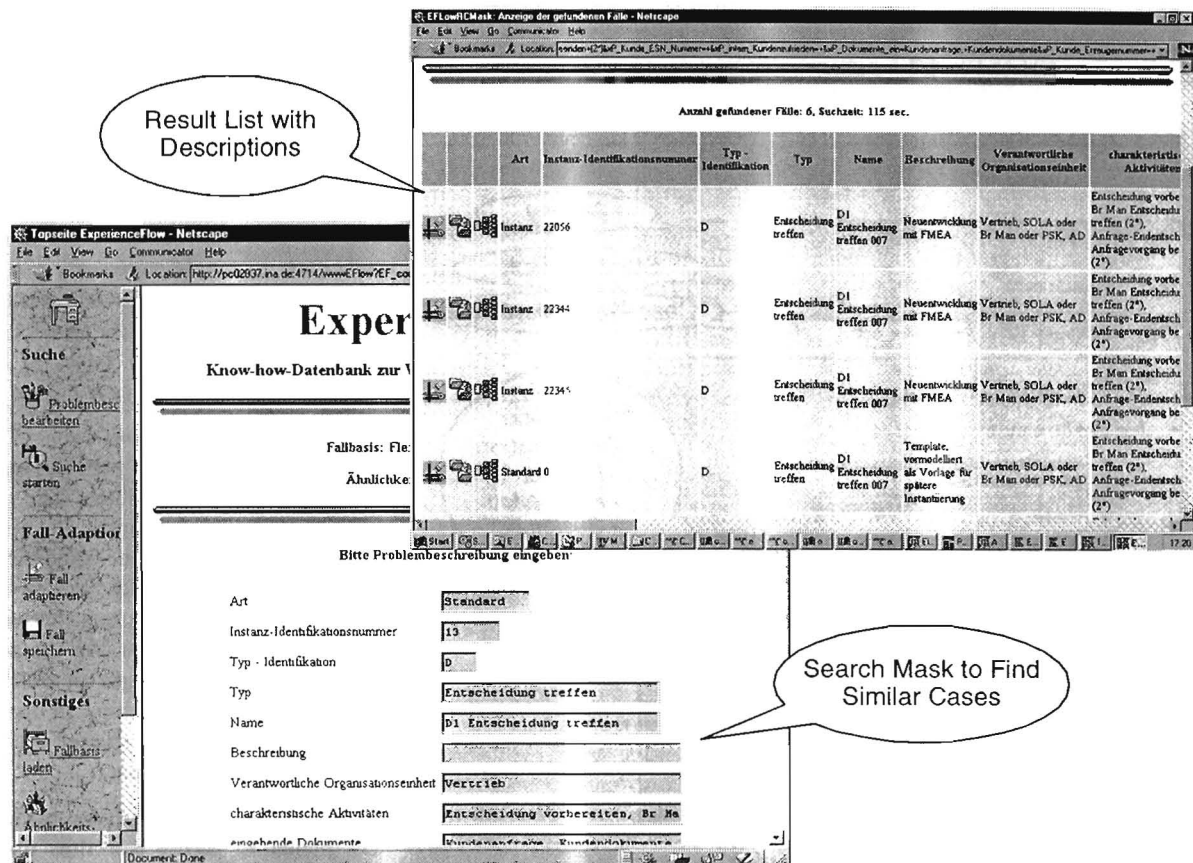


Fig. 9: CBR search and retrieval

6.4 Workflow Execution and Control

After the workflow planner terminated the workflow fine configuration, he saves it and the actual processing of the inquiry starts. The workflow folder is passed from employee to employee and is gradually filled up with documents like drawings, calculation sheets, forms, price lists, and finally the proposal letter to the customer. Due to the process's complexity and its creative nature, it must be able to change the workflow design "on-the-fly". Every user who has the right to change workflow specifications can redesign the workflow with the graphical editor and save the modified version in the engine's database. Changes can be made for all future activities and their relationships to each other. All past activities are blocked.

The rights for workflow modification are coupled with the regular rights-role administration. It is reasonable to keep the number of employees who are allowed and able to modify workflows as large as possible since disturbances and suddenly occurring needs to change the workflow structure can appear at every state of the business case. If someone else than the affected employee has to be involved in order to change the workflow, the advantage of "on-the-fly" changes is lost. On the other side, it is desirable to keep the number of workflow variants as small as possible. Therefore, only certain people should have the possibility to declare

workflow instances to be generic workflow templates. Furthermore, the top level workflow that connects the workflow phases should be "stable", i.e. not be changed for a period of time longer than one or two orders of magnitude of a single workflow's cycle time.

Each activity that has to be performed appears in the to-do lists of all users who possess the related role. If a user chooses an activity with a mouse-click, the activity disappears from all other to-do lists. A new browser window with the "to-do mask" pops up containing the workflow folder with all documents, some workflow data, and the trigger button of application systems which support the execution of the activity, e.g. TADDY. Additionally, checklists are available which have the purpose to guide the employee when performing complex activities like the technical assessment of an inquiry and also in order to document what has been done. The items of the checklists can be connected via hyperlinks with documents like ISO9000 instructions or explanations. The results of the work done with the application systems can be transferred to the workflow folder. When the activity is completed, the user closes the to-do mask and the workflow is continued. Figure 10 shows the principle of the workflow processing.

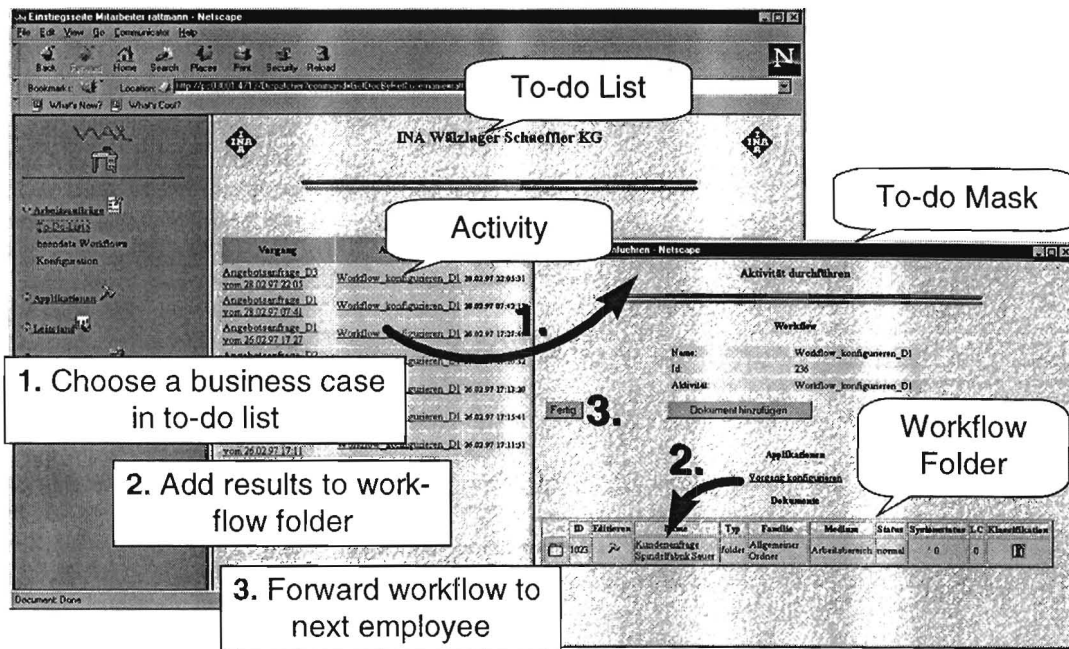


Fig. 10: Principle of workflow management with WAX

6.5 Knowledge Transfer

In order to transfer knowledge between organizational members in general and to couple know-how forwards and backwards within a business process, several mechanisms and components have been implemented:

- The storage of interdependencies between workflows is established. One of the reproaches against process orientation is that business processes are "cuts through an enterprise's body" and do not consider the relationships orthogonal to these cuts [Mert96]. We are at least able to retain the relation of subsequent processes (see section 6.3).



Fig. 11: Know-how of Samsa Gregor

- The cross-sectional transfer of know-how is supported by a special "know-how counting function". Each time a user performs a workflow activity the counter increases by one. Thereby, it is possible to find out who is experienced in certain activities. Figure 11 shows the know-how table of Samsa Gregor. Especially in a know-how-oriented enterprise like INA with a high degree of specialization, it is advantageous for a workflow planner to assign the right person to perform a task. Of course, people often simply know who has special skills for a certain task, but sometimes this is not the case and a know-how bearer storage is an aid.
- One of the most important design elements of the system is the hypermedia user interface. It allows to present the context of formerly isolated workflow information according to the philosophy: "everything is a link".
- A similar effect is accomplished by the possibility to retrieve workflows via document searches. Especially checklists which store also "soft information" like episodes, problems, etc. contribute to making procedural and technical knowledge available – detached from the strict time-logic structure of processes.
- Automated storage of knowledge along with the audited activities is just one part of the management of OM. The other element is communication that generates and distributes knowledge. The discussion forum "WIBIS" (Workflow Issue-based Information System) is based on the idea of issue-based information systems (IBIS). With this forum, it is possible to assign certain topics to roles and

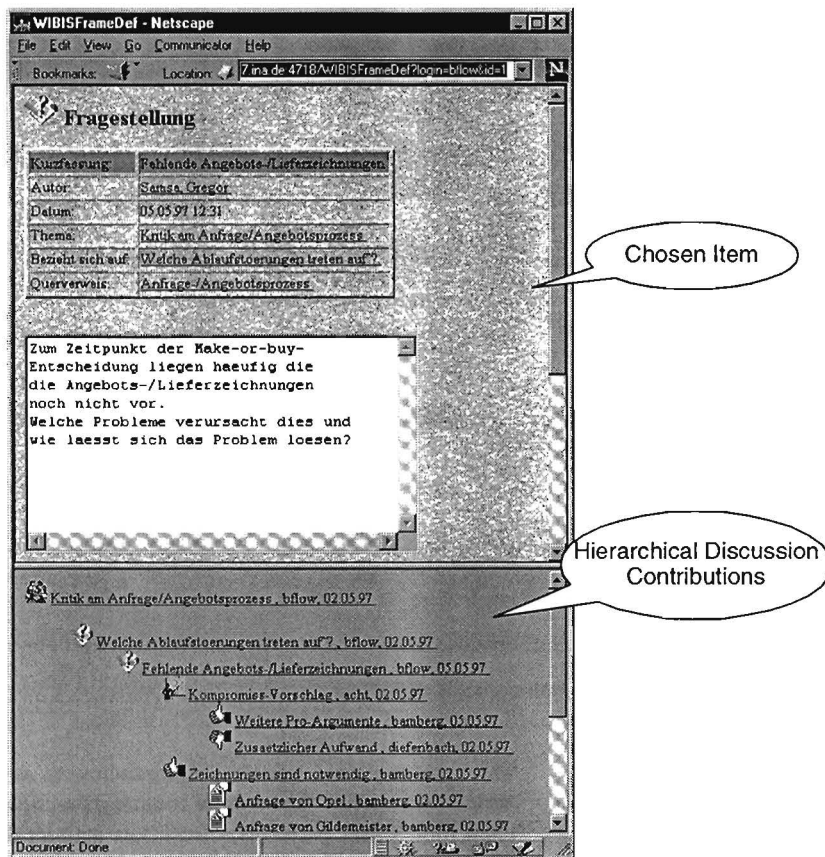


Fig. 12: WIBIS

activities. The roles automatically receive all new messages in WIBIS. Therefore, the discussion of problems can be proliferated in both directions (future and past of a workflow state) within a business process since e.g. certain roles are only "active" in confined workflow phases. The role bearer are actively notified about issues exceeding these limits. Discussion topics can be generated about business case problems (technical or commercial) or workflow management itself. WIBIS can be "started" anytime. It provides functionality to place statements and add pros and cons or examples to them (see figure 12). It is also able to set hyperlinks to certain topics or statements that point to electronic documents (filed anywhere, e.g. workflow folders of the WMS or external WWW documents) that are of some interest for the topic.

7 Summary and Next Steps

The proposed concept addressed the problems of WMS and OMIS described in chapter 2. The difficulties of both systems can be remedied by combining the underlying ideas and designing an integrated system with common functions - an OM-based evolutionary workflow system. It represents an OM in the small with the specific task to support the planning, control, and execution of complex industrial core business processes. Double-loop learning is supported by the concepts "case-oriented workflow configuration" and "hybrid

workflow management", a process-oriented OM storage structure, and special OMIS functions. Procedural and professional know-how is retained workflow-aptly in the OM. The extension of a WMS to an OMIS appears promising since daily work practice and know-how management are integrated. The main focus of the implementation was to integrate the OMIS functions into the working environment of the employees and to create an "easy-to-use" interface that presents information and functions contextually.

There are still deficiencies concerning the outer learning cycles. Work has to be done to systematize the workflow evaluation, the implementation of improvement measures, and the related OM maintenance. Also a framework for breaking down strategic goals to the workflow level is still not available.

We currently perform case studies investigating further complex project-like business processes like the inquiry/proposal process at INA. The results will be used to fine-tune the WMS prototype and to implement

further process examples and to create a workflow component library that can be used as a starting point for the workflow evolution process when implementing WMS in companies.

Parts of the concepts and solutions of our prototype are transferred to the running workflow system at INA and to the WMS products of COI: e.g. the FLEXWARE approach is used for the workflow engine of the next release of BusinessFlow.

8 Acknowledgements

This project is supported by our industry partners: *INA Wälzlager Schaeffler KG* and *Consulting for Office and Information Management (COI) GmbH* and the *Bavarian Research Center for Knowledge-Based Systems (FORWISS)*.

References

- AcMa95 Ackerman, M., Mandel, E., Memory in the Small: An Application to Provide Task-Based Organizational Memory for a Scientific Community, in: Nunamaker, J., Sprague, R. (eds.), Proceedings of the 28th Annual Hawaii International Conference on System Sciences, Vol. IV, Los Alamitos 1995, p. 323-332.
- AcSt96 Ackerman, M., Stein, E., Organizational Memory and Organizational Memory Information Systems, Tutorial, 29th Annual Hawaii International Conference on System Sciences, Wailea, Maui 1996.
- Acke96 Ackerman, M., Organizational Memory, URL: <http://www.ics.edu/ackerman/om.html>, 14th December 1996.
- ArSc78 Argyris, C., Schon, D., Organizational Learning: A Theory of Action Perspective, Reading 1978.
- AuBo96 Aulbur, W., von Bonin, T., Zementiert das System unproduktive Abläufe?, in: Standardsoftware (1996) 7/8, p. 5-9.
- Bala97 Balasubramanian, V., Organizational Learning and Information Systems, URL: <http://eies.njit.edu/333/orglrn.html>, 9.5.97.
- BuSc96 Bullinger, H., Schäfer, M., Lernende Organisationen, Oracle Welt (1996) 3, p. 4-9.
- COI96 Consulting for Office and Information Management (COI) GmbH (eds.), BusinessFlow 3.3 User's Manual, Herzogenaurach 1996.
- Dave93 Davenport, T., Process Innovation, Boston 1993.
- Dodg93 Dodgson, M., Organizational Learning: A Review of Some Literatures, Organizational Studies 14 (1993) 3, p. 375-394.
- Fais96 Faisst, W., Wissensmanagement in Virtuellen Unternehmen, Working Paper no. 8/1996 of the series "Information and Communication Systems as Design Elements of Virtual Enterprises", University of Erlangen-Nuremberg, Nuremberg 1996.
- Gapp93 Gappmaier, M., Kooperative Vorgangsbearbeitung - Eine empirische Untersuchung der Arbeitsorganisation, Working Paper of the Institute of Computer Information Systems of the University of Linz, Linz 1993.
- HaCh93 Hammer, M., Champy, J., Reengineering the Corporation, Boston 1993.
- Harr91 Harrington, H., Business Process Improvement, New York 1991.
- Hart96 't Hart, M., Matching Intelligent Systems with Business Process Reengineering, Intelligent Systems in Accounting, Finance and Management 5 (1996) 5, p. 41-53.
- Herr95 Herrmann, T., Workflow Management Systems: Ensuring Organizational Flexibility by Possibilities of Adaptation and Negotiation, in: Proceedings of the Conference on Organizational Computing Systems (COOCS), Milpitas 1995, p. 83-94.
- Imai94 Imai, M., KAIZEN: Der Schlüssel zum Erfolg der Japaner im Wettbewerb, Berlin 1994.
- Iten96 Iten, A., Workflow-Management bei Landis&Gyr, in: Österle, H., Vogler, P. (eds.), Praxis des Workflow-Managements, Braunschweig/Wiesbaden 1996, p. 253-270.
- Jost96 Jost, K., Pilotprojekterfahrungen, in: Österle, H., Vogler, P. (eds.), Praxis des Workflow-Managements, Braunschweig/Wiesbaden 1996, p. 229-238.
- Kiel93 Kiely, T., Learning to Share, CIO (1993) 7, p. 38-44.
- MaBa94 Mayer-Kress, G., Barczys, C., The Global Brain as an Emergent Structure from the Worldwide Computing Network, and its Implications for Modeling, in: Tech-Report CCSR-94-22, Center for Complex Systems Research, University of Illinois at Urbana-Champaign, 1994.
- Mert96 Mertens, P., Process focus considered harmful, WIRTSCHAFTSINFORMATIK 38 (1996) 4, p. 446-447.
- MoRJ94 Morschheuser, S. et al., Referenzmodell einer integrierten Daten- und Dokumentenverarbeitung im Industriebetrieb, dargestellt am Beispiel der kundenwunschorientierten Anfrage-/Angebotsabwicklung von Maschinenbauunternehmen, Working Paper no. 2/1994 of the Department of Information Systems I, University of Erlangen-Nuremberg, Nuremberg 1994.
- MoRW96 Morschheuser, S., Raufer, H., Wargitsch, C., Challenges and Solutions of Document and Workflow Management in a Manufacturing Enterprise: A Case Study, in: Lynn, M. (ed.), Proceedings of the 29th Annual Hawaii International Conference on System Sciences, Vol. V, Los Alamitos 1996, p. 4-13.
- MüSc96 Müller, S., Scheidt, M., Geschäftsprozessoptimierung - mehr als nur ein kurzlebiger Trend!, Industriemanagement 12 (1996) 5, p. 6-9.

- NeDG95 Nevis, E., DiBella, A., Gould, J., Understanding Organizations as Learning Systems, Sloan Management Review, Winter 1995, p. 73-85.
- NN95 N.N., Workflow-Software wird ein Renner, Computerzeitung 26 (1995) 34, p. 17.
- Oed97 Oed, J., Entwurf und prototypische Realisierung einer Know-how-Datenbank zur fallabhängigen Workflow-Planung und -Steuerung, Diplomarbeit, University of Erlangen-Nuremberg, Nuremberg 1997.
- PaJo96 Paper, D., Johnson, J., A Theoretical Framework Linking Creativity, Empowerment, and Organizational Memory, in: Lynn, M. (ed.), Proceedings of the 29th Annual Hawaii International Conference on System Sciences, Vol. IV, Los Alamitos 1996, p. 20-33.
- Pars59 Parsons, T., General Theory in Sociology, in: Merton, R. et al. (eds.), Sociology Today: Problems and Prospects, New York 1959, p. 3-37.
- Pedl89 Pedler, M. et al., Towards a Learning Company, Management, Education and Development 20 (1989) 1, p. 20-38.
- Pete92 Peters, T., Liberation Management, New York 1992.
- Quin92 Quinn, J., Intelligent Enterprise: A Knowledge and Service Based Paradigm for Industry, New York 1992.
- ScBa92 Schmidt, K., Bannon, L., Taking CSCW Seriously - Supporting Articulation Work, CSCW 1 (1992) 1-2, p. 7-40.
- Schn96 Schnitzke, M., Konzeption einer flexiblen Komponente zur Steuerung und Integration in ein Workflow-Produkt, Prediploma Thesis, University of Erlangen-Nuremberg, Nuremberg 1996.
- Scot94 Scott-Morgan, P., Die heimlichen Spielregeln: die Macht der ungeschriebenen Gesetze im Unternehmen, Frankfurt/Main 1994.
- Stei93 Steinmetz, O., Die Strategie der integrierten Produktentwicklung, Braunschweig 1993.
- Stei95 Stein, E., Organizational Memory Review: Concepts and Recommendations for the Management, International Journal of Information Management 15 (1995) 2, p. 17-32.
- StZw95 Stein, E., Zwass, V., Actualizing Organizational Memory with Information Systems, Information Systems Research 6 (1995) 2, p. 85-117.
- WaUn91 Walsh, J., Ungson, G., Organizational Memory, Academy of Management Review 16 (1991) 1, p. 57-91.
- WaWW97 Wargitsch, C., Wewers, T., Wirthmüller, A., WWW-Front-Ends for Document and Workflow Management Systems, in: Proceedings of the IT-Vision-Week, CD-ROM, Paderborn 1997.
- WaWe97 Wargitsch, C., Wewers, T., FLEXWARE: Fallorientiertes Konfigurieren von komplexen Workflows - Konzepte und Implementierung, Müller, M., Schumann, O., Schumann, S. (eds.), Proceedings of the 11th Workshop 'Planning and Configuration' on the 4th German Conference "Knowledge-based Systems" (XPS-97), Erlangen 1997, p. 45-55.
- WfMC94 Workflow Management Coalition (ed.), Glossary, A Workflow Management Coalition Specification 2/1994, Brussels 1994.
- Yaki96 Yaki, A., Germany - A high growth market, poised for major change, Document World (1996) 5, p. 14-16.

Intelligente Systeme für den Customer Support: *Fallbasiertes Schließen in Help-Desk und Call-Center Anwendungen*

Dr. Stefan Wess^{*}

Stichworte: fallbasiertes Schließen, Customer-Support, Help-Desk, Call Center, Self-Help Tools, kommerzielle Anwendungen, Fallstudien

Zusammenfassung: Im Bereich des Help-Desks, des Call-Centers und der Self-Help Tools finden vermehrt fallbasierte Systeme ihr kommerzielles Einsatzgebiet. In dieser Arbeit werden wir die Potentiale des Einsatzes fallbasierter Verfahren in diesen unterschiedlichen Bereichen des Customer Supports aufzeigen. Dazu beschreiben wir zunächst kurz das jeweilige Einsatzumfeld, die zu lösende Problemstellung und geben dann einige Fallstudien in Form von bereits kommerziell realisierten Anwendungen an.

Intelligent Customer Support Systems

Keywords: case-based reasoning, customer support, helpdesk, call center, self-help tools, commercial applications, case studies

Abstract: Recently the case-based reasoning technology is often used in helpdesk, call center and self-help applications. In this paper we will show some scenarios for the use of CBR within these areas of customer support. We will first summarize the different application areas, describe the key topics and give some case studies of implemented and daily used CBR systems.

^{*} Dr. rer. nat. Stefan Wess, TECINNO GmbH, Sauerwiesen 2, D-67661 Kaiserslautern, wess@tecmath.de

1 Einführung

Ein guter Kundenservice wird zu einem immer wichtigeren Wettbewerbsfaktor. Bedingt durch einen sehr schnellen Modellwechsel, technisch immer anspruchsvollere Produkte und Serviceleistungen werden diese erhöhten Anforderungen an den Kundenservice in vielen Fällen jedoch zu einem Personal- und Kostenproblem. Die Unternehmen stehen damit vor dem Dilemma bei sinkendem Budget einen stetig besseren Service liefern zu müssen. Eine gängige Lösung ist die Unterstützung der oft noch auf Papierbasis betriebenen Supportabteilungen und Help-Desks [Schr93] durch entsprechende Computersoftware. Das Marktvolumen für diese Anwendungen beträgt im Jahr 1995 bereits rund 250 Millionen Dollar (nach Dataquest).

Neben den rein organisatorischen Aufgaben eines computerunterstützten Customer-Support Systems wie das Aufnehmen und Weiterleiten von entsprechenden Kundenanfragen (engl. Problem Management, Call Tracking) steht inzwischen auch die eigentliche Problemlösung, d.h. die effiziente und konsistente Beantwortung der entsprechenden Kundenanfrage (engl. Problem Resolution), bei vielen Unternehmen im Mittelpunkt der Nachfrage. Vermehrt werden daher für die Mitarbeiter im Customer-Support Systeme zur effizienten Entscheidungsunterstützung gefordert [Schi91]. Im Bereich des Help-Desks und des Call-Centers finden daher inzwischen, fast unbemerkt, auch zahlreiche fallbasierte Techniken ihre kommerzielle Anwendung [Vroo91]. Auch im deutschsprachigen Raum sind bereits einige Anwendungen im Einsatz. Bedingt durch die Verständlichkeit und Flexibilität eines fallbasierten Ansatzes, kann dieses Verfahren im Bereich des Customer-Supports daher die mit Abstand höchsten Zuwachsraten vorweisen. Intelligente Customer-Support Systeme sind damit zu einem wichtigen Einsatzbereich fallbasierter Verfahren in der Wirtschaft geworden [Cole92].

1.1 Zielsetzungen

In dieser Arbeit wollen wir einige Möglichkeiten für den Einsatz fallbasierter Verfahren (engl. case-based reasoning, CBR) in drei unterschiedlichen Bereichen des Customer Supports aufzeigen. Im einzelnen wollen wir hier betrachten:

- CBR im Help-Desk
- CBR im Customer Support
- CBR beim (End-)Kunden

Dazu beschreiben wir zunächst kurz das Einsatzumfeld und die zu lösende Problemstellung, geben dann einige Fallstudien in Form von bereits kommerziell realisierten Systemen an und zeigen am Ende die Vorteile eines fallbasierten Ansatzes im entsprechenden Bereich auf. Die Arbeit endet mit einer abschließenden Bewertung der beschriebenen Ansätze.

2 CBR im Help-Desk

Die bislang wirtschaftlich erfolgreichsten Anwendungen der CBR Technologie stammen ohne Zweifel aus dem Gebiet der Help-Desk Anwendungen [Alle94, Simo91]. Hierbei bietet sich natürlicherweise der Einsatz zur Unterstützung des IT-Bereiches an. Weltweit haben zahlreiche Unternehmen inzwischen bereits entsprechende Lösungen im täglichen Einsatz [AlAu95] bzw. haben sich auf die Entwicklung von fallbasierten Lösungen spezialisiert. Fallbasiertes Schließen ist daher ein im Bereich des Help-Desk bereits etablierter Standard zur Problemlösung.

2.1 Problembeschreibung

Die Downsizing und Client/Server Anstrengungen der letzten Jahre haben zu einer Revolution in der IT-Infrastruktur vieler Unternehmen geführt. Ausgehend von einzelnen Abteilungs- oder Büro-PC's wucherte, oftmals unkontrolliert, eine Vielzahl von Insellösungen, die sich nun schrittweise auf Abteilungs- oder Unternehmensebene zu immer größer werdenden Netzwerken zusammenschließen. Die Folge davon ist, daß kaum ein Arbeitsplatz in der Verwaltung oder im Dienstleistungsgewerbe ohne PC existieren kann. Der anfängliche Traum

von der Dezentralisierung und der damit verbundenen Unabhängigkeit von der traditionellen EDV hat sich für viele Unternehmen inzwischen jedoch zu einem Alptraum im Bereich des Anwendersupports entwickelt. Die Komplexität der Produkte wächst ständig, die Einsatzdauer wird kürzer und die Schere zwischen Experten und Anwendern wird trotz aller Anstrengungen größer statt geringer. Wurde anfangs - in Eigeninitiative auf Abteilungsebene - auf die dort vorhandenen "Computerspezialisten" zurückgegriffen, so sind inzwischen zentralisierte Anlaufstellen, sog. Help-Desks (oft auch Hotline genannt) [Schr93], für genervte Computeranwender Standard in vielen Unternehmen. Die Aufgaben eines IT-Help-Desks sind dabei so vielfältig wie die eingesetzte Hard- und Software. Aus der Sicht der Anwender ist die Aufgabenbeschreibung eines IT-Help-Desks sehr schlicht: *„Alles das in Ordnung bringen, was nicht richtig funktioniert.“*

2.2 Einsatzbereich

Bedingt durch das oftmals sehr hohe Anfragevolumen in einem IT-Help-Desk kann nicht immer direkt auf entsprechende Experten zugegriffen werden, sondern es muß oft auch auf Aushilfs- und Teilzeitkräfte zurückgegriffen werden (First-Level-Support). Diese nehmen die Kundenanrufe entgegen und filtern die eingehenden Anfragen, indem sie einfache Probleme direkt lösen und nur schwierige Fälle an die entsprechenden Experten (Second-Level-Support) weiterleiten. Fallbasierte Systeme dienen in diesem Einsatzbereich zur Dokumentation, Verteilung und Verwaltung des aktuell vorhandenen Problemlösewissens und unterstützen somit die Mitarbeiter im First- und Second-Level Support bei der Bearbeitung der Kundenanfragen.

2.3 Fallstudien

In diesem Abschnitt wollen wir die folgenden Fallstudien vorstellen:

- Externes IT-Help Desk:
Compaq SMART
- Internes IT-Help Desk:
Roche IAIS

- Unterstützung von Experten:
Sommerfield CBRX
- Unterstützung vor Ort:
Sainsbury SHINE
- Weltweite Software-Hotline:
Dun & Breadstreet INSIGHT

2.3.1 Externes Help-Desk

Ein externes Help-Desk dient der Unterstützung von (externen) Kunden eines Unternehmens bei Problemen mit Produkten oder Serviceleistungen. Ein entsprechender Service wird als entweder kostenloser Kundensupport oder als kostenpflichtige Dienstleistung angeboten. Die in einem externen Help-Desk zu bearbeitenden Anfragen sind zumeist auf die Produkte und Dienstleistungen des entsprechenden Unternehmens beschränkt. Die Anfragen betreffen aber unter Umständen alle Details des diesbezüglichen Bereiches. Anwender von fallbasierten Systemen sind hier daher zumeist die Mitarbeiter im First-Level-Support, da die bei Kunden des Unternehmens üblicherweise auftretenden Probleme im eigenen Help-Desk zumeist bereits bekannt sind. Neu auftretende Probleme werden daher zunächst vom Second-Level-Support gelöst und in Form von entsprechenden Fallbeispielen im System abgelegt. Damit steht das entsprechende Wissen auch allen Mitarbeitern mit direktem Kundenkontakt zur Verfügung.

Compaq - SMART

Das SMART System beim Hardwarehersteller Compaq ist das inzwischen wohl bekannteste Beispiel für den Einsatz der CBR Technologie im Bereich des IT-Help-Desks. SMART (Support Management Automated Reasoning Technology) [AcWa92] dient dem Support von Compaq Kunden und ist über eine kostenlose Hotline erreichbar. Das System mit 50 Plätzen bereits seit 1992 in Betrieb und besitzt eine ständig wechselnde Fallbasis mit ca. 600 Fallbeispielen. Ein Vergleich bei Compaq [Alle94] mit dem Zustand vor der Einführung des Systems ergab eine um 175% verbesserte Antwortquote bei Kundenanfragen. Durch die Einführung von SMART konnte die durchschnittliche Dauer eines Anrufes

um 30% reduziert, 87% der Anrufer konnte bereits beim ersten Anruf geholfen und 95% der Kundenanfragen innerhalb eines Zeitraumes von 10 Minuten gelöst werden.

2.3.2 Internes Help-Desk

Im Gegensatz zum externen Help-Desk können bei einem internen Help-Desk die Computeranwender in einem Unternehmen Hilfestellung bei Problemen erhalten. Interne IT-Help-Desks sind daher inzwischen in nahezu allen größeren Unternehmen vorhanden. Das Aufgabenspektrum eines IT-Help-Desks reicht dabei von der Installation neuer Geräte, über die Unterstützung der Anwender bei Fragen zur eingesetzten Software bis hin zur Behebung von Netzwerkproblemen. Die hier zu lösenden Problemstellungen gehen daher eher in die Breite, statt in die Tiefe wie bei einem externen Help-Desk. Komplizierte Probleme werden zumeist vom internen Help-Desk des Unternehmens an das externe Help-Desk des zuständigen Herstellers weitergeleitet. Bedingt durch die Breite der in einem internen Help-Desk zu bewältigenden Probleme bietet sich hier jedoch ebenfalls eine Unterstützung der Help-Desk Mitarbeiter durch fallbasierte Systeme an. Die zu erstellende Fallbasis hat dabei jedoch eine gänzlich unterschiedliche Struktur: Häufig auftretende Problemfälle jeglicher Art sollen von jedem Mitarbeiter, zu jedem Zeitpunkt direkt am Telefon behoben werden können.

Roche - IAIS

Das Help-Desk im Vitaminwerk der Roche AG in der Schweiz ist für die Betreuung aller Rechner im Unternehmen zuständig. Das in einem Projekt mit einem Aufwand von ca. 6 Monaten eingeführte IAIS (Informatik Administrations- und Informations-) System [Schu95] ermöglicht das Weiterleiten der Anfragen an die zuständigen Experten, den Zugriff auf Mitarbeiter- und Inventardaten, sowie die interne Verrechnung der am Help-Desk erbrachten Leistungen. Die Fallbasis besteht aus Lösungen zu Standardanfragen (z.B. Probleme beim Anmelden im Netzwerk), aus sehr spezifischen und schwierig zu behobenden

Spezialproblemen (z.B. Konfiguration von Netzwerksoftware), sowie aus typischen Anwenderfragen zur im Unternehmen eingesetzten Standardsoftware (z.B. Probleme mit der Formatierung in Excel). Zur effizienten Beantwortung der Anfragen können auch die entsprechenden Abschnitte der Hilfedateien aus dem CBR System aufgerufen und beispielsweise per Email an den betroffenen Anwender weitergeleitet werden. In einem Zeitraum von 5 Monaten wurden bereits über 1800 Anfragen mit der Hilfe von IAIS gelöst.

2.3.3 IT-Experten

Ist der Aufgabenbereich beschränkt, so können fallbasierte Systeme auch zur direkten Unterstützung von IT-Experten eingesetzt werden. Hierbei wird der entsprechende Anwendungsbereich fast vollständig durch entsprechende Fallbeispiele abgedeckt. Eine solche Vorgehensweise bietet sich beispielsweise bei unternehmenskritischen Anwendungen mit hohen Verfügbarkeitsanforderungen an.

Sommerfield - CBRX

Sommerfield, eine der größten Supermarktketten in Großbritannien, verfügt über einen 7 Tage/24 Stunden IT-Support Service für seine 430 Märkte mit über 4600 elektronischen Kassensystemen. Das interne IT Help-Desk bietet zudem Support für die mehr als 1000 PC Anwender im Unternehmen. Nur durch die Unterstützung der IT-Betreuer und Techniker mittels geeigneter Fallbeispiele war es laut Sommerfield möglich insgesamt 261 Märkte in nur 9 Monaten auf neue elektronische Kassensysteme umzustellen.

2.3.4 Unterstützung vor Ort

Zur Entlastung des Help-Desks kann eine Fallbasis auch allen IT-Anwendern im Unternehmen zur Verfügung gestellt werden. Die Suche nach potentiellen Lösungen und die Möglichkeit zur direkten Problembehebung wird hierbei „vor Ort“ verlagert, so daß selbst unerfahrene Anwender in der Lage sind, Routineproblem selbst zu lösen, ohne daß dazu eine Anfrage an das zentrale Help-Desk des Unternehmens notwendig wäre.

ARGOS

Argos, der größte Versandhandel in Großbritannien, mit außerdem über 300 Verkaufsstellen im ganzen Land, erhält rund 9000 Anrufe im Monat an seinem internen IT-Help-Desk. Viele der gemeldeten Probleme lassen sich sehr einfach lösen und resultieren im wesentlichen aus der Tatsache, das Argos sehr viele Teilzeitkräfte einsetzt, die zumeist nicht über das notwendige Fachwissen verfügen. Mit einer Fallbasis von nur 300 Fallbeispielen, konnten nach der Einführung des Help-Desk Systems bereits über 60% der Anfragen ohne Hilfe des zentralen Help-Desks gelöst werden.

Sainsbury - SHINE

Eine vergleichbare Lösung mit dem Namen *SHINE (Self Help IN-store Expert)* ist inzwischen auch in den Märkten der Supermarktkette Sainsbury (ebenfalls Großbritannien) im Einsatz. Im zentralen Help-Desk des Unternehmens werden zur Zeit etwa 1400 Anfragen pro Tag beantwortet. Mit der Einführung von SHINE soll diese Zahl um etwa 30% gesenkt werden.

2.3.5 Softwarehotline

Die Entwicklung von spezifischen fallbasierten Lösungen im IT-Support ist dann besonders erfolgreich, wenn die Ergebnisse eines solchen Projektes gegebenenfalls weltweit verwertet werden können. Werden beispielsweise bereits im Rahmen der Softwareentwicklung bzw. des Softwaretests entsprechende Fallbasen entwickelt, so können diese, insbesondere im Bereich der Help-Desks zur Softwareunterstützung, sehr leicht zur Erkennung von Softwarefehlern und zur Verbreitung bzw. Verwaltung von entsprechenden Updates verwendet werden.

D & B Software - INSIGHT

Dun & Breadstreet Software, ein bekanntes Unternehmen im Bereich der Software für das Finanzwesen, unterstützt seine 12000 Kunden weltweit durch den On-Line Service INSIGHT. Im Rahmen eines 2 Jahre dauernden Projektes wurden von 25 Dun & Breadstreet Mitarbeitern aus verschiedenen Niederlassungen 15 Fallbasen für den Cu-

stomer Support entwickelt, wobei jede ca. 200 - 500 Fallbeispiele enthält. Eine Untersuchung nach Einführung des Systems ergab, daß ein neuer Mitarbeiter, ohne jede Erfahrung mit D&B Produkten, mit Hilfe des Systems bereits ca. 70% der Anfragen korrekt beantworten konnte.

2.4 Bewertung

Die hier aufgeführten Fallstudien beschränken sich zwar auf den Bereich des IT-Help-Desks, die entsprechenden Ergebnisse lassen sich jedoch auf nahezu jedes Help-Desk, insbesondere auch auf den gesamten technischen Support übertragen [KrBa93]. Die zahlreichen Beispiele von Systemen im täglichen Routineeinsatz belegen auch, daß sich das fallbasierte Schließen in der Praxis bereits als eine brauchbare Technologie zur Unterstützung von Mitarbeitern im Help-Desk erwiesen hat. Aus der Sicht der Anwender dieser Technologie ist dies nach unserer Meinung auf die folgenden Vorteile eines solchen Ansatzes zurückzuführen:

- Der nötige Aufwand zur Bearbeitung einer Kundenanfrage sinkt.
- Ein großer Anteil der Anfragen ist bereits beim ersten Anruf des Kunden lösbar.
- Neues Problemlöswissen wird dokumentiert und kann leicht an alle Mitarbeiter weitergegeben werden.
- Die Einarbeitung neuer Mitarbeiter wird wesentlich erleichtert.
- Fallbasierte Lösungen können mit geringerem Aufwand entwickelt werden.
- Entsprechende Werkzeuge sind seit längerem kommerziell verfügbar.
- Die Projektkosten amortisieren sich zumeist in weniger als einem Jahr.

Nahezu die gesamte heute verfügbare Help-Desk Software verfügt daher bereits über einen fallbasierten Ansatz zur Problemlösung oder bietet zumindest Schnittstellen zu den kommerziell verfügbaren Lösungen an. Aus der Sicht der Entwickler fallbasierter Systeme zeichnet sich der Anwendungsbereich Help-Desk durch die folgenden Vorteile aus:

- Viele Probleme im Bereich des Help-Desks lassen sich relativ einfach lösen.

Die Erstellung einer entsprechenden Fallbasis ist daher verhältnismäßig einfach zu realisieren.

- Die Zahl der zum erfolgreichen Einsatz notwendigen Fallbeispiele ist beschränkt. Die bekannten Probleme mit der Effizienz des Retrievals treten in realen Anwendungen daher zumeist noch nicht auf.
- Die Symptome der Anwenderprobleme lassen sich zumeist sehr gut am Telefon erfragen und in das System eingeben.
- Die von einem fallbasierten System geführte Suche nach Problemlösungen eignet sich besonders für die im Gegenstandsbereich u.U. noch relativ unerfahrenen Anwender des First-Level Support.

Als ein wichtiges Ergebnis dieses Abschnittes kann daher festgehalten werden, daß fallbasiertes Schließen ein inzwischen allgemein akzeptierter und kommerziell erfolgreicher Standard zur Problemlösung im Bereich der computerunterstützten Help-Desk-Anwendungen ist [Simo92].

3 CBR im Customer-Support

Das Help-Desk ist sicherlich ein wichtiger und aktueller Bereich des Customer-Supports. Insgesamt bezeichnet dieser Begriff jedoch einen wesentlich größeren Bereich von Aktivitäten in einem Unternehmen. Die Unterstützung des Customer-Support durch den Einsatz fallbasierter Anwendungen ist daher das Thema dieses Abschnittes.

3.1 Problembeschreibung

Service ist mehr und mehr ein Differenzierungsfaktor im Wettbewerb um den Kunden und ein starkes Werkzeug der Kundenbindung. Dabei wird die Kundenzufriedenheit wesentlich durch eine verbesserte Kundenbetreuung, ein einheitliches Erscheinungsbild zum Kunden sowie die fachliche Kompetenz der Kundenberatung geprägt. In vielen Unternehmen rückt daher die Qualität des Customer-Supports inzwischen vermehrt in den Mittelpunkt des Interesses. Bei komplexen Produkten und immer kürzeren Innovationszyklen werden die erhöhten

Anforderungen an den Customer-Support jedoch oft zu einem Personal- und vor allem zu einem Kostenproblem. Die Herausforderung stellt sich daher heute in einer Effizienzsteigerung bei einhergehender Kostenreduzierung auf einem hohem, gesicherten Qualitätsniveau.

3.2 Einsatzziel

Die zunehmende Produktvielfalt und Informationsflut erschwert in vielen Unternehmen den zielgerichteten Einsatz von Wissen und Erfahrung in der Betreuung des Kunden. Neben den technischen Produktinformationen ist es vor allem der aus der täglichen Arbeit gewonnene Erfahrungsschatz, den es gilt, systematisch zu dokumentieren und allen im Kundenkontakt stehenden Mitarbeitern zugänglich zu machen. Gerade solche Anforderungen und Aufgabenbereiche sind aber eine Stärke fallbasierter Ansätze [Houl95]. Fallbasierte Systeme können daher die Mitarbeiter des Customer-Supports effizient unterstützen und es so ermöglichen die in diesem Bereich gesteckten Ziele zu erreichen.

3.3 Fallstudien

Im folgenden sollen einige Fallstudien aus verschiedenen Bereichen des Customer-Supports angegeben werden, die dies belegen können:

- Business-Cases im Call-Center:
London Electricity & Southern Electric
- After-Sales Management:
Nokia - ASMS
- On-Line Troubleshooting:
Reuters - Calls
- Produktinformation:
Black & Decker

3.3.1 Call-Center

Der große Bereich der telefonischen Kommunikation mit dem Kunden wird inzwischen in vielen Unternehmen in sog. Call-Centern zentralisiert abgewickelt. Beispiele hierfür sind, neben den bereits im vergangenen Abschnitt erwähnten Help-Desks, die Bestellannahme, die Fahrplanauskunft, die Info-Hotline oder auch der Bereich des Tele-Banking. Dient ein Call-Center nur ei-

nem bestimmten Zweck, wie beispielsweise der Aufnahme von Adressen bei einem Preisausschreiben, so sind die entsprechenden Vorgänge und die notwendigen Informationen noch relativ einfach zu durchschauen. Sind jedoch viele unterschiedliche und zum Teil komplexe Vorgänge zu bearbeiten oder ändert sich die Informationslage häufig, so muß die Qualität des Customer-Supports durch eine entsprechende Qualifikation der Mitarbeiter bzw. durch eine geeignete Koordination des Informationsaustausches gesichert werden. Genau in diesem Bereich können fallbasierte Ansätze Impulse bei der Organisation und der Automatisierung des Call-Center Betriebes geben.

London Electricity

London Electricity bietet seinen 1,7 Millionen Kunden einen 7 Tage/24 Stunden Service zur Meldung von Störungen und zur Behandlung von Kundenanfragen. Die über 100 Mitarbeiter im zentralen Call-Center von London Electricity bearbeiten zur Zeit etwa 50.000 Anfragen pro Woche. Ziel ist es dabei alle ankommenden Anrufe nach einer Wartezeit von höchstens 15 Sekunden anzunehmen und eine effiziente und vor allem konsistente Beantwortung der entsprechenden Kundenanfrage zu ermöglichen. Das zur Erreichung dieser Ziele entwickelte System arbeitet im wesentlichen fallbasiert, wobei die Falbasis neben Störungen und Fehlerbeschreibungen vor allem eine Vielzahl von für das konkrete Call-Center typischen Vorgängen, sog. *Business-Cases* (z.B. an/ab und ummelden von Kunden) enthält. Hierbei werden die entsprechenden Vorgänge, bzw. die Informationen die zur Bearbeitung der Vorgänge notwendig sind, in Form von Fallbeispielen repräsentiert und bei Bedarf abgearbeitet. Der Vorteil eines solchen Ansatzes liegt in der schnellen Realisierbarkeit und vereinfachten Wartbarkeit eines entsprechenden Softwaresystems. Dies wird auch eindrucksvoll durch die Tatsache unterstrichen, daß das gesamte System bei London Electricity in nur 6 Wochen installiert und in Betrieb genommen werden konnte. Trotz des vermehrten Einsatzes von Teilzeit- und Aushilfskräften können inzwischen 95%

der Anfragen ohne die Weiterleitung des Anrufers direkt bearbeitet werden (vgl. Financial Times, 17 Oktober 1994).

Southern Electric

Ein mit der Lösung bei London Electricity vergleichbares fallbasiertes System, genannt PHAROS, wurde inzwischen auch bei Southern Electric, für die 500 Mitarbeiter im dortigen Call Center in Betrieb genommen. Sie bearbeiten mit PHAROS die telefonischen Anfragen der über 2.5 Millionen Kunden des Unternehmens in Großbritannien.

3.3.2 After-Sales Management

Jedes Problem eines Kunden ist eine Chance zur Verstärkung der Kundenbindung. After-Sales-Support wird damit mehr und mehr auch zu einem Marketinginstrument. Der weltweit führende Hersteller im Bereich der mobilen Kommunikation NOKIA hat dies erkannt und seinen Customer-Support durch den Einsatz eines fallbasier-ten Systems entsprechend ausgerichtet

NOKIA - ASMS

Im November 1994 startete bei NOKIA in Düsseldorf die Pilotphase des After-Sales-Management-Systems (ASMS). Basis des Systems ist ein fallbasiert arbeitendes „Problem and Trouble Management and Resolution Tool“, welches um spezifische Module zur Verwaltung von Kontaktpersonen, zum Ressourcen-Management, Monitoring und Informationen über die „Installed Base“ erweitert wurde. Das System soll weltweit im 7 Tage/24 Stunden Customer-Support von NOKIA eingesetzt werden und befindet sich derzeit in verschiedenen Ländern in der Roll-Out Phase (vgl. NOKIA Digit 8/94).

3.3.3 On-Line-Fehlerdiagnose

Die Nachrichtenagentur Reuters betreibt ein umfangreiches, weltweites Netzwerk zur Verteilung der Ware Nachricht an ihre Kunden. Die Stabilität und Verfügbarkeit des Netzes ist daher einer der kritischen Faktoren im Geschäft von Reuters. Aus diesem Grunde betreibt Reuters weltweit zahlreiche Customer-Support Centren an die die entsprechenden Ausfälle im Netz-

werk gemeldet werden. Die dort eingeleiteten Maßnahmen bestehen im wesentlichen im Umleiten der Verbindungen auf alternative Routen, seltener in der Bereitstellung von entsprechenden Servicetechnikern, die aufgetretene Fehler vor Ort beheben.

Reuters - CALLS

Das bei Reuters eingesetzte System Calls besteht neben Modulen zur Administration der Anfragen aus einer umfangreichen Fallbasis mit typischen Fehlern und deren Lösungen. Sie wird zur Zeit weltweit von Customer-Support Mitarbeitern in London, Genf und New-York entwickelt. Ein wesentlicher Teil der Fallbasis besteht jedoch auch aus temporär gültigen Empfehlungen für Alternativrouten. Wird von einem Mitarbeiter im Customer Support ein Problem im weltweiten Netzwerk aufgenommen und eine entsprechende Lösung entwickelt, so legt er diese in der Fallbasis von Calls ab. Der aufgetretene Fehler und die mögliche Lösung steht somit allen Mitarbeitern im Help-Desk augenblicklich zur Verfügung. Da bei einem Ausfall wesentlicher Teile des Netzwerkes das Anrufvolumen im Help-Desk von Reuters oft unvermittelt stark ansteigt, ist diese Form der effizienten Kommunikation zwischen den Mitarbeitern wesentlich für einen optimalen Kundensupport.

3.3.4 Produktinformation

Black & Decker hat mit über 4000 Produkten eine breite Palette von Produktinformationen zu verwalten. Kundenanfragen zu bestimmten Produkten wurden im Customer-Support von Black & Decker daher zunächst an den entsprechenden Produktspezialisten weitergeleitet. Diese ineffiziente und von der ständigen Verfügbarkeit der entsprechenden Spezialisten abhängige Arbeitsweise, wurde inzwischen durch den Einsatz eines fallbasierten Systems im Customer-Support von Black & Decker nahezu vollständig abgeschafft.

Black & Decker

Eine zentrale Fallbasis enthält alle Informationen über Einsatzbereiche, Preise und mögliches Zubehör aller Black & Decker

Produkte. Die Fallbeispiele werden ergänzt durch weitere Produktinformationen und Händlerlisten, die aus dem System direkt angefordert werden können. Allen Mitarbeitern im Customer-Support von Black & Decker ist es damit möglich, zu jeder Zeit eintreffende Kundenanfragen über Anwendung, Materialbeschaffenheit, Ersatzteillieferung bis hin zu technischen Details zu beantworten. Anfragen von Kunden zu den unterschiedlichen Produkten können so schnell und ohne die Zuhilfenahme von Produktspezialisten beantwortet werden. Mit Hilfe des seit 1994 eingeführten Customer-Support Systems konnte bei Black & Decker die Zahl der bearbeiteten Anrufe um ca 60% gesteigert werden, wobei die Zahl der notwendigen Rückrufe beim Kunden um etwa den gleichen Anteil gesenkt werden konnte.

3.4 Bewertung

Im Bereich des Customer-Supports sind zahlreiche, zum Teil sehr unterschiedliche Aufgaben zu bewältigen. In den hier angegebenen Fallstudien konnte daher nur ein geringer Teil dieser Aufgabenbereiche aufgegriffen werden. Allen Aufgaben im Customer-Support ist aber gemein, daß ausgehend von den im allgemeinen sehr unspezifischen und unvollständigen Informationen einer Kundenanfrage auf eine konkrete Antwort oder Aktion geschlossen werden muß, wobei sich zudem der Inhalt und die Art der Anfragen häufig wiederholt und daher ausreichende Erfahrung im Gegenstandsbereich das wichtigste Kapital eines Mitarbeiters in diesem Bereich darstellt. Fallbasierte Ansätze sind daher eine geeignete Technologie um die vielfältigen Aufgaben im Customer-Support durch eine geeignete Softwaretechnologie zu unterstützen [BaMo92]. Für den Anwender ergeben sich dabei die folgenden Vorteile:

- Der Aufwand zur Bearbeitung von Anfragen sinkt, mit gleichbleibendem Personal kann so ein größeres Anfragevolumen bearbeitet werden.
- Eine effiziente und vor allem konsistente Beantwortung der entsprechenden Kundenanfrage wird ermöglicht.

- Die Qualität des Supports ist weitgehend unabhängig vom Mitarbeiter, d.h. die Einhaltung eines einheitlichen und gleichbleibenden Servicestandards wird möglich.
- Änderungen in den zu bearbeitenden Vorgängen bzw. in den zugrundeliegenden Informationen sind relativ einfach zu realisieren.

Als Anwendungsbereich für fallbasierte Systeme ist der Bereich des Customer-Supports, bedingt durch die Vielfältigkeit der zu bearbeitenden Aufgabengebiete für Entwickler und Forscher auf dem Gebiet des fallbasierten Schließens sehr attraktiv.

4 CBR beim Endkunden

Sowohl im Bereich des Help-Desk als auch im Customer-Support allgemein bereitet das zu bewältigende Anfragevolumen organisatorische, personelle und vor allem auch finanzielle Probleme. Da fallbasierte Systeme bereits erfolgreich zur Unterstützung des Customer-Supports eingesetzt werden, liegt es nahe, einen Teil des Supportaufwandes wieder zum Kunden zu verlagern, indem ihm ebenfalls eine entsprechende Technologie zur selbständigen Problembewältigung zur Verfügung gestellt wird. Mit unterschiedlichen Strategien, wie dieser Ansatz realisiert werden kann, wollen wir uns in diesem Abschnitt beschäftigen.

4.1 Problembeschreibung

Bedingt durch die Komplexität der Produkte bzw. Dienstleistungen und die damit steigende Zahl an möglichen Supportanfragen wird eine schnelle und vollständige Information von Anwendern und Entwicklern immer wichtiger. Gedruckte Produkt- und Wartungsinformationen sind heute jedoch in einem Umfang vorhanden, die für Kunden eine zielgerichtete und effiziente Suche nach benötigten Informationen immer schwieriger werden lässt. Um die wachsende Belastung der Supportabteilungen bei Erhaltung der bereits erreichten Kundenzufriedenheit zu reduzieren, müssen geeignete Mittel zur Kanalisierung und Strukturierung der Informationsflut gefunden werden. Wie auch bereits in den anderen Bereichen des

Customer-Supports können hierbei fallbasierte Systeme helfen.

4.2 Fallstudien

Neben dem Help-Desk und weiteren Anwendungen im Customer-Support ist das Gebiet der sog. „Self-Help Tools“ ein weiterer Wachstumsmarkt auf dem fallbasierte Ansätze kommerzielle Erfolge erzielen [Mot93]. Die folgenden Fallstudien können dies belegen:

- Call -Avoidance:
Compaq - QuickSolve
- Knowledge-Publishing:
IBM - AskPSP
- Embedded Systems:
Microsoft Windows
- CBR On-Line:
Inference WWW-Server

4.2.1 Call-Avoidance

Das bekannteste Beispiel aus dem Bereich der Self-Help Tools ist das QuickSource Projekt bei Compaq. Ziel von Compaq war es, die ständig wachsende Zahl von Anfragen an den Telefonsupport durch geeignete Hilfsmittel direkt an der Quelle, d.h. beim Kunden, einzudämmen. Eine solche Strategie für den Customer-Support wird auch als *Call-Avoidance* Strategie bezeichnet. Fallbasierte Systeme stellen ein geeignetes Hilfsmittel zur technischen Realisierung einer solchen Strategie dar.

Compaq-QuickSolve

Basierend auf den Erfolgen des SMART Systems, war QuickSolve [NgCzLe93] von Compaq die logische Fortführung des Ansatzes und gleichzeitig das erste fallbasierte System welches mit einem Produkt an Endkunden und Händler ausgeliefert wurde. Das System unterstützt die Installation und Fehlersuche bei der PAGEMARQ Netzwerkdrucker Serie von Compaq. QuickSolve war dabei jedoch nur ein Teil einer großangelegten Customer-Support Initiative von Compaq, die mit einer Reihe von Softwaretools ihren Kunden eine verbesserte Unterstützung bei Problemen mit Compaq-Produkten geben wollten. Dieses Software-

paket mit dem Namen QuickSource umfaßte neben dem fallbasierten System QuickSolve eine elektronische Produktdokumentation (Quick-Tour), eine Konfigurationshilfe (Quick-Config), sowie Hypertext- (QuickHelp) und Multimediasysteme (QuickTutorial). Die Fallbasis von QuickSolve umfaßte dabei ca. 500 Fallbeispiele und wurde aus entsprechenden Kundenanfragen an das SMART System generiert. Neben der Version für Endkunden, welche mit dem entsprechenden Gerät ausgeliefert wurde, gab es zusätzlich eine erweiterte Version für den Einsatz bei Händlern. Diese Version wurde quartalsweise durch Updates auf den neusten Stand gebracht. Die Auslieferung von QuickSource startete bereits im September 1992. Bedingt durch die hohe Zahl an ausgelieferten Druckern (mehrere zehntausend) ist es damit das zur Zeit wohl am meisten verbreitete fallbasierte System. In einem abschließenden Vergleich mit vergleichbaren Produkten, die ohne ein Call-Avoidance Tool wie QuickSource ausgeliefert wurden, ermittelte Compaq, daß die Anzahl der Anfragen an das Help-Desk bei PAGEMARQ Druckern um ca. 20% unter dem Durchschnitt der vergleichbaren Produkte lag. Die entsprechenden Einsparungen bei Compaq übertrafen die Investitionen in QuickSource bei weitem. Bedingt durch eine Änderung in der Geschäftspolitik wurde die Druckerlinie PAGEMARQ von Compaq inzwischen eingestellt. Die erfolgreiche Call-Avoidance Strategie wurde aber für alle Produkte übernommen. Ein mit QuickSource vergleichbares Tool mit dem Namen EZSearch, wird inzwischen mit allen Compaq Geräten ausgeliefert.

4.2.2 Knowledge-Publishing

Aus dem allgemeinen Trend zum Einsatz fallbasierter Systeme im Bereich der Help-Desk und Self-Help Anwendungen ist inzwischen selbst ein prosperierender Markt, genannt „Knowledge-Publishing“, geworden.

ServiceWare - KnowledgePaks

Hierzulande zumeist noch unbemerkt, sind amerikanische Firmen (z.B. ServiceWare oder KnowledgeBroker) in den Markt ein-

gestiegen und bieten professionell entwickelte Fallbasen mit Problemen und Lösungen zu populärer Anwendungssoftware käuflich an. Diese werden dabei sowohl von Privatanwendern als auch von internen IT-Help-Desks zur Problemlösung eingesetzt. Inzwischen sind bereits über 50 (Wissens-) Pakete (engl. KnowledgePaks) dieser Art auf dem Markt. Die Anwender der Systeme sind dabei zur Weiterentwicklung dieser Fallbasen aufgerufen. Für ein in die neue Version übernommenes Fallbeispiel erhält der Anwender von ServiceWare eine Prämie von 10 Dollar.

IBM - AskPSP

Ein vereinfachter Zugang zu aktuellen Produktinformationen und Bug-Fixes für OS/2 und LAN Produkte war das Ziel bei der Entwicklung des *On-Line Help-Desks* AskPSP durch die IBM Personal Software Products (PSP) Division. Das Wissen und die Erfahrung von zahlreichen Entwicklern und Anwendern von IBM Produkten wurde hierzu zunächst in Form von unstrukturierten Dokumenten auf eine CDROM gepreßt. Zur Unterstützung einer zielgerichteten Suche im umfangreichen Informationsbestand wurde anschließend eine Fallbasis entwickelt, die als Problemlösung auf das entsprechende Dokument verweist. Der Einsatz eines fallbasierten Systems ermöglicht hier, im Gegensatz zu konventionellen Information-Retrieval Techniken, bei mehreren potentiellen Problemlösungen, die gezielte Rückfrage des Systems an den Anwender, um so zwischen unterschiedlichen Lösungsansätzen differenzieren zu können. Eine erste Version der AskPSP CDROM mit ca. 1000 Fallbeispielen wurde von IBM im August 1993 an die 2000 Teilnehmer einer Entwicklerkonferenz in Orlando verteilt (vgl. IBM Fact Sheet, 3 August 1993).

4.2.3 Embedded Systems

Einen Schritt weiter als IBM mit AskPSP ist inzwischen der Softwarehersteller Microsoft. In zukünftigen Versionen von Microsoft Produkten soll eine auf Fallbeispielen basierende Unterstützung von Anwendern integriert werden.

Microsoft - Windows

Ein erster noch recht einfacher Ansatz, in diese Richtung wurde mit dem Microsoft Troubleshooter bereits in Windows 95 realisiert. Die eher passive Unterstützung des Anwenders in Form der heute bereits üblichen On-Line Dokumentation und Hilfe verschiebt sich damit mehr zu einer aktiven und selbständigen Problembehebung durch das betreffende System selbst. Es ist damit zu rechnen, daß in Zukunft auch die Anwendungssoftware von Microsoft und, in Folge damit praktisch jede Software, mit einer Fallbasis von bereits bekannten Problemen und deren Behebungsmöglichkeiten ausgeliefert wird. Die Behebung von auftretenden Problemen, sei es automatisch oder durch die Mithilfe des Anwenders, wird dadurch wesentlich vereinfacht (vgl. Microsoft News Release, 7 September 1993).

High-Tech Produkte

Die Möglichkeit zur Integration von fallbasiertem Problemlösungswissen ist jedoch nicht nur auf Produkte aus dem IT-Bereich beschränkt. Nahezu jedes High-Tech Produkt wie etwa Werkzeugmaschinen, Produktionsautomaten oder auch Flugzeuge werden mit einer umfangreichen Dokumentation, welche auch Anweisungen zur Problembehebung einschließt, ausgeliefert. Diese Informationen können zumeist leicht in entsprechende Fallbasen umgesetzt werden. Dieser Einsatz von fallbasierten Systemen im Produkt selbst ist auch technisch kein größeres Problem, da Mikroprozessor gesteuerte Systeme mit entsprechenden Benutzeroberflächen inzwischen eher die Regel als die Ausnahme in allen Arten von High-Tech Produkten sind.

Der Embedded Systems Markt ist daher ein ebenfalls sehr wichtiges Anwendungsgebiet für fallbasierte Systeme. Obwohl wir hierzu im Augenblick noch keine konkrete Fallstudie angeben können, sind entsprechende Anwendungen bereits in der Planungsphase.

4.2.4 CBR On-Line

Alle in diesem Abschnitt angesprochenen Strategien zur Verteilung von Wissen in Form von Fallbeispielen haben den entscheidenden Nachteil, daß auftretende Änderungen im Fallbestand natürlich auch an die Kunden weitergegeben werden müssen. Dies wird zumeist über automatische Updates erreicht, die beispielsweise monatlich oder quartalsweise erfolgen können. Für den Zugriff auf aktuelle Falldaten bietet sich jedoch nur ein Zugang über entsprechende Kommunikationsnetze an. Erste Anwendungen dieser Art im Bereich des On-Line Shopping sind bereits in der Vorbereitung. Der CBR2 Server von Inference ist beispielsweise für den Zugang über das WWW (vgl. www.inference.com) bereits verfügbar. Der Zugriff auf die in einer Fallbasis abgelegten Informationen ist für die Kunden eines Unternehmens über bereits heute durch über Tonwahl-gesteuerte CBR-Server mit Fax-back Services bzw. seit kurzem auch durch die direkte Spracherkennung über Telefon möglich. Wie bereits im IT-Bereich, wo der Customer-Support durch die Bereitstellung von Mailboxsystemen zum On-Line Zugriff auf Treiber und Updates, revolutioniert wurde, so wird diese Möglichkeit auch in anderen Branchen die direkte Kommunikation mit und den Support von Kunden wesentlich vereinfachen. Fallbasierte Systeme können dabei helfen, die entsprechenden Informationen für Kunden strukturiert zur Verfügung zu stellen.

4.3 Bewertung

Die großflächige Verbreitung fallbasierter Systeme an Händler, Servicetechniker und Endkunden ist die logische Fortsetzung der in den ersten beiden Abschnitten beschriebenen Einsatzszenarien fallbasierter Systeme im Customer-Support. Entsprechende Ansätze sind dabei im wesentlichen durch den steigenden Kostendruck oder durch den Zwang zur Aktualität der zur Verfügung gestellten Informationen motiviert. Werden bereits entsprechende fallbasierte Anwendungen im Customer-Support eingesetzt, so können die entsprechenden Fallbasen mit

geringem zusätzlichen Aufwand auch an Kunden ausgeliefert werden.

5 Abschließende Bewertung

Aus der Sicht der Entwickler und Forscher auf dem Gebiet ist fallbasiertes Schließen eine geeignete Problemlösestrategie in Bereichen

- in denen zwar Wissen über wesentliche Hintergründe und Zusammenhänge, aber kein vollständiges Model vorhanden ist.
- in denen ähnliche Situationen immer wieder vorkommen und in denen die Erfahrung aus früheren Situationen hilft, das aktuelle Problem besser zu lösen.

Beide Eigenschaften sind typisch für diejenigen Anwendungsgebiete, in denen Erfahrung eine mindestens genauso entscheidende Rolle wie theoretisches Wissen spielt [Wess95]. Untersucht man unter diesem Blickwinkel den weiten Bereich des Customer-Supports, so ist dieser ein nahezu idealer Einsatzbereich für fallbasierte Anwendungen. Betrachten wir zusätzlich noch die folgenden Punkte, so wird auch der kommerzielle Erfolg dieser Technologie verständlich:

- Die Unterstützung des Customer-Supports durch entsprechende Softwaresysteme ist gegenwärtig noch nicht weit fortgeschritten. Selbst einfache Lösungen bewirken daher einen prozentual sehr großen Fortschritt in der Produktivität. Die im Vergleich dazu geringen Projektkosten amortisieren sich somit sehr schnell.
- Fallbeispiele sind intuitiv sinnvolle und gut handhabare Einheiten. Die meisten Probleme im Customer-Support lassen sich als eine Suche nach entsprechenden Fallbeispielen formulieren.
- Das Prinzip des fallbasierten Schließens ist relativ einfach zu verstehen und an die Mitarbeiter zu vermitteln. Die vom System geführte Suche nach Problemlösungen hat sich auch für unerfahrene Anwender als sehr gut handhabbar erwiesen.
- Fallbasierte Systeme lassen sich relativ einfach entwickeln und warten. Der

Wissenserwerb ist zudem im Bereich des Customer-Supports i.a. meist unproblematisch und damit sehr effizient möglich. Die Verfügbarkeit von kommerziellen Werkzeugen beschleunigt den Entwicklungsprozeß erheblich.

- Hintergrundwissen und Anpassung sind in den meisten Anwendungsbereichen im Customer-Support gegenwärtig noch nicht notwendig. Fallbasierte Werkzeuge der ersten und zweiten Generation sind daher bereits ausreichend zur erfolgreichen Projektdurchführung.
- Die Zahl der zur erfolgreichen Problemlösung notwendigen Fallbeispiele sind zumeist beschränkt. Für einige Bereiche gibt es gegenwärtig bereits Fallbasen käuflich zu erwerben oder diese werden vom Hersteller bereits mitgeliefert.

Vorteile im Wettbewerb werden heute fast ausschließlich über Know-How und Erfahrung realisiert. Oft werden diese Potentiale in den Unternehmen jedoch noch nicht ausreichend genutzt. Archivierte Fallbeispiele, als Ausgangsmaterial für fallbasierte Anwendungen, sind in vielen Unternehmen, gerade im Bereich des Customer-Support, vorhanden. Ein fallbasierter Ansatz ermöglicht die effiziente Nutzung dieser Ressourcen. Damit ist die Möglichkeit gegeben, kumuliertes Wissen zentral zu dokumentieren und gezielt zu verteilen, um es so effizient einem breiten Kreis von Nutzern zur Verfügung zu stellen. Daß diese Vorteile im Wettbewerb sich auch in der Praxis realisieren lassen, beweisen die in dieser Arbeit angegebenen fallbasierten Systeme im täglichen Routineeinsatz. Abschließend betrachtet sind fallbasierte Systeme und Anwendungen damit bereits heute das geeignete Mittel um einen schnelleren, zuverlässigen und kostengünstigeren Kundenservice zu realisieren.

6 Danksagung

Alle in dieser Arbeit beschriebenen Anwendungen und Systeme wurden mit Hilfe der CBR2 Technologie der Inference Corporation, USA realisiert. Ich danke der Firma Inference GmbH, allen Kollegen und

unseren Kunden für die mir zur Verfügung gestellten Informationen. Dank gebührt auch allen Mitgliedern des INRECA Konsortium für die sehr gute Zusammenarbeit während meiner Tätigkeit als wissenschaftlicher Mitarbeiter an der Universität Kaiserslautern. Besonders möchte ich an dieser Stelle noch Dr. Brigitte Bartsch-Spörl, Ralf Schulz und Prof. Dr. Ehrenberg danken ohne deren Unterstützung, Aufmunterung und Nachsicht dieser Artikel sicher niemals entstanden wäre.

7 Referenzen

- [AcWa92] *Acorn, T. und Walden, S.*: „SMART: Support Management Automated Reasoning Technology for Compaq Customer Service“. In: *Procs. of the 4th Innovative Applications of AI Conference*, 1992.
- [AlAu95] *Althoff, K.-D., Barletta, R., Manago, M. und Auriol, E.*: „A Review of Industrial Case-Based Reasoning Tools“. Oxford: *AI Intelligence*, 1995.
- [Alle94] *Allen, Bradley, P.*: „Case-Based Reasoning: Business Applications“. In: *Communications of the ACM*, 37(3), March 1994, 40 - 42
- [BaMo92] *Barletta, R; Mott, S.* „Techniques for employing Case-Based Reasoning in Automated Customer-Service Help-Desks“ *Proc. AI for Customer Service and Support*. Morgan Kaufmann 1992
- [Cole92] *Coleman, K.*: „The AI Marketplace in the year 2000“. In: *AI Expert*, Jan. 1992, 35 - 38.
- [Houl95] *Houlder, V.*: „Back to the future“. *Financial Times*, 7.2.1995.
- [KrBa93] *Kriegsman, M und Barletta, R.*: „Building a Case-Based Help-Desk Application. *IEEE Expert*, Dezember 1993
- [Mott93] *Mott, S.*: „CBR: Market, Applications and fit with other technologies“. *Expert Systems with Applications* 6/93 97-104.
- [NgCzLe93] *Ngyuen, T; Czerwinski, M. und Lee S.*: *Compaq QuickSource: Providing the customer with the power of AI. Proc. IAAI-93. AAAI* 1993.
- [Schi91] *Schild, H.G.*: „Fallbasierte Expertensysteme zur Beratungsunterstützung“. In: *ist - Intelligente Software-Technologien*, 2/91, 17 - 23, 1991.
- [Schr93] *Schröder, M.*: *Help-Desk-System*. In: *Wirtschaftsinformatik*, 35 (1993) 1, 280-282, 1993.
- [Schu95] *Schult, T.-J.*: „Maschinen auf dem Weg zur Selbstheilung“, *Die Zeit*, 9.7.1995.
- [Simo91] *Simoudis, E.*: „The application of CBR to help desk domains“. In: *Procs. of the Case-Based Reasoning Workshop*. Morgan Kaufmann, 1991.
- [Simo92] *Simoudis, E.*: „Using Case-Based Retrieval for Customer Technical Support“. *IEEE Expert*, 7/92, 7-12.
- [Vroo91] *Vrooman G.*: „Commercialising CBR Technology“. *AI Review* 4/91, 43-47
- [Wess95] *Wess, S.*: „Fallbasiertes Problemlösen in wissenbasierten Systemen zur Entscheidungsunterstützung und Diagnostik“. *Dissertation*, Universität Kaiserslautern, 1995.



Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

-Bibliothek, Information
und Dokumentation (BID)-
PF 2080
67608 Kaiserslautern
FRG

Telefon (0631) 205-3506
Telefax (0631) 205-3210
e-mail
dfkibib@dfki.uni-kl.de
WWW
<http://www.dfki.uni-sb.de/dfkibib>

Veröffentlichungen des DFKI

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder (so sie als per ftp erhältlich angemerkt sind) per anonymous ftp von ftp.dfki.uni-kl.de (131.246.241.100) im Verzeichnis pub/Publications bezogen werden. Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or (if they are marked as obtainable by ftp) by anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) in the directory pub/Publications.

The reports are distributed free of charge except where otherwise noted.

DFKI Research Reports

1997

RR-97-04

Serge Autexier, Dieter Hutter
Parameterized Abstractions used for Proof-Planning
13 pages

RR-97-03

Dieter Hutter
Using Rippling to Prove the Termination of Algorithms
15 pages

RR-97-02

Stephan Busemann, Thierry Declerck, Abdel Kader Diagne, Luca Dini,
Judith Klein, Sven Schmeier
Natural Language Dialogue Service for Appointment Scheduling Agents
15 pages

RR-97-01

Erica Melis, Claus Sengler
Analogy in Verification of State-Based Specifications: First Results
12 pages

1996

RR-96-06

Claus Sengler
Case Studies of Non-Freely Generated Data Types
200 pages

RR-96-05

Stephan Busemann
Best-First Surface Realization
11 pages

RR-96-04

Christoph G. Jung, Klaus Fischer, Alastair Burt
Multi-Agent Planning
Using an *Abductive*
EVENT CALCULUS
114 pages

RR-96-03

Günter Neumann
Interleaving
Natural Language Parsing and Generation
Through Uniform Processing
51 pages

RR-96-02

E.André, J. Müller, T.Rist:
PPP-Persona: Ein objektorientierter Multimedia-Präsentationsagent
14 Seiten

RR-96-01

Claus Sengler
Induction on Non-Freely Generated Data Types
188 pages

1995

RR-95-20

Hans-Ulrich Krieger

Typed Feature Structures, Definite Equivalences,
Greatest Model Semantics, and Nonmonotonicity
27 pages

RR-95-19

Abdel Kader Diagne, Walter Kasper, Hans-Ulrich Krieger

Distributed Parsing With HPSG Grammar
20 pages

RR-95-18

Hans-Ulrich Krieger, Ulrich Schäfer

Efficient Parameterizable Type Expansion for Typed
Feature Formalisms
19 pages

RR-95-17

Hans-Ulrich Krieger

Classification and Representation of Types in TDL
17 pages

RR-95-16

Martin Müller, Tobias Van Roy

Title not set
0 pages

Note: The author(s) were unable to deliver this document for printing before the end of the year. It will be printed next year.

RR-95-15

Joachim Niehren, Tobias Van Roy

Title not set
0 pages

Note: The author(s) were unable to deliver this document for printing before the end of the year. It will be printed next year.

RR-95-14

Joachim Niehren

Functional Computation as Concurrent Computation
50 pages

RR-95-13

Werner Stephan, Susanne Biundo

Deduction-based Refinement Planning
14 pages

RR-95-12

Walter Hower, Winfried H. Graf

Research in Constraint-Based Layout, Visualization,
CAD, and Related Topics: A Bibliographical Survey
33 pages

RR-95-11

Anne Kilger, Wolfgang Finkler

Incremental Generation for Real-Time Applications
47 pages

RR-95-10

Gert Smolka

The Oz Programming Model
23 pages

RR-95-09

M. Buchheit, F. M. Donini, W. Nutt, A. Schaerf

A Refined Architecture for Terminological Systems:
Terminology = Schema + Views
71 pages

RR-95-08

Michael Mehl, Ralf Scheidhauer, Christian Schulte

An Abstract Machine for Oz
23 pages

RR-95-07

Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt

The Complexity of Concept Languages
57 pages

RR-95-06

Bernd Kiefer, Thomas Fettig

FEGRAMED

An interactive Graphics Editor for Feature Structures
37 pages

RR-95-05

Rolf Backofen, James Rogers, K. Vijay-Shanker

A First-Order Axiomatization of the Theory of Finite
Trees
35 pages

RR-95-04

M. Buchheit, H.-J. Bürckert, B. Hollunder, A. Laux, W. Nutt,

M. Wójcik

Task Acquisition with a Description Logic Reasoner
17 pages

RR-95-03

Stephan Baumann, Michael Malburg, Hans-Guenther Hein, Rainer Hoch,

Thomas Kieninger, Norbert Kuhn

Document Analysis at DFKI

Part 2: Information Extraction

40 pages

RR-95-02

Majdi Ben Hadj Ali, Frank Fein, Frank Hoenes, Thorsten Jaeger,

Achim Weigel

Document Analysis at DFKI

Part 1: Image Analysis and Text Recognition

69 pages

RR-95-01

Klaus Fischer, Jörg P. Müller, Markus Pischel

Cooperative Transportation Scheduling

an application Domain for DAI

31 pages

1994

RR-94-39

Hans-Ulrich Krieger

Typed Feature Formalisms as a Common Basis for Linguistic Specification.

21 pages

RR-94-38

Hans Uszkoreit, Rolf Backofen, Stephan Busemann, Abdel Kader Diagne,

Elizabeth A. Hinkelman, Walter Kasper, Bernd Kiefer, Hans-Ulrich Krieger,

Klaus Netter, Günter Neumann, Stephan Oepen, Stephen P. Spackman.

DISCO—An HPSG-based NLP System and its Application for Appointment Scheduling.

13 pages

RR-94-37

Hans-Ulrich Krieger, Ulrich Schäfer

TDL - A Type Description Language for HPSG, Part 1: Overview.

54 pages

RR-94-36

Manfred Meyer

Issues in Concurrent Knowledge Engineering. Knowledge Base and Knowledge Share Evolution.

17 pages

RR-94-35

Rolf Backofen

A Complete Axiomatization of a Theory with Feature and Arity Constraints

49 pages

RR-94-34

Stephan Busemann, Stephan Oepen, Elizabeth A. Hinkelman,

Günter Neumann, Hans Uszkoreit

COSMA – Multi-Participant NL Interaction for Appointment Scheduling

80 pages

RR-94-33

Franz Baader, Armin Laux

Terminological Logics with Modal Operators

29 pages

RR-94-31

Otto Kühn, Volker Becker, Georg Lohse, Philipp Neumann

Integrated Knowledge Utilization and Evolution for the Conservation of Corporate Know-How

17 pages

RR-94-23

Gert Smolka

The Definition of Kernel Oz

53 pages

RR-94-20

Christian Schulte, Gert Smolka, Jörg Würtz

Encapsulated Search and Constraint Programming in Oz

21 pages

RR-94-19

Rainer Hoch

Using IR Techniques for Text Classification in Document Analysis

16 pages

RR-94-18

Rolf Backofen, Ralf Treinen

How to Win a Game with Features

18 pages

RR-94-17

Georg Struth

Philosophical Logics—A Survey and a Bibliography

58 pages

RR-94-16

Gert Smolka

A Foundation for Higher-order Concurrent Constraint Programming

26 pages

RR-94-15

Winfried H. Graf, Stefan Neurohr

Using Graphical Style and Visibility Constraints for a Meaningful Layout in Visual Programming Interfaces

20 pages

RR-94-14

Harold Boley, Ulrich Buhrmann, Christof Kremer

Towards a Sharable Knowledge Base on Recyclable Plastics

14 pages

RR-94-13

Jana Koehler

Planning from Second Principles—A Logic-based Approach

49 pages

RR-94-12

Hubert Comon, Ralf Treinen

Ordering Constraints on Trees

34 pages

RR-94-11

Knut Hinkelmann

A Consequence Finding Approach for Feature Recognition in CAPP

18 pages

RR-94-10

Knut Hinkelmann, Helge Hintze

Computing Cost Estimates for Proof Strategies

22 pages

RR-94-08*Otto Kühn, Björn Höfling*

Conserving Corporate Knowledge for Crankshaft Design

17 pages

RR-94-07*Harold Boley*

Finite Domains and Exclusions as First-Class Citizens

25 pages

RR-94-06*Dietmar Dengler*

An Adaptive Deductive Planning System

17 pages

RR-94-05*Franz Schmalhofer, J. Stuart Aitken, Lyle E. Bourne jr.*

Beyond the Knowledge Level: Descriptions of Rational Behavior for Sharing and Reuse

81 pages

RR-94-03*Gert Smolka*

A Calculus for Higher-Order Concurrent Constraint Programming with Deep Guards

34 pages

RR-94-02*Elisabeth André, Thomas Rist*

Von Textgeneratoren zu Intellimedia-Präsentationssystemen

22 Seiten

RR-94-01*Elisabeth André, Thomas Rist*

Multimedia Presentations: The Support of Passive and Active Viewing

15 pages

DFKI Technical Memos**1996****TM-96-02***Harold Boley*

Knowledge Bases in the World Wide Web:

A Challenge for Logic Programming

8 pages

TM-96-01*Gerd Kamp, Holger Wache*

CTL — a description Logic with expressive concrete domains

19 pages

1995**TM-95-04***Klaus Schmid*Creative Problem Solving
and

Automated Discovery

— An Analysis of Psychological and AI Research —

152 pages

TM-95-03*Andreas Abecker, Harold Boley, Knut Hinkelmann, Holger Wache,**Franz Schmalhofer*

An Environment for Exploring and Validating Declarative Knowledge

11 pages

TM-95-02*Michael Sintek*FLIP: Functional-plus-Logic Programming
on an Integrated Platform

106 pages

TM-95-01*Martin Buchheit, Rüdiger Klein, Werner Nutt*Constructive Problem Solving: A Model Construction
Approach towards Configuration

34 pages

1994**TM-94-05***Klaus Fischer, Jörg P. Müller, Markus Pischel*

Unifying Control in a Layered Agent Architecture

27 pages

TM-94-04*Cornelia Fischer*

PAntUDE – An Anti-Unification Algorithm for Expressing Refined Generalizations

22 pages

TM-94-03*Victoria Hall*

Uncertainty-Valued Horn Clauses

31 pages

TM-94-02*Rainer Bleisinger, Berthold Kröll*Representation of Non-Convex Time Intervals and
Propagation of Non-Convex Relations

11 pages

TM-94-01*Rainer Bleisinger, Klaus-Peter Gores*

Text Skimming as a Part in Paper Document Understanding

14 pages

DFKI Documents

1997

D-97-01

Thomas Malik

NetGLTool Benutzeranleitung

40 Seiten

1996

D-96-07

Technical Staff

DFKI Jahresbericht 1995

55 Seiten

Note: This document is no longer available in printed form.

D-96-06

Klaus Fischer (Ed.)

Working Notes of the KI'96 Workshop on Agent-Oriented Programming and Distributed Systems

63 pages

D-96-05

Martin Schaaf

Ein Framework zur Erstellung verteilter Anwendungen

94 pages

D-96-04

Franz Baader, Hans-Jürgen Bürckert, Andreas Günter, Werner Nutt (Hrsg.)

Proceedings of the Workshop on Knowledge Representation and Configuration WRKP'96

83 pages

D-96-03

Winfried Tautges

Der DESIGN-ANALYZER - Decision Support im Designprozess

75 Seiten

D-96-01

Klaus Fischer, Darius Schier

Ein Multiagentenansatz zum Lösen von Fleet-Scheduling-Problemen

72 Seiten

1995

D-95-12

F. Baader, M. Buchheit, M. A. Jeusfeld, W. Nutt (Eds.)

Working Notes of the KI'95 Workshop:

KRDB-95 - Reasoning about Structured Objects:

Knowledge Representation Meets Databases

61 pages

D-95-11

Stephan Busemann, Iris Merget

Eine Untersuchung kommerzieller Terminverwaltungssoftware im Hinblick auf die Kopplung mit natürlich-sprachlichen Systemen

32 Seiten

D-95-10

Volker Ehresmann

Integration ressourcen-orientierter Techniken in das wissensbasierte Konfigurierungssystem TOOCON

108 Seiten

D-95-09

Antonio Krüger

PROXIMA: Ein System zur Generierung graphischer Abstraktionen

120 Seiten

D-95-08

Technical Staff

DFKI Jahresbericht 1994

63 Seiten

Note: This document is no longer available in printed form.

D-95-07

Ottmar Lutzy

Morphic - Plus

Ein morphologisches Analyseprogramm für die deutsche Flexionsmorphologie und Komposita-Analyse

74 Seiten

D-95-06

Markus Steffens, Ansgar Bernardi

Integriertes Produktmodell für Behälter aus Faserverbundwerkstoffen

48 Seiten

D-95-05

Georg Schneider

Eine Werkbank zur Erzeugung von 3D-Illustrationen

157 Seiten

D-95-04

Victoria Hall

Integration von Sorten als ausgezeichnete taxonomische Prädikate in eine relational-funktionale Sprache

56 Seiten

D-95-03

Christoph Endres, Lars Klein, Markus Meyer

Implementierung und Erweiterung der Sprache *ALCP*

110 Seiten

D-95-02

Andreas Butz

BETTY

Ein System zur Planung und Generierung informativer Animationssequenzen

95 Seiten

D-95-01

Susanne Biundo, Wolfgang Tank (Hrsg.)
 PuK-95, Beiträge zum 9. Workshop „Planen und Konfigurieren“, Februar 1995
 169 Seiten

Note: This document is available for a nominal charge of 25 DM (or 15 US-\$).

1994**D-94-15**

Stephan Oepen
 German Nominal Syntax in HPSG
 — On Syntactic Categories and Syntagmatic Relations —
 80 pages

D-94-14

Hans-Ulrich Krieger, Ulrich Schäfer
 TDL - A Type Description Language for HPSG, Part 2: User Guide.
 72 pages

D-94-12

Arthur Sehn, Serge Autexier (Hrsg.)
 Proceedings des Studentenprogramms der 18. Deutschen Jahrestagung für Künstliche Intelligenz KI-94
 69 Seiten

D-94-11

F. Baader, M. Buchheit, M. A. Jeusfeld, W. Nutt (Eds.)
 Working Notes of the KI'94 Workshop: KRDB'94 - Reasoning about Structured Objects: Knowledge Representation Meets Databases
 65 pages

Note: This document is no longer available in printed form.

D-94-10

F. Baader, M. Lenzerini, W. Nutt, P. F. Patel-Schneider (Eds.)
 Working Notes of the 1994 International Workshop on Description Logics
 118 pages

Note: This document is available for a nominal charge of 25 DM (or 15 US-\$).

D-94-09

Technical Staff
 DFKI Wissenschaftlich-Technischer Jahresbericht 1993
 145 Seiten

D-94-08

Harald Feibel
 IGLOO 1.0 - Eine grafikunterstützte Beweisentwicklungsumgebung
 58 Seiten

D-94-07

Claudia Wenzel, Rainer Hoch
 Eine Übersicht über Information Retrieval (IR) und NLP-Verfahren zur Klassifikation von Texten
 25 Seiten

Note: This document is no longer available in printed form.

D-94-06

Ulrich Buhrmann
 Erstellung einer deklarativen Wissensbasis über recyclingrelevante Materialien
 117 Seiten

D-94-04

Franz Schmalhofer, Ludger van Elst
 Entwicklung von Expertensystemen: Prototypen, Tiefenmodellierung und kooperative Wissensrevolution
 22 Seiten

D-94-03

Franz Schmalhofer
 Maschinelles Lernen: Eine kognitionswissenschaftliche Betrachtung
 54 Seiten

Note: This document is no longer available in printed form.

D-94-02

Markus Steffens
 Wissenserhebung und Analyse zum Entwicklungsprozeß eines Druckbehälters aus Faserverbundstoff
 90 pages

D-94-01

Josua Boon (Ed.)
 DFKI-Publications: The First Four Years
 1990 - 1993
 75 pages

Workshop held at the
21st Annual German Conference on AI (KI-97)

A. Abecker, S. Decker, K. Hinkelmann, U. Reimer