Using Case-Based Reasoning for User Modeling in an Experience Management System

Andreas Jedlitschka & Klaus-Dieter Althoff

Fraunhofer Institute Experimental Software Engineering (IESE),
Sauerwiesen 6, D-67551 Kaiserslautern
{jedl|althoff}@iese.fhg.de

1. Introduction

Experience-based continuous learning is essential for improving products, processes, and technologies in emerging as well as in established areas of business and engineering science. It can be facilitated by case-based organizational learning, meaning that relevant experience is captured in the form of cases for reuse in a corporate experience repository (case base; experience base; EB). For obvious reasons, learning from experience needs to be a permanent endeavor. Thus, an organization has to handle a "continuous stream of experience." For this purpose a learning organization, called "Experience Factory" (EF; Basili et al. 1994, Althoff et al. 2000), was established at Fraunhofer IESE, with the COrporate Information Network (COIN) initiative (Althoff et al. 2001). The objectives of COIN are to provide users with valuable information/knowledge at the right time, in an adequate representation, and within the actual context ("just-in-time").

In this paper, we first introduce the Experience Management Content Framework of our IESE EF to clarify the given knowledge management context. Enhancing the presented approach, we introduce new strategies to capture, process, disseminate, and exchange knowledge. Case-based reasoning (CBR) is used for both knowledge modeling/retrieval/adaptation as well as for a "learning from example" based approach to user modeling in the sense of Weibelzahl and Weber (1999).

2. The Experience Management Content Framework of the IESE EF

The Experience Management Content Framework (EMCF) presented in Fig.1 is a further development of the knowledge management infrastructure set up by Tautz (2000). Tautz has demonstrated the benefits of COIN in an experiment. Until now we have gathered nearly two years of operational experience in maintaining COIN, and we have successfully adapted COIN to partners/customers. Based on this experience, we have widened the requirements of COIN towards an organization-wide information and knowledge management system. EMCF acts as a vision for comprehensive management of experience within an organization, thus, representing a generic blueprint of an EB.

The EMCF consists of four basic components: the Presentation Layer, the Repository, the Communities of Practice, and the Maintenance Component.

The *Presentation Layer* is the interface of the EB to the regular user. It (a) provides uniform access to the information residing within the EB, (b) stores user preferences and settings, and (c) adapts and aggregates information within the EB based on those preferences (Sec. 3).

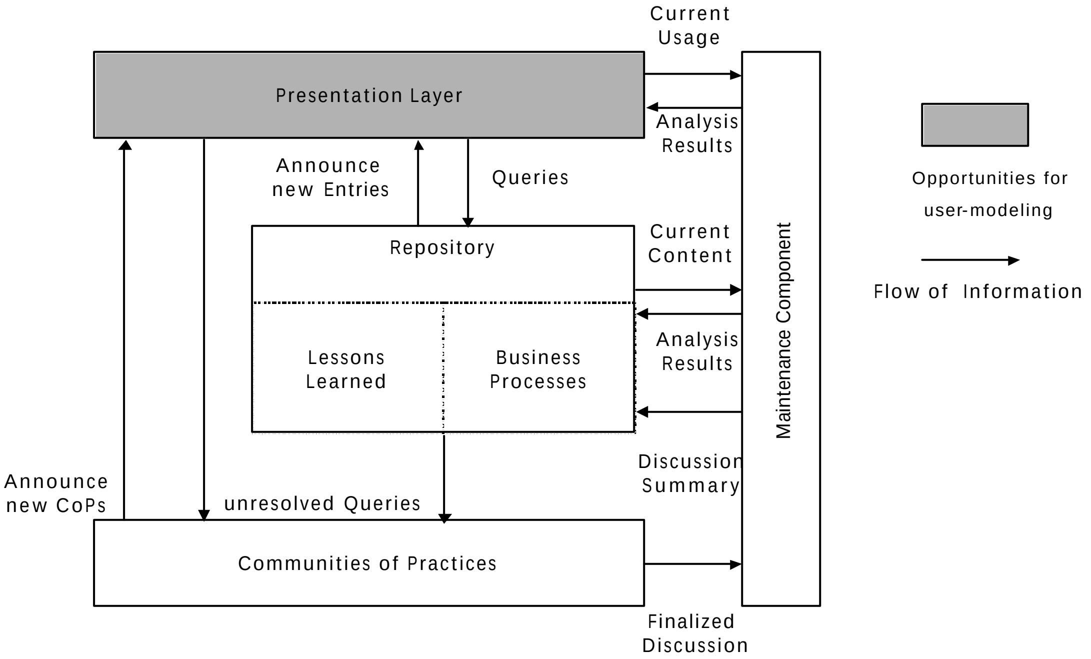

Fig.1 The Experience Management Content Framework (EMCF)

The *Repository* contains the explicitly captured and consolidated experience of an organization. A combination of business process descriptions and lessons learned was chosen as a starting point (Althoff et al. 2001). Further experience management activities can be set up on top of that base (Decker & Jedlitschka 2001)

The *Communities of Practice* component is a forum for the members of an organization to discuss current problems, questions, and open issues (Sec. 3).

Finally, the *Maintenance Component* supports the EF team in 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). This component offers a place for sensible application of data-mining methods: The content as well as the usage of the services can be analyzed to trigger, guide, or otherwise support maintenance activities.

3. New Strategies to Capture, Process, Disseminate, and Exchange Knowledge

Knowledge has actually been identified as the “fourth factor of production”[1]. Therefore, unstructured, non-personalized flooding with information can be counterproductive for building up and exchanging knowledge (Fischer & Ye 2001; Jameson 2001). To better support our employees, we are (a) moving from a “pull” to a “push” strategy in the sense of providing the right information at the right time (context-sensitive), (b) developing more flexible and faster mechanisms for sharing information, and (c) developing a method for aggregating and adapting information to users’ context and needs by using CBR.

The main challenge is to convince the users of the systems helpfulness. The users should remark a personal gain. Only this will bring him to spend some voluntary effort. At least a break even according to time/effort spent and time/effort gained should be reached (Kluge 1999).

[1] Besides work, capital, raw material (e.g. Stewart 1997)

3.1. “Push” of Information/Knowledge

We do not want to burden users with overhead for searching information or asking for experience. Our solution grants a single point of access, admission to all knowledge and information produced in an organization, only restricted by access rights defined by (a) the organization in the form of the employee’s role within it, (b) the projects and the corresponding role the employee plays, and (c) the owner of a piece of information. Therefore, a user interface has to be developed corresponding to the presentation layer shown in Fig.1.

With his login in combination with stored but also dynamic user data (organizational role, project roles, skills, and interests (Fig.4)) and a chosen view (e.g., concrete project, information channel), the user provides the actual context, for example: “project: x; role: developer; task: code testing” (the task is determined from the project plan). The given context is, on the one hand, used to build his individual navigation bar (e.g., below the topic projects, only those projects that he is a member of are listed). On the other hand, the context is necessary for the delivery of knowledge (knowledge is gained within a context, anyway). When the context (including the user model) is treated as a case, it can be compared with other contexts. CBR helps to identify similar contexts. Thus, it is possible to deliver knowledge gained within former similar contexts without an explicit user query (“push” of information). The user can ignore the delivery but, hopefully, he will at least evaluate the utility of the delivered information within his actual context. The evaluation is used on the one hand, to “educate” or “edge” agents for users’ business and personal information needs and, on the other hand, to get more accurate evidence in accordance with the usability of this information for other users as well. The agents observe users’ behavior (i.e., navigation), and they are also “responsible” for discovering desired information. Personal needs can be context-sensitive and/or free of user’s choice.

In the case of new, improved, or changed content, the user will be informed automatically, if he has registered for this service. This can happen by mail, directly within the news window of the application residing on the physical presentation layer. It is thought of as a multi-step news channel structure: The most important news, categorized by the author, are presented directly within the main window (not more than five), whereas the others are presented within their context. The user registers only once for those components he wants to be informed of.

Additionally, he can send specific queries to the EB (“pull”) (see Fig.2).

We plan to support this approach using a combination of structural and textual CBR. While case-based retrieval is used to find the most similar knowledge items based on the login/context information and/or the specific query, a user case base is used for bridging the gap between the known information about the user and the knowledge the user is “really” currently looking for (Weibelzahl & Weber 1999).

3.2. Community of Practice Base (CoP)

To get information, current users have to send a query to the EB. As practice shows, sometimes there is no appropriate case available for the specified problem. The user has to find his own solution, which tends to be available only to a very small group of people, unless he tells the COIN team, who implements the organizational part of the EF (Basili et al. 1994), about the gained experience. According to Nonaka and Takeuchi (1995), this means the externalization of implicit knowledge towards explicit knowledge. Currently, project experiences are collected periodically and at the end of a project, using project analysis interviews (i.e., a structured interview for acquiring lessons learned from project members). In those interviews the project members tell their experiences to the interviewing member of the EF team. The EF team is responsible for extracting and deriving lessons learned in the form of guidelines, observations, and problems, and for putting them into the EB. For some problems occurring within the projects, this process is too slow. To present a solution, we are aiming at extending the EF through a more flexible concept, namely communities of practice (CoPs). They can be used as tool support for task-oriented collaborative learning, pointing out team-learning and collective intelligence (Kluge 1999).

CoPs handle specific problems for which there is no information in the EB available, so far. In such a case, and if the user agrees, the query is forwarded to the project-specific community of practice (PCoP) (see Fig.2). They are initiated by the project manager, who also arranges project membership with the line manager and the desired employees. The project members then are already members of the PCoP. Every project member who currently has got a view on this project will see the question nearly at the same time. This can also include customers or partners outside the organization. They can assist by providing their own experience and, simultaneously, they extend the knowledge base. At the end of a project, the content of the respective PCoP is (manually) checked for integration into the knowledge repository. After a certain time the PCoP is closed.

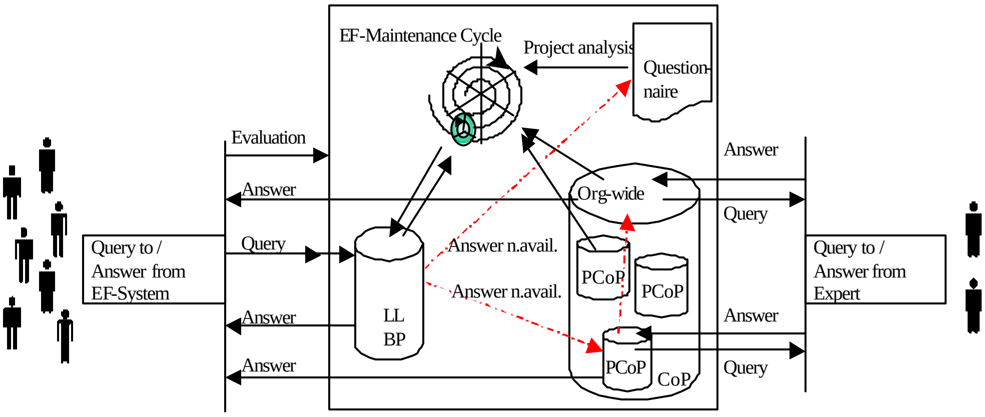

Fig.2 Correlation: Experience base (EB) and communities of practice (CoP)

Intuitively, the CoP supports the collection of tacit, personal knowledge. If after a while (the asking user can give a deadline) nobody answered the question (sufficiently), it is sent to an organization-wide CoP (if the user agrees), where every user can answer the question. In addition, it could become one of the duties of some very experienced IESE members to look up the CoPs at least once a week. The CoPs, also the PCoPs for bigger projects, may be divided by topics according to expert domains. Everyone, especially the experts, can propose CoPs for a specific topic. Employees can apply to the experts for membership. The EF team is responsible for initiation and maintenance. In order fro the system to find the right CoP automatically, the user has to classify the question. If there is no unique classification, the question is forwarded to all topic CoPs. An ontology-like structure of the CoPs can be used for building up a hierarchy supporting the classification.

After obtaining the answer, the asking user should be able to evaluate the utility of the given answer according to his specific context by giving bonus points. These points can be gathered and an award like the “Expert of the Month/Year”, together with a financial bonus, can be instantiated (for a knowledge market, a new currency for knowledge units, similar to a stock market (supply and demand) can be considered). This is expected to motivate people to use this feature of COIN. On the other hand, the user evaluation supports maintenance and further development of the system.

To support project analyses in a more specific way, such questions can also extend the questionnaire for the interview. The project member, who sends the query now, should be able to answer the question because of experiences that solved the task, which were given by others through the CoP or made by himself. In this context it seems to be important to mention that the collection of both positive *and* negative experiences is necessary in the case of a knowledge network (Bartsch-Spörl et al. 2001). An approach to archive project-specific CoPs within the project will be developed to avoid loss of experience.

Another part of the operative work of an EF, besides the collection of experiences during project analysis, is the maintenance of the EB content (Nick et al. 2001). With every new input to the EB, existing cases can be confirmed or questioned. This work will be supported by the introduction of utility evaluation by the users. The EF Maintenance Cycle mentioned in Fig.2 shall symbolize the necessary activities. Rejected content can be discussed and widely evaluated using the CoP. At any rate, questions not yet answered or rejected content have to be considered as hints for maintenance, that is, in-depth analyses. The results will help to improve (a) EB content but also (b) information aggregation and adaptation, which includes education of the agents through the user by “carrot (bonus points) and stick (rejection)” (further motivation of the user through better performance of the agents). To save expenses, maintenance of the CoPs is done mainly automatically by using the given bonus points and timestamps. If the information is repeatedly regarded as worthless and the timestamp is reached, it is moved to trash. In contrast, information regarded as valuable is forwarded to the EF Maintenance Cycle for final inspection before it is stored in the EB.

3.3. Aggregation and Adaptation of Information

Every member of an organization or, more abstractly, every role, has different needs with regard to the granularity of information. Stepping higher on the organizational or project level, information has to be aggregated and adapted more and more with respect to the urgency and criticality. Therefore, different information pieces have to be fragmented (Fig.3). In accordance with users’ needs, the relevant fragments have to be chosen (adaptation) and brought together in a convenient way (aggregation). This concept is well known in data mining methodologies. Extending these approaches, we are dealing with experience in the form of un-/structured documents. Data mining shall support the gaining of valuable information to confirm/reject experience.

The user gets standard information in addition, with an attribute telling him about the degree of utility (personalized or evaluated experience) and the name of the author. Highly aggregated and adapted information can usually not be assigned to a unique input source. The level of aggregation and adaptation is then given to the user, so he is able to comprehend the outcome. Detailed information (source information) is available, on demand, which is especially of interest if a state is detected as being critical.

While project members need specific and in-depth information about their status within the project, the project leader is more interested in an overview of all project activities. For him, knowing that a deviation will occur (e.g., because of illness of a project member) is valuable information. Experiences from similar cases enriched with input from the risk plan can assist him in evaluating the critical potential of this state. If he detects a business-critical state, the information is forwarded on a "red-phone" channel to the respective persons. This channel will also be used if an addressed person does not react after a pre-defined period of time.

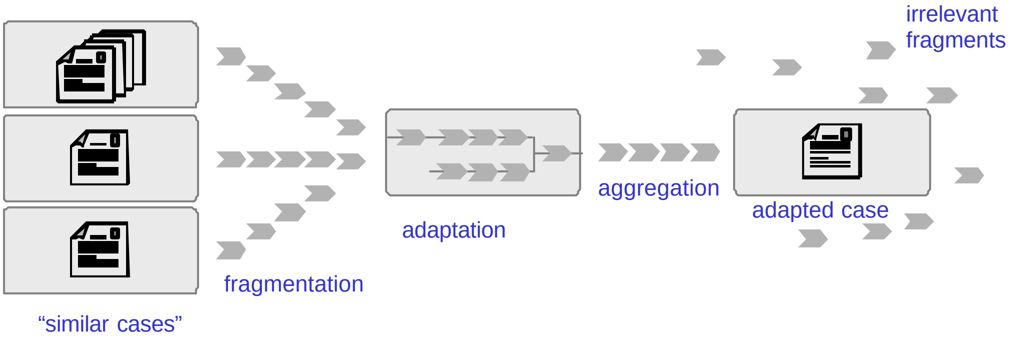

Fig.3 The principle of fragmentation, adaptation, aggregation

The approach of retrieving parts of information from different sources and configuring (Wilke 1999) and adapting (Bergmann 1996) these items to one item to be presented is analogous to the case adaptation step in the basic CBR process model (Aamodt & Plaza 1994). Techniques available here can be found in Althoff (1997) and Bergmann (2001). The technical realization of the aggregation and adaptation component resides within the representation layer of Fig.1.

It is planned to offer various different information services (IS) through COIN, roughly grouped into organization (management), business (projects, business areas), research (core competencies), service (help), employee, and up-to-date (news, absence list, lunch plan) topics. Every user can subscribe to some IS: whereas some are specific to the roles (business area manager, project manager in project x) he plays in the organization, others are optional (competence areas like knowledge management). Each IS is realized by a query case base (initially one query case) and a content case base (provided content of the IS) (Fig.4). The cases are characterized by attributes describing the context (project, process) or by key words. This approach will be realized by a combination of structural and textual CBR using a commercial CBR tool (CBR-Works/orenge from empolis knowledge management GmbH, Germany; www.tecinno.de).

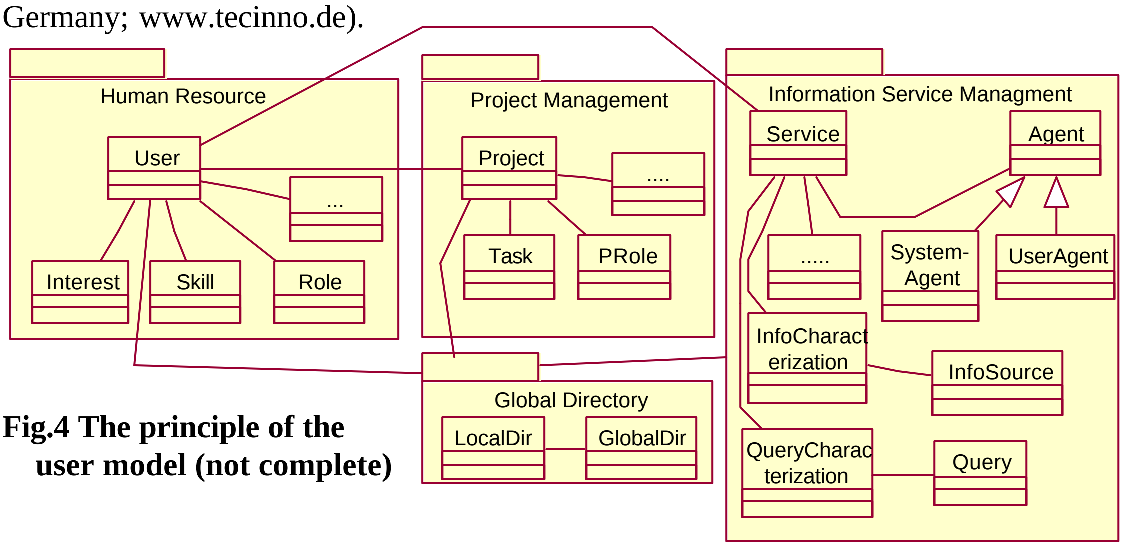

Fig.4 The principle of the user model (not complete)

The user model is partially kept within an extended human resource system (Fig.4), where each user is represented by his organizational (roles, skills) and individual (interest, navigation) data. Additional information comes from the projects the user is involved in. Thus the user model is physically distributed and has to be dynamically joined if required (e.g., change of context). In the words of CBR, an object user is treated as a case, which is used for case-based retrieval. Similarities are used for a better forecast of users needs according to information. Because users cannot know which information sources are available, it can be useful to use queries that have been used successfully by users with similar interests (Weibelzahl & Weber 1999). The evaluation of delivered information by the user is necessary for future evolution of the whole system. If the user agrees with the information, the value of the query that led to the content and the content itself are increased (the same will happen to successful reused experience packages mentioned before). In the case of rejection, this decreases at first only the value of the query, but it is also stored as a hint for maintenance. In particular, advanced users can pose a new query, which is then stored in the query case base.

To clarify the functionality we provide an example for an IS named “What is interesting to read”. For instance, a business area manager reads a study on knowledge management tools. She evaluates it as useful and offers it for this information service. The service asks her to characterize the study briefly. The characterization is needed for similarity based access to the study in future. She notes that it may be useful for business area managers and colleagues who are interested in knowledge management. If another business area manager logs into the COIN system later, (with respect to his role he has also subscribed this IS) the link to the study is offered automatically, because there is increased similarity due to the match in the role. If someone logs in who is interested in knowledge management, which is described as part of his individual user model, the link is also offered. Further on, if a person managing a knowledge management project, which is described in the project information, logs in, again an increased similarity would cause the link to be provided.

4. Outlook

Managers tend to find information more valuable when they are convinced of the reliability of the respective source (Traphöner 2001). Solving this problem is a major topic in the work on future knowledge management technology. We try to solve it by using CBR as the starting point, and by using an open environment that will be extended with other techniques as appropriate (e.g., from machine learning, knowledge management, etc.):

- Fostering the case base with information *and* sources, on the one hand, and evaluation points that provide information on successful application, on the other hand.
- Forecasting of user’s agreement with delivered information based on user’s history stored in the case base. If a similar person (i.e., role) within a similar context considers a similar information item useful, hopefully the addressee has the same opinion. For how to improve the similarity assessment based on a knowledge discovery approach, see Rech et al. (2001).

Currently we are working on virtual competence centers for software engineering. For these projects knowledge/experience repositories have to be constructed. The difference with respect to COIN is that in the beginning we do not know anything about the user. Therefore, we are developing a system – again based on CBR technology – to acquire informational requirements from the user. We intend to adapt the approach of Weibelzahl and Weber (1999) on "real estates" for "users".

5. References

Aamodt, A. and Plaza, E. (1994). Case-Based Reasoning: Foundational Issues, Methodological Variations and System Approaches. *AI Communications Vol. 7, No. 1*, 39-59

Althoff, K.-D. (1997). *Evaluating Case-Based Reasoning Systems: The INRECA Case Study*. Postdoctoral thesis (Habilitationsschrift), Department of Computer Science, University of Kaiserslautern

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.

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 Data Mining Conference*, Leipzig, 24.-25. July 2001, Institut für Bildverarbeitung und angewandte Informatik

Basili, V.R.; Caldiera, G.; Rombach, D.: Experience Factory; In Marciniak, J.J. ed., Encyclopedia of Software Engineering, vol 1, 469–476; John Wiley & Sons; 1994.

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. March 2001, (http://demolab.iese.fhg.de:8080/AK-Praktisches-Wissensmanagement/WM2001-WS5/).

Bauer, M., Gmytrasiewicz, P. J. & Vassileva, J. (eds.) (2001). *User Modeling 2001*. Springer Verlag, LNAI 2109

Bergmann, R. (1996). *Effizientes Problemlösen durch flexible Wiederverwendung von Fällen auf verschiedenen Abstraktionsebenen*. PhD Thesis, Department of Computer Scienece, University of Kaiserslautern

Bergmann, R. (2001). *Experience management - foundations, development methodology, and internet-based applications*. Postdoctoral thesis (Habilitationsschrift), Department of Computer Science, University of Kaiserslautern (submitted).

Decker, B.; Jedlitschka, A. (2001). The Integrated Corporate Information Network iCoiN: A Comprehensive, Web-based Experience Factory. In: K.-D. Althoff, R. Feldmann & W. Müller (eds.) , Advances in Learning Software Organizations, Springer Verlag

Fischer, G. & Ye, Yunwen (2001). Personalizing Delivered Information in a Software Reuse Environment. In Bauer et al. (2001) *User Modeling 2000*, 178-187

Jameson, A. (2001). *Personalization for E-Commerce*. Tutorial presented at UM 2001

Kluge, A.: Erfahrungsmanagement in lernenden Organisationen; Verl. für angewandte Psychologie; Göttingen 1999;

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

Nonaka, I.; Takeuchi, H.: The Knowledge-Creating Company – How Japanese Companies Create the Dynamics of Innovation. New York 1995.

Rech, J. et al. (2001). Using Knowledge Discovery Technology in Experience Management. Submitted to Workshop "Maschinelles Lernen (FGML01)", GI-Workshop-Woche „Lernen – Lehren – Wissen – Adaptivität (LLWA01)", Universität Dortmund, 8.-12. Oct. 2001

Stewart, Thomas A.: "Der vierte Produktionsfaktor: Wachstum und Wetbewerbsvorteile durch Wissensmanagement", München: Hanser 1998

Tautz, C.: Customizing Software Engineering Experience Management Systems to Organizational Needs; Ph. D. diss., Dept. of Computer Science, University of Kaiserslautern, Germany; 2000; Stuttgart: Fraunhofer IRB Verlag

Traphöner, R. (2001). Knowledge Sharing Medium. In *Proc. 5th International Conference on Business Information Science (WI 2001)*, University of Augsburg, 19.-21. Sep, 2001

Weibelzahl, S. & Weber, G. (1999). Benutzermodellierung von Kundenwünschen durch Fallbasiertes Schließen. In Proc. 7. GI-Workshop "Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen", LWA-1999, Otto-von-Guericke-Universität Magdeburg, Germany, 29.-30. Sep.1999, 295-300

Wilke, W. (1999). *Knowledge Management for Intelligent Sales Support in Electronic Commerce*. PhD Thesis, Department of Computer Science, University of Kaiserslautern