Integrating Experience-Based Knowledge Management with Sustained Competence Development

Klaus-Dieter Althoff, Dietmar Pfahl
Fraunhofer Institute for Experimental Software Engineering
Sauerwiesen 6, D-67661 Kaiserslautern, Germany
{althoff | pfahl}@iese.fraunhofer.de

Abstract

Integrating experience management and e-learning provides a combination of work-based competence development and problem solving with experience-based generation and usage of learning materials and concurrent and geographically distributed learning. Adequately Packaged experience provides the starting point for the preparation and design of learning materials as well as an idea of how to incorporate learning materials in an experience management system. An integrated experience management and e-learning system has to be much more sensible for its context than a stand-alone subsystem. In this chapter we suggest the so-called "3P-integration" concept, which considers for context modeling not only the processes and projects, but also the involved persons.

1 Introduction and Background

Development and management of content is an area of increasing importance. With data, information, knowledge, and experience, there are different kinds of content as well as different kinds of usage and different research (sub-)fields that deal with these issues. There is a need for establishing an integrative view on available approaches. In this chapter we identify the integration of experience management and e-learning as a core issue and develop suggestions of how to integrate them. Experience, i.e. knowledge that has been acquired in practice, is used to support decision-making in information technology projects, to support individual competence development, and to support the interleaving of experience management and e-learning. Therefore, we address content management from both the experience management and the sustained competence development perspective, and suggest how to integrate these two. For this, we describe the current basics in experience management as well as in e-learning. We show how to extend the current state of the art in experience management through integration with process learning, discourse analysis, experience base maintenance, and a product line approach for experience base development. We show how to enhance existing approaches to collaborative web-based learning by offering support to a broad range of learning processes, their flexible combination, their integration with knowledge management processes, and the possibility to comprehensive reuse of all learning and knowledge resources. We then describe how the integration of e-learning concepts and methods can enhance experience management and we discuss how e-learning can benefit from systematically managed experience. Finally, an outlook for future work is given.

1.1 State-of-the-Art in Experience Management

Experience management defines and develops methods for structuring and dealing with experience of experts on a particular subject, and it is becoming an increasingly important sub-domain of knowledge management. Software engineering is a highly dynamic field in terms of research and knowledge, and it depends heavily upon the experience of experts for the development and advancement of its methods, tools, and techniques. For example, the tendency to define and describe "best practices" or "learned lessons" is quite distinctive in the literature [BT98, AW00, TAN00]. As a consequence, it was the software engineering field where an organization was introduced that was explicitly responsible to deal with experience: The experience factory (EF) [BCR94]. An EF is a logical and/or physical infrastructure for continuous learning from experience and includes an experience base (EB) for the storage and re-use of knowledge. The EF approach was invented in the mid-eighties [Bas85, BR88, BR91, BCC92, BCR94]. As practice shows, it is substantial for the support of organizational learning that the project organization and the learning organization are separated [BCR94].

The initial example for an operating EF was the NASA Software Engineering Laboratory (SEL) [RU89, BCC92, MP90]. In the meantime EF applications were developed in the USA and also in Europe [Hal96, HSW91, Ses96, Rom96, HSW98, TA00, AB+02]. The great amount of successful EF applications gave the ignition to study “Learning Software Organizations“ more intensively regarding the methodology for building up and running an EF. This also includes the definition of related processes, roles, and responsibilities and, last but not least, the technical realization [BR00, CD00, Din00, AB+00]. The most detailed methodology for the build-up of an EF/EB on project knowledge also for the presentation of the according processes is given in [Tau00].

In the areas of cognitive science and artificial intelligence, case-based reasoning (CBR) emerged in the late seventies and beginning eighties as a model for human problem solving and learning [Sch82, SA77]. In artificial intelligence, this led to a focus of knowledge-based systems on experience (experience knowledge, case-specific knowledge) in the late eighties and beginning nineties, mostly in the form of problem-solution cases [BaS87, AK+89, Aha99]. Since several years there has been a strong tendency in the CBR community [Kol93, Alt01] to develop methods for dealing with more complex applications. One example is the use of CBR for knowledge management (KM) [AB+99]. Another one is the integration of CBR with experience factories: Since the mid-nineties CBR is used both on the organizational EF process level as well as the technical EB implementation level [Hen95, AW97, TA97]. Meanwhile this approach establishes itself more and more [BB+99, KM+00, Alt01].

In the eighties and nineties, various approaches in economical and social science as well as in business information systems, which explicitly dealt with knowledge as a resource of increasing importance, merged under the notion of knowledge management [Rom98, Leh00]. In spite of the high number of approaches and their heterogeneity, two main categories can be identified [ADK98, BJA01]. On the one hand, there are process-oriented approaches, which base mainly on communication and collaboration [JH+00], on the other hand, product-oriented approaches, which base on documentation, storage, and reuse of enterprise knowledge [AB+00]. While the former use techniques from computer supported collaborative work and workflow management, the latter build on information technology tools for documenting knowledge: Database systems, repository systems, hypertext systems, document management systems, process modeling systems, knowledge-based systems, case-based reasoning systems, etc. [Goo99, Gro99, AM+00]. From a more general perspective it can be stated that product- and process-oriented approaches are still not integrated. Usually they are used independently from each other, or as alternatives. As a first step forward here, meanwhile a deep – that is the cognitive science foundations considering – integration of the approaches of EF and CBR has been achieved [Tau00, Alt01].

EF is increasingly emerging towards a generic approach for experience management as an organizational structure for reuse of knowledge and especially experience. This includes also applications independent from the software engineering domain. For example, supporting the continuous improvement process in hospitals [AB+99a], the field of help-desk and service support [SW98], and the management of "non-software"-projects [BE+01]. Future trends in the scope of EF include the detailing of all necessary policies, validation, and empirical evaluation [CD00, Tau00, BSL99, BLC01], gaining experience with the technical realization of huge EF's [RDA01], integration with the according business processes [AD+01, DJ01, DA02], and the running of EFs [NAT01].

Meanwhile many papers have been published that are related to the use of CBR in knowledge management. [WAB01] give an overview on intelligent lesson learned systems, which includes CBR approaches. While [War98] describes how CBR can be used for workflow support, [CRS00] focus on the support for business modeling in general. [DJ01] present a first step how business processes and KM/CBR can be integrated. Further approaches on process-oriented knowledge management and CBR can be found in [WG01]. CBR-based knowledge reuse for project management is described in [ANT99], [Tau00], [BN01], and [FI+02]. CBR for supporting knowledge mediation is the topic underlying [GHD99].

1.2 State-of-the-Art in E-Learning

Computer-supported learning and teaching can be traced back to the theory of behaviorism [Tho14, Tho32], and its first practical implementation in the form of so-called "programmed instruction" in the early 1950s [Ski53, Ski54]. Derived from this original work and its extensions, e.g., the inclusion of decisions, and thus the possibility of multiple paths instead of simple "linear programs" [Cro59], Computer Aided Instruction (CAI) became a hype in the 1970s. Important for the success of CAI was the possibility to separate learning methods (practice and examination, tutoring, simulation, etc.) from the subject matter contents. This separation allowed for transferring similar learning methods to various contents. The modular structure of CAI systems consisting of a presentation module, and separate modules for learner response analysis, learning method, and data administration, facilitated the flexible combination of these modules into so-called Computer Based Training (CBT) systems for specific learner groups using a PC. Not advantageous, however, was the strict hierarchical structuring of learning units and the limitations this implies on the workflow of a learner. Most current CBT systems still rely on the old concepts and thus can only be successfully applied when restricted knowledge about subject matter facts and methods shall be trained.

Traditional CBT systems neither have an "understanding" of the contents that are to be delivered to the learners nor do they have information about the varying levels of knowledge and training progress of the learners. The first reaction to these limitations was the attempt in the early 1980s to rely on artificial intelligence approaches. This lead to the concepts of Intelligent Computer Aided Instruction (ICAI) and intelligent tutoring systems [Car70, BBB75, SB82]. The main achievement of ICAI consisted in adding an expert module to the training system which derives correct solutions to given problems and compares them to the answers supplied by the learner. The results of these comparisons are stored in a learner model and analyzed in order to derive individual behavior and knowledge accumulation of the learners. Based on these data, for each learner individually customized learning strategies can be selected. Several ICAI systems were quite successful in their respective application domains (e.g. [BBK82]), but although very modern concepts of knowledge representation (e.g. black boards) and user modeling were applied, in general, the effectiveness of ICAI was rather limited.

Modern concepts for organizing and representing complex knowledge for the purpose of learning and training have their origins in the 1970s when the first hypertext media were developed. The hypertext idea is based in work done during the early 1940s [Bus45] where information units ("cards") were assembled into "knowledge maps". Well-known computer-based systems were NoteCards [Hal88] or Hypercard [Goo87]. A card can be accessed through its name or a link in another card. The hypertext approach strictly distinguishes between structure and content: a hypertext machine administers the cards and their relationships, a database administers the contents of the cards. Unfortunately, the development of "hypermedia networks" turned out to be quite effort consuming and for a long time no adequate authoring systems were available. On the other hand, the usage of established hypertext systems was quite successful and very quickly supported by browsers that provide search and presentation functionality for hypertext information. By adding audio and video functionality hypertext systems were quickly enhanced to hypermedia systems. Today, hypertext and hypermedia form the basis of the World Wide Web (WWW) which provides a common platform for practically all modern e-learning systems.

Since the mid 1990s multimedia PCs have become a standard at the workplace and even in many private households. By its world-wide connection through the Internet and the technical convergence of media and related access systems a huge amount of stored information – but also teachers – can potentially be accessed. This lead to new concepts of university education like the vision of a "virtual campus". So-called "virtual universities" began to offer subject matter contents and services of one or more networked (real) universities over the Internet. In these virtual universities, students enroll electronically, subscribe for courses, access specifically prepared and stored subject matter contents, work on problems and participate in test, communicate with teachers and tutors, and cooperate with peers in order to solve problems.

The trend towards virtualization is not limited to the traditional educational institutions like universities and schools. The modern infrastructures are also used to virtualize trade fairs, congresses and professional workshops. Presentations are transmitted synchronously and asynchronously in the form of videos and multicast streaming media, business television and video-on-demand facilities provide (partly interactive) learning contents that are adapted to the needs of specific learner groups. Workshops and seminars can be supported through electronic meeting support systems that combine in-class with geographically distributed work groups. In private households, learners have the possibility to access an "electronic learning space" full of all sorts of learning media simply by using an Internet browser. Learning sessions can be recorded and repeated whenever necessary, simulations facilitate the comprehension of dynamic problems by interactive variation of parameters. Application sharing allows learners to enter such learning spaces, to conquer them collaboratively with their peers and supervisors – as distributed virtual spaces – and to evolve them constructively.

The possibilities of the new hypermedia (virtual) learning spaces offer new possibilities for learning. Learning can happen at any time and at any place, synchronously and asynchronously, in a self-learning mode or in cooperation and collaboration with peers, in a self-driven (constructive) mode, or guided by tutors and predefined curricula. These new possibilities are of particular importance for the concept of life long learning, where the limit between private and professional competence and skill development becomes fuzzy.

The new technologies that facilitate and support life long learning threaten the traditional distinction between producers of learning contents (i.e., teachers, trainers, tutors) and consumers of learning contents (i.e., learners). In the knowledge-sharing information society every learner can evolve from an information consumer to an information producer, by producing new information offers that others can consume. E-commerce is one of the new business areas where this idea has been most fruitful up to know. But also the new educational systems – either public or private – will be deeply affected by the overlapping of production and consumption of learning contents. The traditional roles of learners and teachers (or trainers, tutors) will eventually disappear [GH01]. For industrial organizations this transfers to the following vision of professional life long learning:

(1) Everybody is a knowledge worker, i.e., everybody consumes and produces knowledge.
(2) The various learning processes of knowledge workers – both self-directed or guided by others – are deeply supported by constantly evolving knowledge networks.
(3) For individual knowledge workers it becomes less important to privately "store" professional subject matter related knowledge. The possibility to access repositories of learning resources (either in the form of persons, or in the form of any other type of content – from simple files to sophisticated, adaptive courseware) makes this obsolete.
(4) The emergence of so-called communities of practice [Wen98] will become crucial because they will guarantee that new knowledge will be transformed into content and that existing knowledge (and associated contents) will be updated according to the needs of professional life.

However, in order to let this vision become reality, some obstacles have to be overcome and some misunderstandings have to be clarified [HMS00]. Many of the promises the Internet (or Intranet) made simply have not (yet) materialized. For example, [DG+99] list the following problems:

(1) The production and maintenance of learning contents has become more effort and time consuming than ever.
(2) The accessibility of learning contents to a wide audience is questionable due to hardware and economic limitations of the end users.
(3) Due to the increasing problem of "broken links" in the web, it is unrealistic to believe that semantically and didactically rich and deep collections of learning contents cannot simply be created through linking with other web servers.
(4) Communication is basically restricted to e-mail, a very limited means for conducting structured discussions (discourses) among several persons on complex problems.

In order to understand the problems of web-based training better, it is helpful to distinguish two main viewpoints: author and learner.

The main interest from the author's point of view is good support for efficient content development. This requests that the following issues be adequately addressed:

- In order to allow the prospective learners to quickly select from a potentially large set of contents, support for detailed and adequate description with content-related metadata is needed [DG+98].
- In order to facilitate adaptation to the specific competence level and previous knowledge of the learner, support for the modularization of contents is essential.
- In order to facilitate personalization to various learning styles of the learners, support for the generation of variants of the contents with adequate instructional strategies must be available.
- In order to facilitate assessment of the learners' knowledge and skills, support for the development of adequate (self-)tests is important.
- In order to support the learners' use of the contents for self-directed learning, a sufficiently large set of references and background literature has to be provided for each topic covered.
- In order to facilitate future improvement of the contents, there must be possibilities to provide feedback from the learners to the author, e.g. in the form of annotations.
- In order to increase productivity of the authors, and to facilitate the application of concepts such as "authoring-on-the-fly" [BO96], complex semantic structures are needed that provide a unique user interface for authors (and learners) and help to reuse and (semi-)automatically integrate (link) existing modules into large courses [MS98].

The main interest from the learner's point of view is good support for efficient problem solution and learning content delivery at any time and at any place. This requests that powerful communication and personalization means be provided. In particular, the following issues need to be adequately addressed [DG+99]:

- Annotations and links: Each learner must have the possibility to annotate learning contents and to make links to other learning contents, either privately or public (at several levels [FM95]). Since public annotations can become subject of annotation by other learners this implies that asynchronous discussions can be initiated from any content on any screen of any learner (who is included in the respective level of publicity).
- Each learning content should possess an associated electronic background library that can be browsed by a learner either via search facilities or via existing links. As mentioned above, learners should have the ability to add new links (either for themselves or for others). The administration of such a background library is quite challenging as it should not only include HTML or XML documents but all kinds of formats, e.g., PDF, Microsoft Office, audio and video clips.
- Besides asynchronous communication resulting from the concept of annotations, an e-learning system should facilitate asynchronous communication by discussion forums. In order to avoid loss of focus (e.g., input submitted to the wrong forum, discussion of the same topic in different forums, input "disappears" in the mass of discussion threads) it is essential provide an adequate level of moderation.
- Each learner must have the opportunity to ask questions about a specific issue. These answers must be responded within an acceptable time period. Ideally, question- and answer dialogues should be implemented in away that after some

time recurring questions on the same (or similar) issues are responded by the e-learning system automatically [Mau98].

- Each learner must have the opportunity to search for a particular piece of information in related background libraries at any time and place.
- A modern e-learning system should provide adequate synchronous communication functionality, either in the form of simple chat functionality or more sophisticated audio/video conferencing systems, possibly with whiteboard functionality.
- A good e-learning system should provide powerful mechanisms for the learners to actively work with the contents, either individually or in groups. This requires an adequate set of personalization and collaboration functionality allowing learners to structure relevant information in their private work space according to their specific needs and preferences, to connects own knowledge with the knowledge of others, and to share parts of the private space with others.

All in all, it has become clear in the recent years that the provision of powerful learning content management systems that address the above listed issues adequately is much more important (and much more difficult) than the provision of sophisticated but monolithic and large CBTs or WBTs on specific topics. In particular, modularization, annotation, information retrieval, and the combination of synchronous and asynchronous communication require the extensive application of knowledge management techniques and methods. In particular, (semi-)automatic retrieval of information for problem solving, and the proactive offering of learning contents for preparation for new tasks can be addressed with innovative techniques and methods stemming from experience management research.

2 Innovations in Experience Management

Several methodologies have been introduced that can be used for developing experience management systems [Tau00, BB+99, BLC01]). The most detailed one is DISER (Design and Implementation of Software Engineering Repositories), an approach for the building-up and operation of experience management systems including a technical infrastructure for its support. DISER consists of the following nine main steps (see also Figure 1):

(1) Developing a vision for the experience management system
(2) Setting goals
(3) Setting subject areas
(4) Defining usage and filling scenarios
(5) Modeling the experience ontology
(6) Implementing the experience management system
(7) Going online with the experience management system
(8) Maintaining the experience management system
(9) Integrating existing and generating new knowledge

DISER usually starts with developing a vision for the experience management system. This means to go through all the following eight steps on a rather abstract level. Such a vision explicates in particular, where the experience transfer can be supported by the experience management system. Based on the vision, concrete goals are defined that are to be achieved. This occurs with consideration of the interests of the stakeholders. With each of these goals appropriate success criteria are associated, which allow a measurement of the progress concerning the goals. By vision and goals in the next step relevant topics, which can contribute to achieving the objectives, are identified and selected. As soon as objectives and relevant topics are known, the acquisition and use of the experiences can be described by scenarios. In the context of the scenarios the need for information is captured more in detail. This allows developing a representation pattern for experiences (ontology), which is usually implemented based on a rapid application development approach. Based on the prototype system the continuous operation of the experience management system is prepared, which includes business process integration, evaluation and maintenance, as well as the integration of available knowledge. DISER includes the creation of a top-down rationale for the implementation (pattern and knowledge acquisition plan). This rationale documents the reference of the components of pattern and knowledge acquisition plan over scenarios and relevant topics to targets and system vision and thus, becomes understandable.

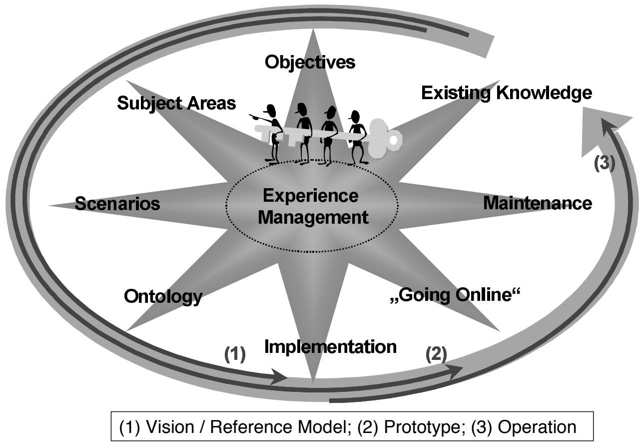


Figure 1. Development and operation of experience management systems

In the following sections we will introduce four different kinds of innovation that improve current state-of-the-practice experience management approaches at least in one aspect. These approaches focus on process learning, discourse analysis, experience base maintenance, and a product line approach for experience bases.

2.1 Integration of Experience Management and Process Learning

The goal of the indiGo project [AB+02] is to develop an integrated solution for process-oriented knowledge management in a software engineering environment. It supports the evaluation and improvement of process models as well as their introduction into an organization.

Therefore, it offers for members of such an organization

- to participate in discourses about process models and
- to access process-related lessons learned, fitting to the current project context.

As shown in Figure 2 various actions are used to support all users of the indiGo system. To achieve this goal, indiGo’s key objective is to create, sustain, and learn about living process models, that is, process models that are

- accepted by the organizations members,
- adapted to organizational changes on demand, and
- continuously enriched with experience from the operating business of the organization.

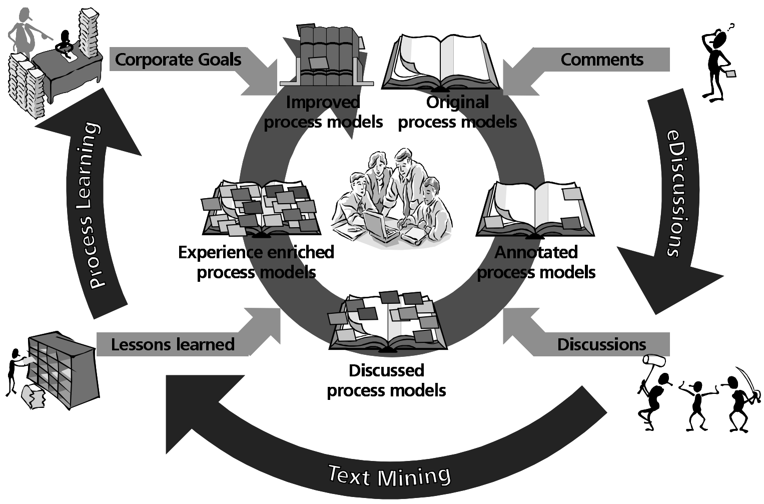


Figure 2. Integrating process learning and experience management

The process models are analyzed and evaluated in goal-oriented, moderated eDiscussions by the process users and experts. An eDiscussion about a process model before and during its implementation supplies feedback to the process author that can be used to continuously adapt and improve it. Information about the discourse has to be collected, analyzed, processed and summarized in eModeration. Guidance of the discourse as well as the management of the participants is needed to achieve the predefined discussion goals.

Collected information is used to cluster and classify discourse contributions to support the various organizational members in indiGo based on real data. Completed discourses are analyzed and summarized to improve related process models and capture previously unknown lessons learned. The primary goals are to support at least the following stakeholders in the indiGo system:

- eModerators to create summaries of long discussions as well as to find relevant information in other discussions,
- process users to detect previously posted answers to similar problems,
- process authors to extract experiences of the process users (lessons learned) from discussions, and
- all organization members to elicit experiences from previous projects.

2.2 Integration of Experience Management and Discourse Analysis

Text mining is concerned with the task of extracting relevant information from natural language text and to search for interesting relationships between the extracted entities. From a linguistic viewpoint natural language exhibits complex structures on different hierarchical levels, which are interconnected to each other [HA96]. These structures, however, are tuned to human cognitive abilities. From the perspective of a computational system, which is adopted here, linguistic information appears to be implicitly encoded in an unstructured way and presents a challenge for automatic data processing.

Text classification is one of the basic techniques in the area of text-mining. It means that text documents are filtered into a set of content-categories. For the task of text classification, there are promising approaches, which stand for different learning paradigms, among them, support vector machines (SVM) are one of the most promising solutions [Joa98]. SVM have been applied to different classification problems - topic detection and author identification [KD+02], multi-class classification [KPL01] - on different linguistic corpora: Reuters newswire, English and German newspapers [LK02], as well as radio-broadcastings [EK+02]. The major problem of applying text classification techniques in application scenarios like the above mentioned process learning and similar experience management situations is the amount of data (Figure 2). The training of a SVM requires some hundred positive and negative examples for each class to be considered. These data must be collected, for instance, in the group discussions. The contributions in a discussion group have to be annotated with respect to the desired classes by the moderator.

An especially challenging task to text mining systems is to map the unstructured natural text to a structured internal representation (basically a set of data objects). Scenarios like the above process learning require to map text documents generated in the group discussions to structured information of project experiences. However, the limited scope of process learning - many roles can only be fulfilled by a finite number of subjects (e.g., the number of an organization’s employees or customers is finite) - makes it possible to invent simplifying solutions to many problems, which are not feasible in the general case.

The context of an utterance consists of all elements in a communicative situation that determine the understanding of an utterance in a systematic way. Context divides up into verbal and non-verbal context [Buß90]. Non-verbal context cannot - or at best to a small extent - be conveyed in written text. Abstracting away from the non-verbal context of the situation which a text (spoken or written) is produced, means, that the lost information has to be substituted by linguistic means in order to avoid misunderstandings resulting from the loss of information. This is why spoken and written language differ. Speaker and hearer are exposed to the same contextual situation, which disambiguates their utterances, whereas writer and reader - in the traditional sense of the word - are not.

Computer-mediated communication adopts an intermediate position in this respect. Writer and reader react on each other's utterances as speaker and listener do. They are in the same communicative situation. But their opportunity to convey non-verbal information is limited as well as the chance to obtain information about the contextual situations of their counterparts.

The context of the communicative situation becomes crucial in settings as presented in Figure 2 when discussions are condensed to project experiences. The communicative situation of the discussion is lost and respective information has to be added to the natural language data. This limits the degree of information compaction of linguistic data. Consequently decontextualization has to be carefully performed in order to not end up in compressed but nevertheless senseless "structured information". How and to what extent information about the communicative situation can be concentrated or discarded is an interesting research objective of the indiGo project [AB+02].

To provide the moderator with information about the problem-orientation of the participants in a discussion in indiGo an "index of speciality of language" is used, which can be calculated on the basis of the agreement of the vocabulary of writer and reader. Self-organizing maps (SOM) [Koh01, Mer97] can give an overview over a set of documents, and thus inform the moderator about similar themes that are discussed in different threads. Standard clustering procedures as well the hierarchical analysis of textual similarities [Meh02] can enhance the presentation of textual data in order to support the moderator in formalizing discussion contributions as reusable experiences or cases.

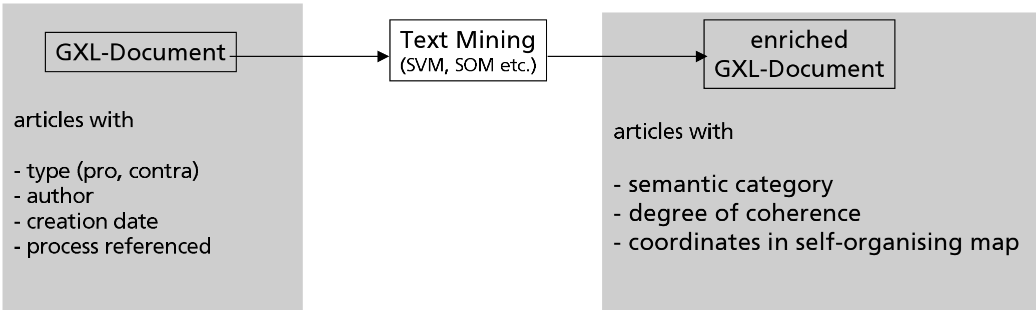


Figure 3. Integrating experience management with discourse analysis

In indiGo discourse structures are evaluated and enriched them with higher level information (Figure 3). For that purpose, the discussions will be exported in GXL, an XML dialect for graph structures. Private annotations remain private and will not be subject to text mining.

2.3 Experience Base Maintenance

After the extension of experience management with process learning has been described, the maintenance of experience base systems (EBS) is presented in the following. First, this section gives the definition of maintenance and how it is handled within an EBS. Then the maintenance-supporting framework is presented.

In the following a wide-scoped definition of maintenance is used: The goal of maintenance is to preserve and/or improve the value of an EBS for the respective organization [NAT01]. The main driving force of maintenance is the EF team. The EF team either performs the maintenance activities themselves, or distributes them among other organization members.

Compared to a dedicated, full-time organizational unit performing maintenance, the distribution of maintenance and the often occurring part-time basis of the EF team demand (a) increased coordination and tracking of the execution of maintenance activities and (b) capture of the knowledge needed during maintenance. The last point (b) also allows delegating parts of the maintenance activities to lower ranking members of an organization. In the long run, the effects of personnel turnover in the EF team are minimized. However, one needs to take the different forms of maintenance knowledge into account: Quality Knowledge, Maintenance Process/Procedure Knowledge, and Maintenance Decision Knowledge.

Quality knowledge describes how the quality of the EBS is measured and the current status of the system with respect to quality as well as the rationale for the definition of quality [Men98]. Quality knowledge deals with quality aspects of the EBS as a whole, that is, the EB's contents and conceptual model as well as retrieval mechanisms, usability of the user interface, etc. An example for content-related quality knowledge is a definition of measures for the utility or value of single cases [NF00]. There are several types of quality knowledge that are related as follows: The measures define what data is collected. The data collection is performed automatically or manually by respective data collection procedures. The collected data is analyzed using predefined models or procedures. The results of the analyses can be used for justifying an EB and as input for decisions about maintenance [NAT01, NF00].

Maintenance process and procedure knowledge defines how the actual maintenance activities are performed. The actual maintenance can be performed as a mix of automatically and manually performed activities. For the automatically performed activities (maintenance procedures), tool support by components of a CBR system or separate tools is required. The remaining activities have to be performed manually (maintenance processes). To improve guidance for the maintainers, descriptions of these processes are provided (e.g., detailed description of the acquisition of new cases through collecting cases, reviewing these cases, and publishing them in the case base, see DISER [Tau00] and INRECA methodology [BB+99] for examples). To combine manual and automatic maintenance, a maintenance process can have automated sub-processes/steps, which use input from or provide input for manually performed steps.

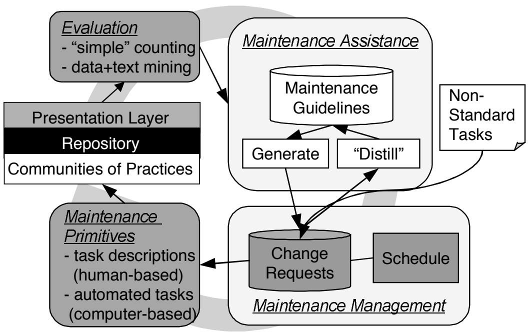


Figure 4. Integrating experience management with maintenance

Maintenance decision knowledge links the quality knowledge with the maintenance process knowledge. It describes under what circumstances maintenance processes/procedures should be executed or checked for execution. Such maintenance knowledge can be described in an informal manner as maintenance policies [LW98], which define when, why, and how maintenance is performed for an EBS. The "why" addresses not only the reason of maintenance but also the expected benefits of the maintenance operation, which should be related to the objectives of the EBS or to the general goal of maintenance (i.e., to preserve and improve the EB's value [NAT01]). Since these objectives are typically very high-level, it is not very meaningful to address the EB objectives directly. Instead, we use a refinement of the objectives: the quality criteria from the evaluation program or the recording methods. The "how" is a combination of maintenance processes and procedures with additional steps as "glue".

Fraunhofer IESE's solution to coordinating experience and capturing the relevant maintenance knowledge is the EMSIG (Evaluation and Maintenance of Software Engineering Repositories) framework as depicted in Figure 4 [NAT01]. This framework includes a method as well as a technical infrastructure and is currently being developed and employed for various EBS's. The evaluation component supports analysis of the content and usage of services, thus, is responsible for the quality and value issues and deals with the "why" of maintenance. The results of these analyses provide the basis and input for making maintenance decisions. The maintenance assistance component supports the decision-making task by exploiting the evaluation in order to propose change requests (i.e., basic maintenance activities to be done). This deals mainly with knowledge issues and the "what" of maintenance ("what" to do for "what" knowledge/experience) and has to consider the "why" (justification from evaluation in the form of expected benefits vs. expected maintenance effort). To support the task of learning about maintenance, typical tasks or patterns of maintenance activities are identified and captured ("distill maintenance guidelines"). These maintenance guidelines can be used for generating change requests automatically. The maintenance management component supports the task of organizing maintenance and, thus, is responsible for handling the change requests in an appropriate order. When a change request is executed, the maintenance primitives component provides the actual methods, technique, and/or tool(s) to perform the basic maintenance activities as demanded by the change request.

2.4 A Product Line for Experience Bases

Methodologies like DISER do not only support the reuse of experience but also of important building blocks like EB-schema patterns [Fel02]. Based on a domain analysis using a goal-oriented evaluation approach for characterizing the requirements for the EBS, a query to an EB of EB-schema patterns provides several typical EB-schema patterns that have to be integrated with a semi-automatic schema integrator (Figure 5). Dependent on the kind of technical infrastructure that is to be supported three different kinds of export functions can be realized by the schema generator: HTML for web-based prototypes, XML for installations based on the orenge™ CBR tool, or SQL-Script for DBMS based installations.

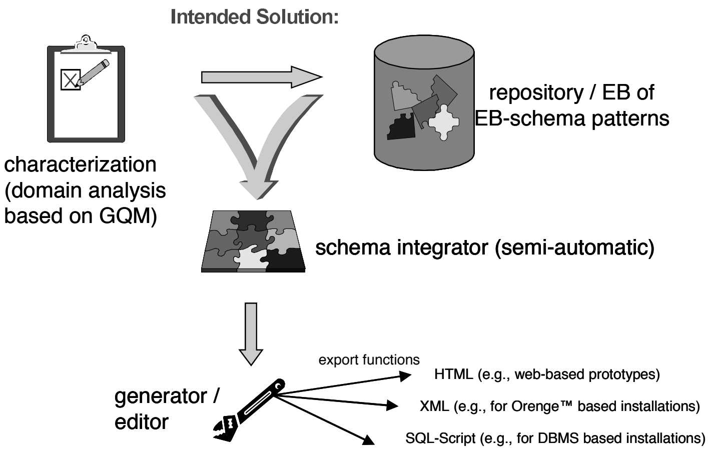


Figure 5. Integrating experience management with a product line approach for EB's

3 Innovations in E-Learning

In the past, the transfer of innovative software engineering know-how was mostly done by University education or classroom-based professional training courses. The new methods of web-based training (WBT) can shorten and considerably enhance this process by making expertise available directly at the workplace. However, many software organizations are still reluctant to introduce web-based training because they have their doubts about the efficiency of these forms of training. They are looking for smooth technical solutions and easily applicable forms of learning and co-operation.

The European research project CORONET (Corporate Software Engineering Knowledge Networks for Improved Training of the Workforce) addresses these issues [PT+02]. It aims to provide a comprehensive methodology and a technical platform for unrestricted professional education and training in the best software engineering practices. The concept of corporate knowledge networks is playing a key role. It is used as a framework for creating and exploiting knowledge assets, sharing knowledge for use and reuse, and learning from others and with others.

3.1 Collaborative Learning with the CORONET System

The CORONET system represents a fundamental paradigm shift from the conventional "online course" model, which tries to capture the best elements of what works so well in classroom learning, to using internet technologies in order to transfer human knowledge in a much more general sense. The system identifies and supports a number of innovative training, collaboration and knowledge management scenarios.

In industrial software organizations, there is a broad range of learning settings[1] that potentially apply to knowledge workers who are in need of evolving their professional knowledge and skills. One extreme of the range is the learning setting ‘participating in a course’, characterized by the organizational need for long-term competence development with predefined individual learning goals, well-structured subject matters, and availability of dedicated trainers/tutors and learning materials. On the other extreme, there is the setting ‘learning within daily work’, characterized by short-lived learning goals of knowledge engineers and the need for spontaneous information search, mainly aimed at solving problems that emerge from daily work.

A major strength of the CORONET system is its ability to cover the whole bandwidth of learning settings, from short-term problem-solving through quick information access to long-term competence development through dedicated web-based training, tutoring and mentoring. The important characteristic of the CORONET system is its focus on collaborative approaches for all relevant learning settings. In particular, the CORONET system promotes and supports the development of sustained interpersonal relationships in combination with comprehensive functionality for accessing, annotating, and extending materials (from others and for others). In this way, it helps to establish learning networks in which people of equal and different competence levels practice both individual and group learning, experience-based learning, learning with multiple activities and resources, and knowledge sharing.

The CORONET system is a combination of two components: the CORONET-Train methodology for collaborative learning at the workplace and the CORONET platform (WBT-Master) supporting the methodology. Each component offers innovations to the current state-of-the-art of web-based learning.

---

[1] The term ‘learning setting’ in this paper is defined as the implementation of a didactical design consisting of topic-related content, instructional strategies, learning activities, and tool support.

3.2 The E-Learning Methodology CORONET-Train

Collaborative learning is generally characterized by the goal to augment and optimize the shared knowledge of a group or community, but it is also meant to support the individual knowledge development. This is reached by co-operatively working on a project, negotiating on the learning goals and problem definitions and collectively constructing knowledge in the group. To realize collaborative learning processes a learning community has to be established. Learning communities are characterized by (1) an individual and collective learning process, (2) experienced-based learning, (3) learning with multiple activities and resources in the group, (4) the sharing of knowledge. A learning culture is practiced which is focused on the active participation of every group member. In this context the group members are not learning all at the same time the same contents; rather, they are developing their knowledge and skills according to their own needs and interests, but in a way that the whole group can profit from that afterwards. So, different learning interests and the development of different kinds of expertise are accepted and actively supported.

Focusing on web-based collaborative learning, the design of the methodology CORONET-Train was guided by the following general principles for collaborative learning environments [Res89, RRM99]:

(1) Learning tasks are based on real-life tasks or authentic situations concerning the development and maintenance of software.
(2) Learning tasks require and motivate the co-operation or collaboration (co-construction and exchange of knowledge) of learners in a group.
(3) Learning contents must be applied in different situations and from different perspectives so that the software-engineers learn to use the acquired knowledge and skills flexibly.
(4) Instructional support is provided; this is required because success of self-organized and social learning processes in net-based learning environments depends very much on the provision of a sufficient information base for the learning process and the adaptive support of the group processes.
(5) The learning environment supports collaborative learning adapted to different competence levels of software engineers (novice, practitioner, expert).
(6) The learning environment supports the access to relevant knowledge repositories in the organization.
(7) The learning environment supports the access and communication to relevant subject matter experts and peer learners, which are engaged in a similar learning topic or process.
(8) The collaborative learning process realizes central features of a “learning community”: it promotes the development of both, individual and socially shared knowledge; it supports the software-engineers to learn from their (good and bad) experiences and mistakes; it supports and instructs the learning group how to reflect their individual and collective experiences, identify their learning needs, and continually evaluate their knowledge and experience development (promotion of meta-cognitive group processes); it initiates the sharing and negotiation of knowledge (development of a positive learning culture; provision of software tools for the exchange and negotiation of knowledge); it takes care that the group members are structurally dependent on each other and that they remain open-minded to external knowledge resources; it takes care that the group members respect each other, even when they have opposite opinions and discuss them

controversial; and it strives to support the development of a group-oriented identity.

(9) The learning environment provides specific methods to support collaborative net-interactions, i.e., it introduces roles or guideline questions for a collective task collaboration, or define specific interaction rules for the virtual co-operation, it uses graphic representation tools to create a collectively visible problem space, it implements a network moderator to help the group regulate their communication processes, to co-ordinate the group activities and to initialize and maintain their negotiation processes.

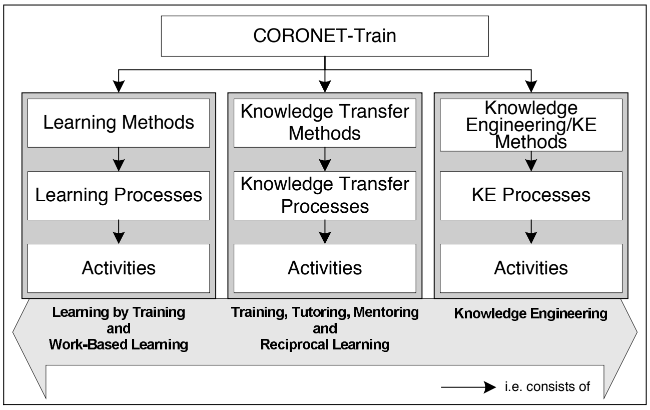


Figure 6. Overview of CORONET-Train

In accordance with these general principles, the innovative character of CORONET-Train can be summarized as follows [PA+01]:

- CORONET-Train offers a long-term perspective to any learning and training activity by providing a career-path orientation to subject matter expertise (systematic development of competencies) [OW00, WO+00].
- CORONET-Train is focused on web-based collaboration between learners on different competence levels, and on usage of a corporate knowledge network, which is transformed into a learning network [HH+95].
- CORONET-Train promotes the integration of web-based training with collaborative learning at the workplace (work-based learning [Rae00]). This can happen in different ways, e.g.:
  - Learners participate in “real life” work processes and take over certain real tasks or parts of tasks while a tutor or mentor guides them [CBN89, MO91].
  - Learning contents emerge from the work situation; starting out from the real-world problem, authentic problem constellations for training sessions can be generated.
- CORONET-Train integrates collaborative learning with knowledge management by linking training processes with knowledge creation, knowledge structuring and knowledge dissemination processes [DN98].
- CORONET-Train meets learning needs as they occur in the workplace (learning on demand) and supports modern concepts of instructional design, such as the engagement theory [KS99].
- CORONET-Train introduces the idea of reciprocal learning into software organizations, i.e. every knowledge worker can receive or offer training, depending on the subject matter and the competence dimension.
- CORONET-Train is adaptive to the current situation of a software organization which may have training needs across all organizational levels – from operational staff to senior management.

CORONET-Train offers three classes of methods, each method consisting of a set of processes and activities (see Figure 6):

(1) Learning methods: Five methods (Case-Based Learning, Theme-Based Learning, Web-Based Training, Web-Based Tutoring, and Knowledge Sharing) define learning processes[2] and activities that are adequately tailored to specific learning situations and learning needs of software engineers. The description of processes and activities is made from the perspective of those who wish to acquire new knowledge and skills related to a specific subject matter.
(2) Knowledge transfer methods: Three methods (Training, Tutoring, Mentoring) define processes and activities that subject matter experts can apply in order to disseminate their know-how and help software engineers satisfy their learning needs.
(3) Knowledge engineering methods: Four methods (Authoring, Structuring, Administration, Management) define processes and activities that are needed to develop, structure, and maintain learning resources, to set-up and maintain the software infrastructure, to administer the users of the infrastructure, and to introduce and manage the learning environment.

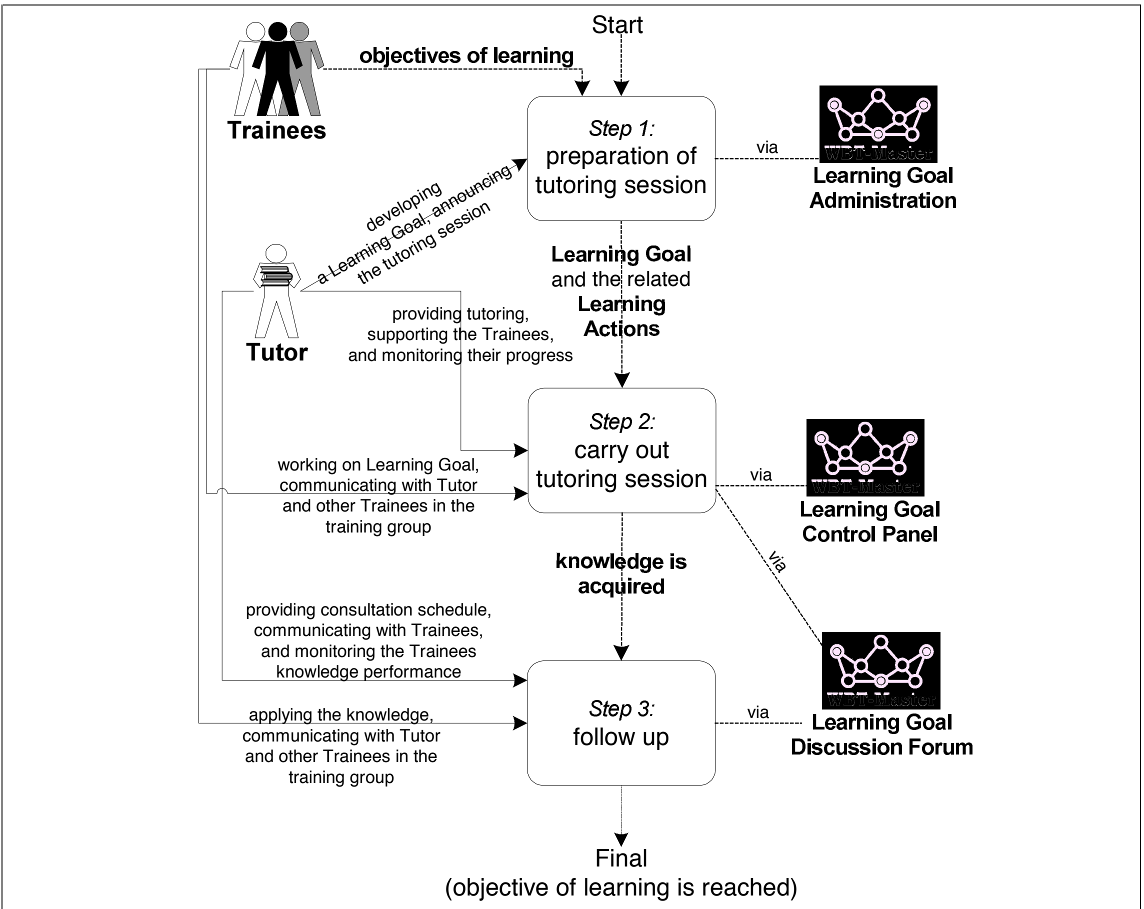


Figure 7. Example of a situational diagram (learning scenario: Web-Based Tutoring)

[2] In CORONET-Train, the term 'learning process' is used to define a sequence of learning activities. This differs from the usage of the term 'learning process' in educational science, where it refers to the internal processing of information by a learner.

A learning scenario is an implementation of one or more CORONET-Train methods or parts of them (i.e. processes and their activities). In a learning scenario, processes and activities are adapted to a particular learning situation and supporting software infrastructure. The purpose of learning scenarios is to organize and maintain relationships among individuals involved in a learning situation by defining the sequence of tasks and their associated actions, which have to be performed in order to reach a learning objective. Guidebooks with situational diagrams (an example is shown in Figure 7) help learners perform a specific learning scenario [AP+02].

During the CORONET project, a number of learning scenarios have been developed in order to show how CORONET-Train can be applied to frequently occurring learning situations in software organizations, e.g. web-based mentoring, web-based knowledge mining, web-based collaborative problem-solving, and web-based virtual classroom. Once managers responsible for human resource or competence development are familiar with CORONET-Train, they can develop new learning scenarios to adapt CORONET-Train to different learning situations.

3.3 The E-Learning Platform WBT-Master

The CORONET infrastructure WBT Master provides the users with the adequate information and communication technology based functionality needed to perform selected learning scenarios including:

- E-learning functionality: Learning Courses; Learning Goals; Structured Discussion Forums; Virtual Classrooms; Brainstorming Sessions; Mentoring Sessions; Progress tracking, testing and certification.
- Knowledge management functionality: Knowledge Cards; Knowledge Domains; Personal Desktop; Content Taxonomies.

Conventional Web-Based Training (WBT) systems utilize HTML documents as learning resources. Ordinary Internet hyperlinks (references) are used to create such navigable data structures as courses, chapters, books, etc. Typically, various WBT tools such as annotations, email, discussion forums, personal bookmarks are used to add additional value to the basic documents published on the WWW. WBT-Master considerably extends this state-of-the-practice technology [HM+01]:

- In addition to existing data structures based on hypermedia links, it introduces such new innovative composite learning resources as reusable Learning Units, Learning Goals, Knowledge Cards, Mentoring Sessions, Knowledge Domains and more.
- WBT-Master enables synchronous and asynchronous communication and collaboration between distributed teams and team members. This includes discussion forums, brain storming sessions, chats, annotation facilities etc.
- In addition to especially prepared training materials, anything that is part of the stored enterprise knowledge, such as technical documents, presentations, or the personal experiences of employees can be used as learning resources via the internet or intranet. The system essentially supports the involvement of human subject matter experts as learning resources.
- Since all information services operate with unified data structures, results of any collaboration (discussion sessions, brainstorming sessions, annotations, question-answer dialogues, etc.) can be seen as new training material and can be reused by others.

3.3.1 Collaborative Learning with WBT-Master

By using WBT-Master, knowledge workers (learners) in a software organization can perform a broad range of collaborative learning scenarios as described by the methodology CORONET-Train:

- Web-Based Training: An experienced knowledge worker acting as a trainer conducts training sessions on a regular basis. In collaboration with a courseware author the trainer develops a Learning Course related to a specific subject matter and makes an announcement on the WBT-Master server [GPR02]. Potential learners may access the announcement board and subscribe to a particular training session.
- Web-Based Tutoring: This scenario is similar to the Web-Based Training scenario. The principal difference is that after having analyzed the subject matter, the tutor or trainer does not trigger the development of courseware, but instead collects a number of heterogeneous documents (textual files, WinWord files, PowerPoint presentations, simulations, etc.) which can be used for the training session. The tutor uploads the documents to the WBT-Master server and defines a special training schedule recommending which document should be accessed at each particular stage of the training session and what actions are expected from a learner working with the document.
- Web-Based Mentoring: Starting point of this scenario is that a learner (or a learning group) needs to solve a particular problem. The learner has a stable partnership with an experienced knowledge worker who can act as a mentor. The mentor is supposed to help the learner acquire new knowledge in the related subject matter area. The mentor can access the server to initiate a special one-to-many synchronous communicational session with interested learners. This kind of communication is called a "mentoring session" [HMS01a]. The mentor explains the problem solution by guiding the mentoring session. The mentor may select a document which is automatically visualized on the learner's screen (share a document), provide an explanation (text, voice and/or drawings) attached to the document, or request the learner to perform an action which may be monitored from the mentor's screen. Similarly, the learner may provide comments (text, voice) to the shared document or ask questions (text, voice) in the context of the shared document.
- Web-Based Knowledge Mining: In this scenario, a knowledge worker needs learning material on a particular subject matter to acquire additional knowledge. He/she is aware of a knowledge network supported by the WBT-Master server, containing relevant information about documents or subject matter experts. The information is structured by means of Knowledge Cards which can be used by the knowledge worker to find relevant learning resources, work through relevant materials and communicate with experts and with others working on similar materials.
- Web-Based Knowledge Delivery: In this scenario, knowledge workers need to acquire knowledge on a particular subject matter in a long-term perspective. They are aware about a WBT-Master server that contains relevant information and is periodically updated by the subject matter experts. The knowledge workers access the server to configure their personal profiles in such a way that relevant learning resources are automatically delivered to their personal desktops and they are automatically notified about new learning resources. Communication with subject matter experts and peers working on similar learning resources is possible via the desktop.

- Web-Based Collaborative Problem Solving: In this scenario, a number of knowledge workers need to solve a particular problem. They are aware that the WBT-Master server can facilitate a so-called “brainstorming session”. A moderator is selected to initiate and organize the brainstorming session to elaborate a solution to the problem. Other knowledge workers that join later can catch up the problem solving process asynchronously from the recorded session.
- Web-Based Gathering and Integration of Personal Knowledge: In this scenario, an experienced knowledge worker needs to gather know-how and experience from a number of experts on a particular topic, and would like to present this knowledge in the form of a training resource. Typical examples of this application are collaborative document writing or co-operative courseware authoring [GAP02, HMS02]. He/she selects a moderator and they discuss the topic via a structured discussion forum, and work co-operatively to develop relevant documents through shared folders. The subject matter experts write contributions, attach documents from their local drives or provide references to relevant documents available from the Internet. Finally, the structured discussion (or selected components of the discussion) is converted into a homogeneous HTML document or a new Learning Unit.
- Web-Based Virtual Classroom: In this scenario, a Virtual Classroom is used for highly interactive and intense training courses in which a trainer/tutor wants to retain the human element of interaction while relying upon an IT infrastructure. The Virtual Classroom can be seen as a working place for the trainers/tutors in which they prepare training sessions for a group of trainees. For each training session, a trainer/tutor creates a new classroom library by selecting the necessary learning resources and moves them to the trainees’ computers. Trainers/tutors can also describe the learning paths to be followed by the trainees in setting up a classroom curriculum.

3.3.2 Knowledge Management with WBT-Master

The corporate memory, or experience base, of a software organization may be seen as a combination of resources and operations applicable to such resources. The operations allow users to access and create new resources, or to add an additional value to existing resources. The WBT-Master platform works with the corporate memory by offering the possibility to access and process huge collections of documents, portals (i.e. references to information resources available from the internet), on-the-fly material (i.e. annotations to documents, contributions to discussions, question-answer dialogues, etc.) and personal knowledge of individuals in the organization. The resources of the corporate memory can be seen as basic Learning Resources. Basic Learning Resources may be organized into composite structures that serve to accomplish a particular learning or problem-solving task. Learning Resources combined into a composite structure may be seen as a new Learning Resource. In other words, Learning Resources may always be reused by a member-wise inclusion of these resources into other ones.

WBT-Master Content Structuring Paradigms

WBT-Master supports a hierarchy of content structuring paradigms, and is based on sound principles of multi-level data modeling [HMS99]. The overall content structuring model is defined as three levels of content abstraction:

- Basic Elements or indivisible chunks of multimedia information (i.e., documents, portals, questionnaires): Basic Elements can be seen as actual pieces of information presented in internationally recognized data encoding formats. For example, basic elements can be HTML documents, WinWord or PDF files, Power Point presentations, plain GIF images, etc. No inter-document relationship is supposed to be defined on this level.
- Logical Composites (i.e., Learning Units, Learning Goals, discussion threads): Logical Composites combine a collection of basic elements and other logical composites into a navigable structure. It can be primitively seen as a collection of hypermedia links that are separated from a document content and combined as a new entity called a logical composite. It should be noted that such composites deal with inter-document relationships and cannot affect a document content.
- Semantic Composites (i.e., Knowledge Cards, Knowledge Domains, content taxonomies): Semantic Composites provide a semantic structuring of server content as such. For example, any basic elements and logical composites can be attached to special Knowledge Cards, in this case, materials get a special meaning defined by the card and can be inferred as so-called "best-match" training resources for users interested in this or another related topic [HMS01c]. Similarly, basic elements and logical composites may be put into a number of content taxonomy folders and thus be accessed by browsing of the semantic net [HMS01d].

Document Repositories

Alternatively to logical and semantic content structures supported by WBT-Master all basic elements and logical composites are stored as files into so-called physical repositories on the server. A particular repository is created by the server file management system as a directory possibly containing files and other subdirectories. While WBT-Master logical and semantic composites may be accessed only by means of the system tools, WBT-Master repository may be accessed as ordinary directories by means of content management tools (say, for example, by means of a file browser locally or by means of an FTP client remotely).

WBT-Master supports five types of repositories: shared files (public level), group resources (restricted to defined groups of knowledge workers), on-the-fly material (group level), personal files, and personal bookmarks.

Knowledge Cards

WBT-Master offers a simple but practical way of accessing preferred learning resources (e.g., Learning Units, Learning Courses, forums, Learning Goals) based on so-called Knowledge Cards. A Knowledge Card is the description of particular concept (i.e. semantic entity). For example, a semantic entity "Code Inspection" may be seen as a knowledge card. Knowledge cards may be combined into a semantic network using just one type of relationship: "is a part of" (the inverse relationship may be called "consists of"). For example, the knowledge card "Perspective Based reading (PBR)" may be related as "is a part of" to the knowledge card "Code Inspection".

The semantic relationships essentially define a graph structure (as opposed to just a hierarchy). For example, the same knowledge card "PBR" may be defined as a part of "Quality Assurance", "Verification Techniques", etc. Moreover, there may be

Knowledge Cards defining areas of personal interest, say "Expertise of Mr./Mrs. XY" which may also refer to the previously mentioned card "PBR", etc.

To be more concrete, each Knowledge Card may provide access to a number of associated Learning Resources. For example, a Learning Course "PBR Techniques" may be associated with the Knowledge Card "PBR", other Learning Units, Learning Goals, Discussion Forums, Documents, etc. may be associated with the same Knowledge Card. Moreover, WBT-Master considers users to be Learning Resources (so called "Peer Helpers") as well, thus, Peer Helpers may be also associated with a Knowledge Card.

Whenever a content provider contributes to the server with new material, he/she is supposed to associate it with one or more Knowledge Cards or create a new Knowledge Card and place it into a proper position within the semantic network. This could also be done by a specially designated role, i.e. the Knowledge Engineer.

The semantic net defined by the set of Knowledge Cards offers the possibility to infer Learning Resources using semantic relationships. Whenever a user accesses a Knowledge Card, the system infers all Learning Resources that are associated with this particular Knowledge Card and with Knowledge Cards related to it. The advantage is that knowledge workers are not supposed to browse through countless learning resources but simply can browse the semantic net consisting of previously defined Knowledge Cards.

Knowledge Domains

The main purpose of Knowledge Domains is to create and maintain well-structured repositories. The Knowledge Domain concept allows for imposing different types of data structures on top of existing collections of Learning Resources, or – seen from another point of view – for reusing Learning Resources in different contexts [HMS01b].

A Knowledge Domain can be defined as a set of documents belonging to a number of predefined semantic categories where each semantic category is linked to a set of Learning Resources that are instances of the category. The definition of a semantic category includes the definition of a number of attributes, which are properties of instances of the semantic category. An attribute is a standard key-value pair. A value of an attribute is defined to be of a specified type, i.e., a value may be a string, a number or a selection from a list of possible values. For example, the category "Author" may have two associated attributes: Name (string) and E-Mail address (string). Similarly, the category "Module" may have just one associated attribute - programming language (selection from a list of languages).

The Knowledge Domain schema defines common properties of all the category instances. Any resource may be inserted (stored) into a particular knowledge domain as an instance of a predefined category. For example, if a new instance of the category "Module" is created, the system automatically requests to select a programming language (attribute predefined for the category), and to provide references to the module author and a particular project (relationships predefined for the category).

4 Perspectives of Integrating Experience Management with E-Learning

An integrated experience management and e-learning system has to be much more sensible for its context than a stand-alone subsystem. In the following we suggest the so-called "3P" integration concept, which considers for context modeling not only the processes and projects [Tau00, AD+01], but also the respective person. First, we describe how experience management can be enhanced by integrating e-learning concepts and methods. Second, we show how e-learning can benefit from systematically managed experience.

4.1 Enhancing Experience Management with E-Learning

A typical experience package does not include deep knowledge, because it is often reused during project work or other kinds of services, and reading it must not take too much time. The provided experience package depends on the respective project and process context based on which the most similar one is provided. However, not every knowledge worker has the same qualifications, interests, and tasks, that is, there exists also a third context dimension, the "Person" dimension. Based on this insight we developed the "3P-Integration[3]" concept for Experience Management and e-learning. Extending the characterizations of experience packages by adding person-related attributes enables the system to provide the knowledge workers with information tailored to their individual needs. In addition, by offering "richer" artifacts than just simple lessons learned, i.e., specifically tailored learning resources, proactive competence development can be supported and thus the risk of mistakes in applying processes, methods, or tools during project performance can be decreased. In Figure 8 the integrated approach is presented in some detail with a focus on the usual project life cycle (initialization, performance, wrap-up).

Project Initialization

At project start, knowledge workers may have two types of needs: information needs and learning needs.

In order to satisfy their information needs, involved roles (i.e., Project Manager – PM) retrieve all relevant information, such as:

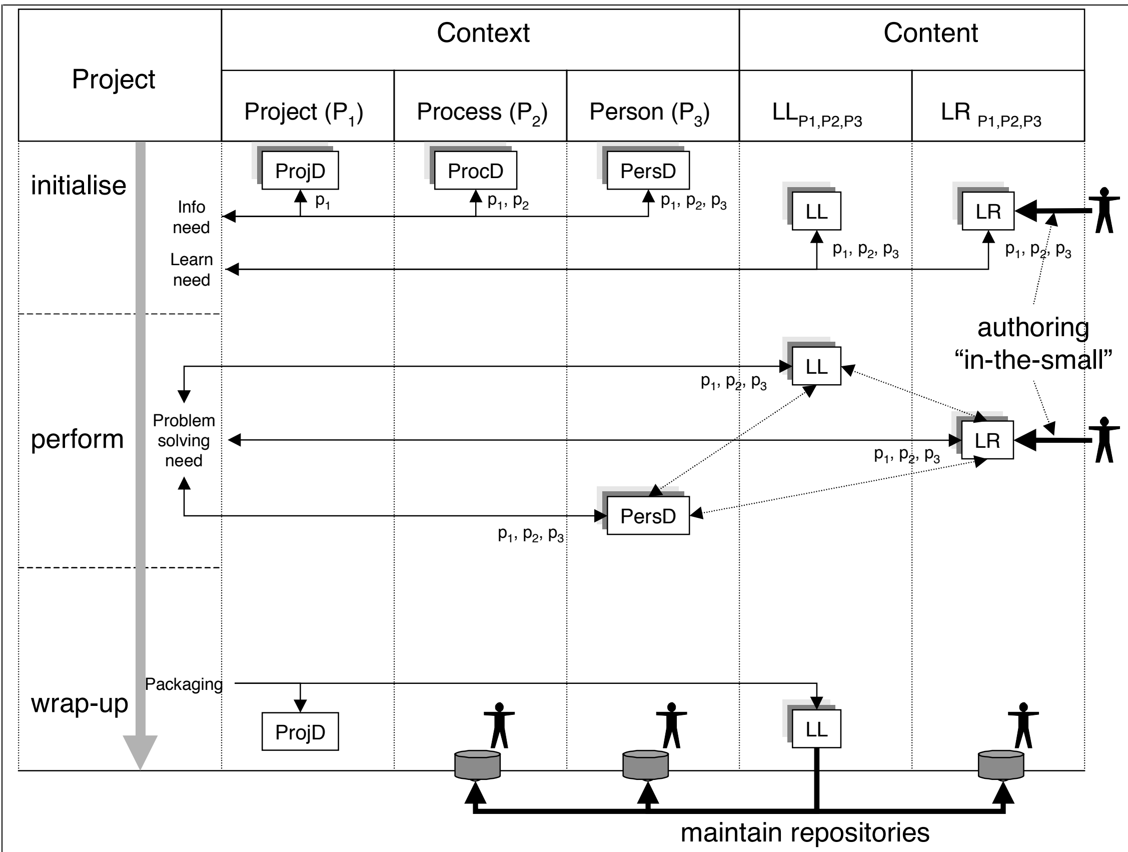
(1) Descriptions of similar projects: similarity is based on current project (attribute P_1). [Example: public funded EU project with research partners X, Y, Z, and industry partners A, B, C]
(2) Descriptions of related processes: relationship is based on current project (attribute P_1) and process (attribute P_2). [Example: process for detailed planning which might vary according to project type]
(3) Descriptions ("business cards") of related persons (e.g., subject matter experts): relationship is based on current project (attribute P_1), process (attribute P_2), and person (attribute P_3). [Example: all experts for detailed planning with experience in similar projects; since PM is an expert in CPM and PERT, no expertise is needed on that, but since PM has no experience with cost estimation, expertise on that is needed]

In order to satisfy their learning needs, involved roles (i.e., PM) may:

(1) Retrieve all related LLs (e.g., offering guidelines, tips and tricks, etc.).

[3] "3P" stands for "Project, Process, Person".

(2) Retrieve all related LRs (e.g., offering little web-based training courses and explanations with an adequate instructional design). It should be noted that the presentation of the LR depends on the personal characteristics of the PM, e.g. learning style and competence level. If adequate LRs are not available as-is, they can be generated either (semi-)automatic [Cau00] or by "authoring in-the-small" [GAP02].



Legend:
P1 = Project Dimension, P2 = Process Dimension, P3 = Person Dimension
Context Repository:
ProjD = Project Descriptions,
ProcD = Project Descriptions (models / e.g. in the form of an EPG),
PersD = Person Description ("business card" with contact details, experience, role, status, expertise, etc.)
Content Repository: LL = Lesson Learned, LR = Learning Resource (i.e., WBT courseware module or fragment)

Figure 8. Enhanced experience management approach

Project Performance

While the project is running, information and learning needs can be satisfied as they occur in the same way as during project initialization. When problems occur that cannot be resolved by reading the process description, there are three possibilities to get help:

(1) Retrieval of a solution to the same or similar problems that occurred in the past (LL).
(2) Retrieval of related learning materials (LR), suited to the context and the personal learning style.
(3) Retrieval of contact information to relevant experts (PersD).

The prioritization of the retrieved information is based on a set of rules, e.g., generally, experts should not be bothered with questions if the problem can be solved by consulting a LL or a by refreshing the knowledge by self-learning with a LR.

Project Wrap-Up

At the end of the project, a new project description (ProjD) and a set of lessons learnt (LLs) is generated. In a simple setting, the LLs are derived from wrap-up interviews. In a more advanced setting, they can be (partly) derived from annotations on project, process, person descriptions, LLs, and LRs.

Based on the analysis of the LLs, existing ProDs, PersDs, and LRs are updated, and/or new ProcDs, PersDs, and LRs are created.

4.2 Enhancing E-Learning with Experience Management

To base competence development of software engineers exclusively on a) reading the related process documentation and b) learning from experience of previous (similar) experiences (e.g. packaged into tips and tricks) is not always efficient. There are two reasons for that:

(1) Theoretical process, method, or tool related knowledge is very complex and needs some sort of training in order to provide a minimal set of related skills upfront, that is, before the first application of the process.
(2) Theoretical process, method, or tool related knowledge is very complex and too boring to be acquired by simple reading of the process description; in order to avoid mistakes during execution (e.g., due to omission) some sort of initial training can be beneficial.

Thus it is necessary to provide adequate LRs at the right time and with little search effort. In order to do, so the following research problems have to be tackled:

(1) Which parts of a process description (and associated methods and tools) need to be trained/taught before application?
(2) How has the training material to be prepared in order to be most effective?
(3) How and when do training materials have to be delivered to be most effective?
(4) How can learning materials (i.e., their content, their presentation, and their delivery) be adequately adapted to the personal profiles (previous knowledge, preferred learning style, etc.) of knowledge workers?

We believe that most of these questions can be answered by exploiting experience. Once the problems have been resolved, new training materials (LRs) can be generated by (re-)using authentic project experience (as captured in LLs), for instance, real application examples, typical mistakes, tips and tricks, and enriching them with didactically relevant enhancements (e.g., explanations, exercises, tests). As soon as an initial set of learning materials and delivery policies have been defined, the learning and training materials (LRs) associated with a particular process description can be treated in the same way as the process descriptions itself, that is, they can undergo the same experience-based improvement paradigm (cf. Figure 9).

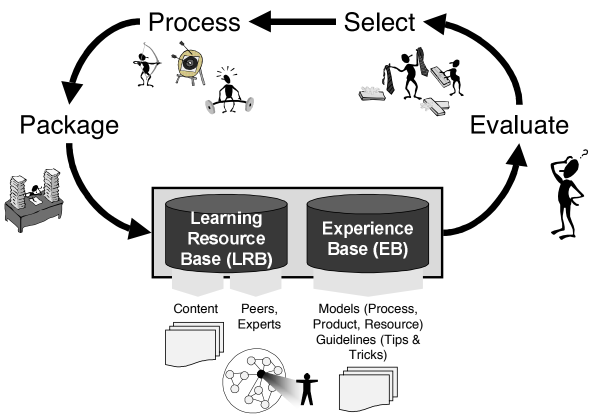


Figure 9. Experience-based identification and improvement of LRs and LLs

Eventually, experience-based e-learning will switch from reactive to proactive, i.e., instead of requesting a knowledge worker to retrieve adequate LRs as a learning need occurs (pull strategy; process-oriented learning), the e-learning environment automatically offers adequate LRs as soon as it detects a potential need during the performance of a particular development process by the knowledge worker (push strategy; process-integrated learning). The automatic offering of adequate LRs as the need occurs can be called "eCoaching".

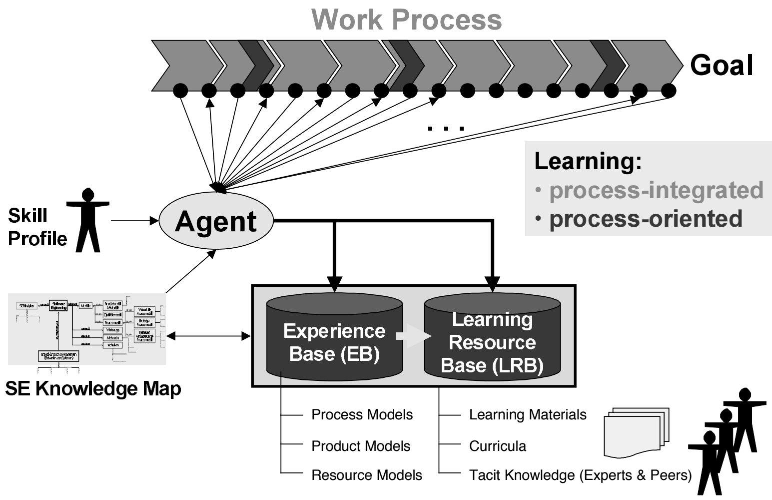


Figure 10. Active guidance of knowledge workers through eCoaching

5 Conclusion and Future Work

E-learning may no longer be seen as an isolated process but as an aspect of knowledge transfer, and thus as an element of knowledge management. The traditional perception of e-learning as a discipline to produce complex courseware that is delivered over the web emerged from the practice to use CBT or WBT as a complement or even as a substitute for traditional in-class training. Today, and particularly in the context of industrial organizations, it has become clear that e-learning must be seen as a method to retrieve needed information (to enhance the knowledge) and needed training (to enhance the skills) at any place and at any time. This may range from a simple search for a particular piece of information to a discussion with peers on a specific topic, the contacting of an expert who provides answers and guidance, the retrieval of personalized learning contents for self-directed training (which may be assembled on-the-fly) or the participation in scheduled web-based training sessions with tutor support. All this can only happen if e-learning takes advantage of many of the achievements that knowledge management – and in particular experience management – has developed.

Core processes of knowledge management are knowledge creation, knowledge structuring, and knowledge dissemination (cf. Figure 11). Knowledge dissemination and knowledge creation also occur in e-learning. Typically, however, in e-learning, the links between knowledge dissemination and knowledge creation are rather weak from an organizational perspective. On the one hand, new (individual) knowledge generated through learning is not (sufficiently) made explicit, e.g. in the form of new or enhanced learning materials. On the other hand, there is a lack of adequate structuring mechanisms that would allow for easy retrieval and reuse of existing knowledge (generated by others).

As we have discussed in Section 3, the recently developed CORONET system bridges this gap by providing flexible annotation and recording mechanisms for discussion threads, and by applying advanced knowledge structuring methods for all sorts of learning resources. Application of these mechanisms facilitates on-demand creation, evaluation and evolution of learning materials based on effective retrieval, feedback, and reuse support. In addition, knowledge creation, knowledge structuring and knowledge dissemination can benefit from learning networks and their associated collaborative learning processes by systematically capitalizing upon individual expertise gained during collaborative learning.

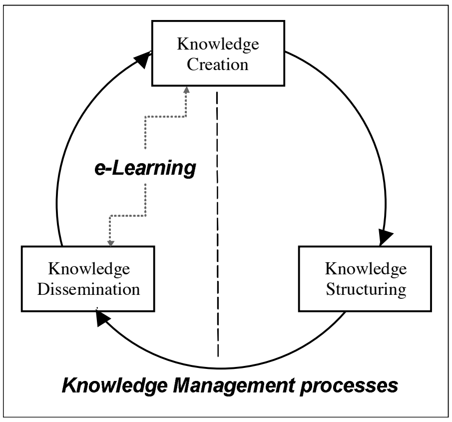


Figure 11. Integration of e-learning with knowledge management in CORONET

By combining innovative e-learning approaches such as the one developed in the CORONET project with recent research in the field of experience management, further synergies can be expected. For example, systematically collected and managed experience can provides the starting point for the preparation and design of learning materials as well as an idea of how to incorporate learning materials into an experience management system. This combination of experience based knowledge management and e-learning offers several benefits for the user:

- First of all, there is no need for the user to decide for the one approach or the other, both can be used in an integrated fashion.
- From the experience management perspective, the advantage is that additional learning materials can be offered if necessary for the user's current work process or his current interests. These learning materials supplement the available

experience packages and, thus, really enable and combine case-based learning [Sch82] and situated learning [KS99].

- From the e-learning perspective, learning materials can be based on already available experience. In addition, learning materials can be reused, feedback for them can be collected, and a continuous improvement process can be established.

A vision of how the integration of experience management with e-learning in a specialized application for software engineering process learning could look like the framework presented in Figure 12.

The integration framework consists of four components: presentation layer, repository, communities of practices, and maintenance component. The presentation layer is the interface to the regular user. It provides (a) uniform access to the information residing within the EB, (b) stores the user preferences and settings, and (c) aggregates information based on those preferences. The repository contains the explicitly captured and consolidated experience of an organization. Starting with a combination of business process descriptions and lessons learned is reasonable, since this combination is likely to reveal synergies [DA+01]. Furthermore, business processes and lessons learned could act as a starting point for further EM activities [DJ01]. The communities of practice (CoP) component is a forum for the members of an organization to discuss current problems, questions and open issues, and to evaluate the content of the EB. Finally, the maintenance component supports maintaining and developing the content of the EB (i.e., the data within the repository) and the services offered to the organization (via the Presentation Layer). Text mining techniques are used for analyzing discussions and other textual contributions of organization members or external consultants.

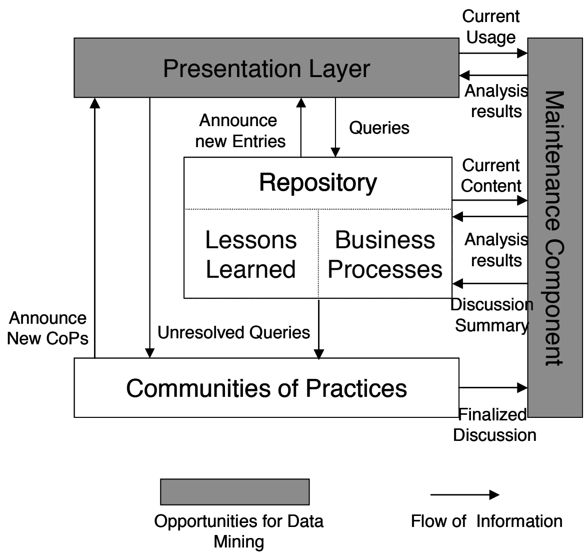


Figure 12. Integrating experience management with process learning, discourses and their analysis, maintenance, and e-learning

Generally, it is important to make sure that systems supporting combined knowledge management and e-learning do not offer – as most systems do today – monolithic, static, hierarchical structures. It is necessary that users can create new forums for specific user groups, that it is possible to search, sort, and filter information according to varying criteria; it is also necessary that the production and consumption of multi-media contents is supported, and that there is a complex information (announcement) mechanism. For example, when content is put into a discussion forum, the authors should be automatically informed when others put annotations to the content, when a new content is directed to a specific audience, this audience should be automatically informed, etc. In addition, a good communication space has to provide means to support real cooperation between users. For example, it is important to facilitate the creation of document versions, and associated comparison mechanisms, and it is important to allow that several users can access the same resources (data and programs) at the same time, and that one user can guide others by putting additional information to the documents that can be read and mutually commented (tutoring and mentoring). Apart from communication (which might create new valuable permanent information in some cases), a good knowledge management system provides large document repositories as kind of digital background library. These repositories contain reports, manuals, etc. that were created within a specific organization, and additional electronic publications that can be retrieved from various external sources: electronic books and journals, dictionaries, standards, image databases, maps, audio- and video clips, etc. All these documents do not necessarily have to be locally available but might as well be accessible via links from other web servers and databases.

To summarize, one can say that in an integrated knowledge management and e-learning system users should not only be able to read (consume) the offered information, but that they should be enabled to work with it. This requires functionality for (semi-)automatic personalization and adaptation, as well as for annotation, recording of discussion threads, help requests to peers and experts, etc., and for communication between users and between users and documents. The last point yields the vision of "active documents" [HM00, Mau01]. It means that a future user who is sitting in front of a PC screen and types in a question will receive a unique answer or a list of possible answers – and all this without any human intervention. With the framework proposed in this article, which particularly focussed on the integration of a modern approach to e-learning with recent advances in experience management, the fulfillment of this and many other visions might become reality in a not too far away future.

6 Acknowledgement

The work presented in this chapter was partly supported by the European Commission (EU-Project CORONET / Grant No. IST-1999-11634) and the German Ministry of Education and Science (BMBF-Projects indiGo / Grant No. 01 AK 951 A).

7 References

[AB+99] Aha, D.W., Becerra-Fernandez, I., Maurer, F., and Muñoz-Avila, H. (Eds.) (1999). *Exploring Synergies of Knowledge Management and Case-Based Reasoning: Papers from the AAAI 1999 Workshop* (Technical Report WS-99-10). Menlo Park, CA: AAAI Press.

[AB+99a] Althoff, K.-D., Bomarius, F., Müller, W. & Nick, M. (1999). Using a Case-Based Reasoning for Supporting Continuous Improvement Processes. In: P. Perner (ed.), *Proc. German Workshop on Machine Learning*, Technical Report, Institute for Image Processing and Applied Informatics, Leipzig, 8 pages.

[AB+00] Althoff, K.-D., Birk, A., Hartkopf, S., Müller, W., Nick, M., Surmann, D. & Tautz, C. (2000). Systematic Population, Utilization, and Maintenance of a Repository for Comprehensive Reuse. In G. Ruhe & F. Bomarius (Eds.), Learning Software Organizations - Methodology and Applications, Springer Verlag, Lecture Notes in Computer Science, LNCS 1756, 25-50.

[AB+02] Althoff, K.-D., Becker-Kornstaedt, U., Decker, B., Klotz, A., Leopold, E., Rech, J. & Voss, A. (2002). The indiGo Project: Enhancement of Experience Management and Process Learning with Moderated Discourses. In P. Perner (Ed.), Data Mining in E-Commerce, Medicine and Knowledge Management, Springer Verlag, Lecture Notes in Computer Science

[AD+01] Althoff, K.-D., Decker, B., Hartkopf, S., Jedlitschka, A., Nick, M. & Rech, J. (2001). Experience Management: The Fraunhofer IESE Experience Factory. In P. Perner (ed.), *Proc. Industrial Conference Data Mining*, Leipzig, 24.-25. Juli 2001, Institut für Bildverarbeitung und angewandte Informatik

[ADK98] Abecker, A., Decker, S. & Kühn, O. (1998). Organizational Memory. *Informatik Spektrum 21*:213-214.

[AFM01] Althoff, K.-D., Feldmann, R. & Müller, W. (eds.) (2001). *Advances in Learning Software Organizations*. Springer Verlag, LNCS 2176, September 2001.

[Aha99] Aha, D.W. (1999). The AAAI-99 KM/CBR Workshop: Summary of Contributions. *Proceedings of the ICCBR ´99 Workshops*, II-37–II-44. Technical Report, LSA-99-03E, Department of Computer Science, University of Kaiserslautern: Center for Learning Systems and Applications.

[AK+89] Althoff, K.-D., Kockskämper, S., Maurer, F., Stadler, M. and Wess, S. (1989). Ein System zur fallbasierten Wissensverarbeitung in technischen Diagnosesituationen. In: Retti, J. and Leidlmeier, K. (eds.), *5th Austrian Artificial-Intelligence-Conference*, 65-70, Springer Verlag.

[Alt01] Althoff, K.-D. (2001). Case-Based Reasoning. In: S.K. Chang (Ed.), Handbook on Software Engineering and Knowledge Engineering. Vol. 1 “Fundamentals”, World Scientific, 549-588.

[AM00] Althoff, K.-D. & Müller, W. (eds.) (2000). *Learning Software Organizations*. Proc. of the 2nd Internat. Workshop (LSO'00), Oulu, Finland.

[ANT99] Althoff, K.-D., Nick, M. & Tautz, C. (1999). An Application Implementing Reuse Concepts of the Experience Factory for the Transfer of CBR System Know-How. In: E. Melis (ed.), *Proc. of the Seventh German Workshop on Case-Based Reasoning* (GWCBR'99).

[AW97] Althoff, K.-D. & Wilke, W. (1997). Potential uses of case-based reasoning in the experience-based construction of software systems. In: R. Bergmann & W. Wilke (eds.), *Proceedings of the 5th German Workshop in Case-Based Reasoning (GWCBR’97),* LSA-97-01E, Center for Learning Systems and Applications (LSA), University of Kaiserslautern.

[AW00] Aha, D., and Weber, R., (eds.) (2000). *Proceedings of the Workshop on Intelligent Lessons Learned Systems at 17th National Conference on AI (AAAI-00)*. American Association for Artificial Intelligence.

[BaS87] Bartsch-Spörl, B. (1987). Ansätze zur Behandlung von fallorientiertem Erfahrungswissen in Expertensystemen. *KI, 4*, 32-36.

[Bas85] Basili, V.R. (1985). Quantitative evaluation of software methodology. In *Proceedings of the First Pan-Pacific Computer Conference, Melbourne, Australia, September 1985*.

[BBB75] Brown, J. S., Burton, R. R. & Bell, A. G. (1975). SOPHIE: A Step towards a Reactive Learning Environment. In: International Journal of Man Machine Studies, 6, 7, 670 ff.

[BBK82] Brown, J. S., Burton, R. R. & de Kleer, J. (1975). Pedagogical, Natural Language, and Knowledge Engineering Techniques in SOPHIE I, II, and III. In [SB82].

[BB+99] Bergmann, R., Breen, S., Göker, M., Manago, M., and Wess, S. (1999). *Developing Industrial Case-Based Reasoning Applications - The INRECA-Methodology*. Springer Verlag, LNAI 1612.

[BCC92] Basili, V.R., Caldiera, G. & Cantone, G. (1992). A reference architecture for the component factory. *ACM Transactions on Software Engineering and Methodology, 1(1).*

[BCR94] Basili, V.R., Caldiera, G. & Rombach, H.D. (1994). The experience factory. In: J. J. Marciniak (ed.), *Encyclopedia of Software Engineering*, Vol. 1, Wiley, New York, 469-476.

[BE+01] Brandt, M., Ehrenberg, D., Althoff, K.-D. & Nick, M. (2001). Ein fallbasierter Ansatz für die computergestützte Nutzung von Erfahrungswissen bei der Projektarbeit. In H. U. Buhl, A. Huther &

B. Reitwiesner (Hrsg.), *Information Age Economy, Proc. der 5. Internationalen Tagung Wirtschaftsinformatik (WI'01),* Heidelberg: Physica Verlag, 251-264.

[BJA01] Bartsch-Spörl, B., Jargon, C. & Althoff, K.-D. (2001). *Wissensmanagement in der Praxis - Erfahrungen aus erfolgreichen und weniger erfolgreichen WM-Projekten.* Workshop auf der 1. Konferenz Professionelles Wissensmanagement: Erfahrungen und Visionen, Baden-Baden 14.-16. März 2001, (http://demolab.iese.fhg.de:8080/AK-Praktisches-Wissensmanagement/WM2001-WS5/).

[BLC01] Basili, V.R., Lindvall, M. & Costa, P. (2001). Implementing the Experience Factory concepts as a set of Experience Bases. *Proc. Internat. Conf. on Software Engineering and Knowledge Engineering 2001 (SEKE'01)*

[BN01] Brandt, M. & Nick, M. (2001). Computer-Supported Reuse of Project Management Experience with an Experience Base. In *[AFM01],* 178-191.

[BO96] Bacher, Ch. & Ottmann, Th. (1996). Tools and Services for Authoring on the Fly. In: Proceeding of ED-MEDIA, Boston, 7-12.

[BR88] Basili, V.R. & Rombach, H.D. (1988). The TAME Project: Towards improvement-oriented software environments. *IEEE Transactions on Software Engineering SE-14(6),* 758-773, June 1988.

[BR91] Basili, V.R. & Rombach, H.D. (1991). Support for comprehensive reuse. *IEEE Software Engineering Journal 6(5)*, 303-316, September 1991.

[BR00] Broomé, M. & Runeson, P. (2000). Technical requirements for the implementation of an experience base. In: *[RB00],* 87-102.

[BSL99] Basili, V.R., Shull, F., and Lanubile, F. (1999). Building Knowledge through Families of Experiments. *IEEE Trans. on Software Engineering 25, no. 4*, 456-473.

[BT98] Birk, A.; Tautz, C. (1998). Knowledge Management of Software Engineering Lessons Learned. *Proceedings of the Tenth Conference on Software Engineering and Knowledge Engineering*; San Francisco Bay; USA. Skokie, Illinois, USA: Knowledge Systems Institute; 1998.

[Bus45] Bush, V. (1945). As we may think. In: Atlantic Monthly, July 1945, 101-108.

[Buß90] Bußmann, Hadumod (1990). *Lexikon der Sprachwissenschaft*, Kröner: Stuttgart.

[Car70] Carbonell, J. R. (1970). AI in CAI: An Artificial Intelligent Approach to Computer Assisted Instruction. In: IEEE Transactions on Man Machine Systems 4, 11, 190 ff.

[Cau00] Caumanns, J. (2000). Automatisierte Komposition von wissensvermittelnden Dokumenten für das World Wide Web. PhD Thesis. Cottbus: Tech. Univ.

[CBN89] Collins, A., Brown, J. S. & Newman, S. E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing and mathematics. In [Res89], 453-494.

[CD00] Conradi, R. & Dingsoyr, T. (2000). In F. Bomarius & M. Oivo (eds.), *Product Focused Software Process Improvement (PROFES'00),* Springer Verlag LNCS 1840, 391-406.

[Cro59] Crowder, N. A. (1959). Automatic Tutoring by Intrinsic Programming. In [LG59].

[CRS00] Chen-Burger,Y.H., Robertson, D. & Stader, J. (2000). A Case-Based Reasoning Framework for Enterprise Model Building, Sharing and Reusing. *Proc. ECAI 2000 Workshop on Knowledge Management and Organizational Memories*, Berlin.

[DA02] Decker, B., Althoff, K.-D., Nick, M., Jedlitschka, A., Tautz, C. & Rech, J. (2002). Die Fraunhofer IESE Experience Factory „Corporate Information Network (CoIN)" – Ein Beispiel für Geschäftsprozessorientiertes Wissensmanagement in Software Organisationen. In A. Abecker, K. Hinkelmann, H. Maus. & H.-J. Müller (Hrsg.), *Geschäftsprozessorientiertes Wissensmanagement*, Springer X-pert Press, 367-391.

[DG+98] Dietinger, Th., Gütl., Ch., Maurer, H., Pivec, M. & Schmaranz, K. (1998). Intelligent Knowledge Gathering and Management as New Ways of an Improved Learning Process. In: Proceedings of WebNet 98, Orlando, AACE, Charlottesville, USA, 244-249.

[DG+99] Dietinger, Th., Gütl., Ch., Maurer, H., Scherbakov, N. & Schmaranz, K. (1999). Kriterien für ein flexibles System für die Unterstützung von Ausbildungsaufgaben mit moderner Web-Technologie. HMD - Praxis der Wirtschaftsinformatik 20, 22-33.

[Din00] Dingsoyr, T. (2000). An Evaluation of Research on Experience Factory. In *[AM00],* 55-66.

[DJ01] Decker, B. & Jedlitschka, A. (2001). The Integrated Corporate Information Network iCoiN: A Comprehensive, Web-based Experience Factory. In: *[AFM00],* 192-206.

[DN98] DiBella, A. J. & Nevis, E. C. (1998). How Organizations Learn – An Integrated Strategy for Building Learning Capability. Jossey-Bass Publishers, San Francisco.

[EK+02] Eickeler, S., Kindermann, J., Larson, M., Leopold, E. & Paaß, G. (2002). Classification of Spoken Documents using SVM and Sub-word Units. *submitted to Neurocomputing, Special Issue on Support Vector Machines.*

[Fel02] Feldmann, R.L. (2002). *A Product Line for Experience Repositories.* PhD. Thesis, Department of Computer Science, University of Kaiserslautern (to appear).

[FI+02] Friedrich, R., Iglezakis, I., Klein, W. & Pregizer, S. (2002). Experience-based decision support for project management with case-based reasoning. In: *[MS02],* 139-150.

[FM95] Flinn, B. & Maurer, H. (1995). Levels of Anonymity. In: Journal of Universal Computer Science 1, 1, 35-47.

[GAP02] Grützner, I.; Angkasaputra, N. & Pfahl, D. (2002). A Systematic Approach to Produce Small Courseware Modules for Combined Learning and Knowledge Management Environments. In: Association for Computing Machinery (ACM): The Fourteenth International Conference on Software Engineering and Knowledge Engineering. SEKE'2002. New York, 533-539.

[GH01] Gunzenhäuser, R. & Herczeg, M. (2001). Lehren und Lernen im Zeitalter der neuen digitalen Medien. In: i-com, 0/2001.

[GHD99] Griffiths, A.D., Harrison, M.D. & Dearden, A.M. (1999). Case Based Reasoning Systems for Knowledge Mediation. in M. A. Sasse and C. Johnson (eds.), *Proceedings of Interact 99,* (Edinburgh), IOS Press. pp. 425-433.

[Goo99] Goodall, A. (ed.) (1999). Survey of Knowledge Management Tools - Part I & II, volume 8. *Intelligence in Industry*, January & February 1999.

[Goo87] Goodman, D. (1987). The Complete HyperCard Handbook. Bantam Books: New York.

[GPR02] Grützner, I.; Pfahl, D. & Ruhe, G.: Systematic courseware development using an integrated engineering style method. In: Natural and Artificial Intelligence Systems Organization (NAISO): Networked Learning in a Global Environment, Challenges and Solutions for Virtual Education. NL'2002. World Congress - Proceedings. Millet: ICSC-NAISO Academic Press. 8 pp.

[Gro99] Gronau, N. (ed.) (1999). Lernende Unternehmen. *Industrie Management 15 (1999) 6*, GITO-Verlag.

[HA96] H•ebíc•eck, L. & Altmann, G. (1996). The Levels of Order in Language, *Glottometrika 15*, 38-61.

[Hal88] Halasz, F. G. (1988). Reflections on NoteCards: Seven Issues for the Next Generation of Hypermedia Systems. In: Communications of the ACM, July 1988, 836-852.

[Hal96] Haley, T.J. (1996). Software process improvement at Raytheon. IEEE Software 13(6), 33-41.

[Hen95] Henninger, S. (1995). Developing domain knowledge through the reuse of project experiences. In M. Samadzadeh (ed.), *Proc. of the Symposium of Software Reusability (SSR'95),* 186-195.

[HH+95] Harasim, L., Hiltz, S. R., Teles, L. & Turoff, M. (1995). Learning Networks: A Field Guide to Teaching and Learning Online. Cambridge: MIT Press.

[HM00] Heinrich, E. & Maurer, H. (2000). Active Documents: Concepts, Implementation and Applications. Journal of Universal Computer Science 6, 12, 1197-1202.

[HMS99] Helic, D.; Maglajlic, S. & Scherbakov, N. (1999). Educational Materials on the WEB: data Modelling Approach. In: Proceedings of 22nd International Symposium on Multimedia and Hypermedia systems, MIPRO'99, Rijeka, Croatioa, pp. 139-142.

[HMS00] Helic, D., Maurer, H. & Scherbakov, N. (2000). Web Based Training: What do we expect from the system. In: Proceedings of ICCE 2000 – 8th International Conference on Computers and Education, Taiwan, 1689-1694.

[HMS01a] Helic, D., Maurer, H. & Scherbakov, N. (2001). Mentoring Sessions: Increasing the Influence of Tutors on the Learning Process in WBT Systems. In: Proceedings of WEBNET-2001 World Conference of the Web Society. Charlottesville, USA, 515-519.

[HMS01b] Helic, D., Maurer, H. & Scherbakov, N. (2001). Knowledge Domains: A Global Structuring Mechanism for Learning Resources in WBT Systems. In: Proceedings WEBNET-2001 World Conference of the Web Society. Charlottesville, USA, 509-514.

[HMS01c] Helic, D., Maurer, H. & Scherbakov, N. (2001). Accessing Best-Match Learning Resources in WBT Environments. In: Proceeding World Conference "Educational Multimedia and Hypermedia" ED-MEDIA'01. Charlottesville, USA, 206-212.

[HMS01d] Helic, D., Maurer, H. & Scherbakov, N. (2001). Creating and Maintaining Semantic Nets of Learning Resources in a WBT Environment. In: Proceedings of 24th International Symposium on Multimedia and Hypermedia Systems MIPRO `01. Rijeka, Croatia, 136-143.

[HMS02] Helic, D., Maurer, H. & Scherbakov, N. (2002). Aspects of Collaborative Authoring in WBT Systems. In: Proc. Intl. Conf. on Advances in Infrastructure for Electronic Business, education, Science, and Medicine on the Internet, SSGRR 2002w. CD-ROM publication (ISBN 88-85280-62-5), paper 37.

[HM+01] Helic D., Maurer, H., Lennon, J. & Scherbakov, N.: Aspects of a Modern WBT System. In: Proc. Intl. Conf. on Advances in Infrastructure for Electronic Business, Education, Science, and Medicine on the Internet, SSGRR 2001. CD-ROM publication (ISBN:88-85280-61-7), paper 38.

[HSW91] Humphrey, W.S., Snyder, T.R. & Willis, R.R. (1991). *Software process improvement at Hughes Aircraft. IEEE Software 8*, 11-23.

[HSW98] Houdek, F., Schneider, K. & Wieser, E. (1998). Establishing experience factories at Daimler-Benz: An experience report. *Proc. 20th Internat. Conf. on Software Engineering (ICSE'98).*

[JH+00] Johansson, C., Hall, P. & Coquard, M. (2000). "Talk to Paula and Peter - They Are Experienced" - The Experience Engine in a Nutshell. In G. Ruhe & F. Bomarius (Eds.), Learning Software Organizations - Methodology and Applications, Springer Verlag, Lecture Notes in Computer Science, LNCS 1756, 171-185.

[Joa98] Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. *Proc 10th European Conference on Machine Learning (ECML 98*), LNCS 1398, Springer Verlag, 137-142.

[KD+02] Kindermann, J. & Diederich, J. & Leopold, E. & Paaß, G. (2002). Identifying the Author of a Text with Support Vector Machines; accepted at *Applied Intelligence*.

[KM+00] Kalfoglou, Y., Menzies, T., Althoff, K.-D. & Motta, E. (2000). Meta-Knowledge in Systems Engineering: Panacea or Undelivered Promise? *The Knowledge Engineering Review 15(4),* December 2000.

[Koh01] Kohonen, T. (2001). *Self-organizing maps*. Springer Verlag.

[Kol93] Kolodner, J.L. (1993). *Case-Based Reasoning*. Morgan Kaufmann Publishers.

[KPL01] Kindermann, J. & Paaß, G. & Leopold, E. (2001). Error Correcting Codes with Optimized Kullback-Leibler Distances for Text Categorization. *Proc. PKDD'01*, 3 - 7 September 2001 in Freiburg.

[KS99] Kearsley, G. & Shneiderman, B. (1999). Engagement Theory: A framework for technology-based teaching and learning. http://home.sprynet.com/~gkearsley/engage.htm.

[Leh00] Lehner, F. (2000). *Organisational Memory - Konzepte und Systeme für das organisatorische Lernen und das Wissensmanagement*. Carl Hanser Verlag.

[LG59] Lumsdane, A. A. & Glaser, R. (1959). Teaching Machines and Programmed Learning. Washington.

[LK02] Leopold, Edda & Kindermann, Jörg (2002). Text Categorization with Support Vector Machines. How to Represent Texts in Input Space?; in: *Machine Learning 46*, 423 - 444.

[LW98] Leake, D. & Wilson, D. (1998). Categorizing case-base maintenance: Dimensions and directions. In Smyth, B. & Cunningham, P. (Eds.) *Advances in Case-Based Reasoning (EWCBR'98),* Lecture Notes in Artificial Intelligence, Springer, Berlin, Heidelberg, 196-207.

[Mau98] Maurer, H. (1998). A critical look at current Web-Based Training efforts. In: Proceedings of ICCE'98, Bejing, CHEP, Bejing and Springer Heidelberg, 30-33.

[Meh02] Mehler, Alexander (2002): Hierarchical analysis of text similarity data. In: T. Joachims & E. Leopold (eds.), *Künstliche Intelligenz 2/02*, Schwerpunkt Textmining, 12-16.

[Men98] Menzies, T. (1998). Knowledge maintenance: The state of the art. *The Knowledge Engineering Review.*

[Mer97] Merkl, D. (1997). Exploration of Document Collections with Self-Organizing Maps: A Novel Approach to Similarity Representation, a novel approach to similarity visualization. *Proceedings of the European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD'97*), (Trondheim, Norway. June).

[MO91] Murray, M. & Owen, M. A. (1991). Beyond the Myths and Magic of Mentoring: How To Facilitate an Effective Mentoring Program. San Francisco: Jossey-Bass.

[MP90] McGarry, F. & Pajersky, R. (1990). Towards Understanding Software - 15 Years in the SEL. *Proc. of the 15th Annual Software Engineering Workshop*, NASA Goddard Space Flight Center, Greebelt, MD, Software Engineering Laboratory Series, SEL-90-006, Nov. 1990.

[MS98] Maurer, H. & Scherbakov, N. (1998). Course Development Environment for Hyperwave. In: Proceedings of WebNet 98, Orlando, AACE, Charlottesville, USA, 623-628.

[MS01] Maurer, H & Schinagl, W. (2001) Automatisch wachsende Wissensbestände in eLearning Situationen. In: Proceedings Learntec'2001. Karlsruhe, 453-459.

[MS02] Minor, M. & Staab, S. (eds.) (2002). *1st German Workshop on Experience management – Sharing Experiences about the Sharing of Experience*. Berlin, March 7-8, 2002, Lecture Notes in Informatics, Gesellschaft für Informatik (Bonn).

[NAT01] Nick, M., Althoff, K.-D. & Tautz, C. (2001). Systematic Maintenance for Corporate Experience Repositories. Computational Intelligence 17(2), 364-386.

[NF00] Nick, M., and Feldmann, R. (2000). Guidelines for evaluation and improvement of reuse and experience repository systems through measurement programs. In *Proc. 3rd European Conference on Software Measurement (FESMA-AEMES 2000),* Madrid, Spain, Oct. 2000.

[OW00] Ostrand, T. J. & Weyuker, E. J. (2000). learning environment for software testers at AT&T. In: Proceedings of the 2nd workshop on learning software organizations, Fraunhofer IESE, Kaiserslautern, Germany.

[PA+01] Pfahl, D., Angkasaputra, N., Differding, C. & Ruhe, G. (2001). CORONET-Train: A Methodology for Web-Based Collaborative Learning in Software Organisations. In: Althoff, K.-D. et al. (ed.): Advances in Learning Software Organizations. Third International Workshop, LSO 2001 - Proceedings. Berlin: Springer-Verlag, Lecture Notes in Computer Science 2176, 37-51.

[PT+02] Pfahl, D., Trapp, S., de Teresa, J., Oliveira, J., Stupperich, M., Rathert, N., Molu, R., Scherbakov, N. & D'Ambra, J. (2002). CORONET Final Report, Fraunhofer IESE, Technical Report no. 045.02/E.

[Rae00] Raelin, J. A. (2000) Work-Based Learning – The New Frontier of Management Development. Prentice-Hall, New Jersey.

[RB00] Ruhe, G. & Bomarius, F. (eds.) (2000). *Learning Software Organizations - Methodology and Applications*. Springer Verlag, LNCS 1756.

[RDA01] Rech, J., Decker, B. & Althoff, K.-D (2001). Using Knowledge Discovery Technology in Experience Management Systems. *Proc. Workshop "Maschinelles Lernen (FGML01)", GI-Workshop-Woche „Lernen – Lehren – Wissen – Adaptivität (LLWA01)“*, Universität Dortmund, 8.-12. Okt. 2001.

[Res89] Resnick, L. B. (1989). Knowing, learning and instruction. Essays in the honour of Robert Glaser. Hillsdale (NJ), Erlbaum.

[Rom96] Rombach, H.D. (1996). New institute for applied software engineering research. *Software Process Newsletter No 7*, 12-14, Fall 1996.

[Rom98] Romhardt, K. (1998). *Die Organisation aus der Wissensperspektive - Möglichkeiten und Grenzen der Intervention*. Wiesbaden: Gabler Verlag.

[RRM99] Reinmann-Rothmeier, G. & Mandl, H. (1999). Teamlüge oder Individualisierungsfalle? Eine Analyse kollaborativen Lernens und deren Bedeutung für die Förderung von Lernprozessen in virtuellen Gruppen, Forschungsberichte LMU, 115, Lehrstuhl für empirische Pädagogik und pädagogische Psychologie, November 1999.

[RU89] Rombach, H.D. & Ulery, B.D. (1989). Establishing a measurement based maintenance improvement program: Lessons learned in the SEL. *Proc. of the Conference on Software Maintenance,* 50-57, IEEE Computer Society Press, October 1989.

[SA77] Schank, R.C. & Abelson, R. (1977). *Scripts, Plans, Goals, and Understanding*. Lawrence Erlbaum Associates, Hillsdale, New Jersey.

[SB82] Sleeman, D. & Brown, J. S. (1982). Intelligent Tutoring Systems. Academic Press: London.

[Sch82] Schank, R.C. (1982). Dynamic Memory: A Theory of Learning in Computers and People. Cambridge University Press.

[Ses96] Seshagiri, G. (1996). Continuous process improvement: Why wait till level 5? *Proc. 29th Hawaii Internat. Conf. on System Sciences*, 681-692, IEEE Computer Society Press.

[Ski53] Skinner, B. F. (1953). Science and Human Behavior. New York: Mac Millan.

[Ski54] Skinner, B. F. (1954). Science of Learning and the Art of Teaching. In Harvard Educational Review 2, 24, 86 ff.

[SW98] Stolpmann, M. & Wess, S. (1998). *Optimierung der Kundenbeziehungen mit CBR systemen-Intelligente Systeme für E-Commerce und Support,* Addison Wesley Longmann (Business & Computing), Bonn.

[TA97] Tautz, C. & Althoff, K.-D. (1997). Using Case-Based Reasoning for Reusing Software Knowledge. In: D. Leake and E. Plaza (eds.), *Case-Based Reasoning Research and Development, Second International Conference on Case-Based Reasoning (ICCBR97),* Springer Verlag, 156-165.

[TA00] Tautz, C. & Althoff, K.-D. (2000). A Case Study on Engineering Ontologies and Related Processes for Shar-ing Software Engineering Experience. In: Proc. 12th International Conference on Software Engi-neering and Knowledge Engineering (SEKE'00)

[TAN00] Tautz, C., Althoff, K.-D. & Nick, M. (2000). A Case-Based Reasoning Approach for Managing Qualitative Experience. In: *[AW00].*

[Tau00] Tautz, C. (2000). *Customizing Software Engineering Experience Management Systems to Organizational Needs*. Ph. D. Thesis., Dept. of Computer Science, University of Kaiserslautern, Germany; 2000; Stuttgart: Fraunhofer IRB Verlag.

[Tho14] Thorndike, E. L. (1914). The Psychology of Learning.

[Tho32] Thorndike, E. L. (1932). Fundamentals of Learning.

[WAB01] Weber, R., Aha, D.W., & Becerra-Fernandez, I. (2001). Intelligent lessons learned systems. Expert Systems with Applications, 20, 17-34.

[War98] Wargitsch, C. (1998). *Ein Beitrag zur Integration von Workflow- und Wissensmanagement unter besonderer Berücksichtigung komplexer Geschäftsprozesse*. Doctoral Dissertation, University of Erlangen-Nürnberg, Germany.

[Wen98] Wenger, E. (1998). Communities of Practice: Learning, Meaning, and Identity. Cambridge University Press, New York.

[WG01] Weber, R. & Gresse von Wangenheim, C. (eds.) (2001). Proceedings of the Workshop Program at the Fourth International Conference on Case-Based Reasoning, Technical Note AIC-01-003, Washington, DC: Naval Reasearch Laboratory, Naval Research Center for Applied Research in Artificial Intelligence.

[WO+00] Weyuker, E. J., Ostrand, T. J., Brophy, J. A. & Prasad, R. (2000). Clearing a career path for software testers. In: IEEE Software, March 2000, 76-82.