Fraunhofer Institut
Experimentelles
Software Engineering

Corporate Information Network (COIN): The Fraunhofer IESE Experience Factory

Authors:
Andreas Jedlitschka
Klaus-Dieter Althoff
Björn Decker
Susanne Hartkopf
Markus Nick

IESE-Report No. 034.01/E
Version 1.0
May 2001

A publication by Fraunhofer IESE

Fraunhofer IESE is an institute of the Fraunhofer Gesellschaft.
The institute transfers innovative software development techniques, methods and tools into industrial practice, assists companies in building software competencies customized to their needs, and helps them to establish a competetive market position.

Fraunhofer IESE is directed by
Prof. Dr. Dieter Rombach
Sauerwiesen 6
D-67661 Kaiserslautern

Abstract

Experience-based continuous learning is essential for improving products, processes, and technologies in emerging as well as 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. 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, an "Experience Factory" was established at Fraunhofer IESE, with the COrporate Information Network (COIN) initiative. 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").

Keywords: Experience Factory, Experience Base, learning organization, Community of Practice, lessons learned

Table of Contents

1 Introduction

In all emerging areas of business and engineering science, there is normally a lack of explicit knowledge about their underlying processes, products, and technologies. Usually, such knowledge is built up through individual learning from the experience of the people involved. The area of organizational learning, as one part of knowledge management, tries to increase the effectiveness of individual human learning for the whole organization. Besides improving internal communication (group learning) [12], organizational learning also includes documenting relevant knowledge and storing it (for reuse) in an organizational, corporate memory [1, 18]. The learning target for a learning organization is to enable its members to effectively handle situational requirements, taking past experience into account. Providing a higher number of alternative decisions or procedures to employees than they would have had based on their individual repertoire characterizes a learning organization [13].

An approach known from software engineering - called experience factory (EF) [5] - goes one step further. Knowledge (in the form of processes, products, and technologies) is enriched by explicitly documented experience (e.g., lessons that were learned during the practical application of the knowledge). The EF approach explicitly addresses capturing, documenting, storing, maintaining, evaluating, and disseminating such experience. These "experience packages" are stored in an experience base (EB), which is an organizational memory for relevant knowledge and experience. The EF approach tries to explicitly rebuild human “learning from experience” to further support organizational learning. The EF has to be supplemented on a technical system implementation level to realize the EB.

Within this paper we first introduce our IESE EF as an example for an operative EF (Sec. 2). After presenting the content framework (Sec. 3), we detail the acquisition of lessons learned (Sec. 4). Enhancing the presented approach, we introduce new strategies to capture, process, disseminate, and exchange knowledge (Sec. 5). Finally, a short summary is given (Sec. 6).

2 Corporate Information Network: The IESE Experience Factory

The EF deals with the typical problem that the major parts of business relevant experience of an organization resides in the brains of some few experts. With the rapid growth of our institute aggravates this problem, because these experts are rarely accessible due to their involvement in many different tasks. Therefore, this group of experts becomes a scarce resource as information and knowledge providers. Hence, it is important

- to provide the less experienced people with default processes and guidelines to "jump-start" them
- to facilitate experience sharing among "experts and newbies" to build up their expertise more quickly, and
- to avoid doing work/errors twice. Since the increasing size of our institute does not allow to talk to all people regularly in a short period of time (e.g., on a weekly basis), experience sharing on a personal basis does not work any more in such an environment.

Therefore, a project named COIN (Corporate Information Network) was launched. Beside brigding the above mentioned gap, COIN is used as a real project environment for the development and validation of technologies and methods for goal-oriented experience management. This includes knowledge elicitation, processing, dissemination, presentation, maintenance, and evaluation. COIN consists of three main parts: the EB, the COIN team, and an Intranet representation.

Within the EB included in COIN, all kinds of experience necessary for our daily business is stored (e.g., projects, business processes, document templates, lessons learned (LL) such as guidelines, observations, improvement suggestions, problems that occurred and problem fixes that were applied). Defined processes (e.g., structured interviews within project touch-down meetings, see Sec. 4) populate this EB systematically with experience typically needed by our project teams. Dedicated improvement processes analyze problems that have occurred, devise improvement actions to avoid their recurrence, and implement strategic decisions by the institute's leadership.

However, elicitation, distribution, and integration of process descriptions and LLs need an investment of effort [9]. The project teams using the process descriptions and gaining the experiences cannot be expected to invest this effort. Compared to the objectives of the organization, projects have a short-term perspective, focusing on the development goals of the project. Therefore, an organizational unit that is responsible for knowledge management is required. This organizational unit has to be separated from the project teams. According

to [5, 4], this separate organizational unit is called EF, which for the IESE is run by the COIN team [17].

Currently our solution focuses on two major subject areas: business process descriptions and LL (see Sec. 4). Besides this, many different kinds of experience like artifacts developed during projects are to be stored in the EB. Each is called an experience package. In addition, these experience packages are highly interrelated. For example, projects produce deliverables in the form of slide presentations and reports. Slide presentations may be summaries of reports. Observations and problems are gained during a project while a particular business process is performed, that is, we have to deal with context-sensitive experience. Such kind of experience is unique in the sense that the same context will not recur. Therefore, people will be searching for experience that has been gained in *similar* contexts. Both, the requirement for supporting different kinds of interrelated experience packages and the need for context-sensitive, similarity-based retrieval, demand a specialized technical infrastructure for the EB.

These are common requirements for an EB [17]. Our solution to meet these requirements is INTERESTS (Intelligent Retrieval and Storage System) [3]. It consists of a general purpose browser for accessing and presenting the EB contents using a standard web browser, an EB server synchronizing (and logging) access to the EB, and case-based reasoning (CBR) [2] as basic technology for retrieving, reusing, revising, and retaining the EB contents. For the actual EB a commercial tool (CBR-Works/orenge from tec:inno, Germany; e.g. [16]) is used. Each experience package is implemented as a "case" based on a structural CBR approach [7]. This includes a domain ontology for modeling the different types of case concepts, formal and informal case attributes together with the respective similarity measures, as well as relations between cases.

Within an experiment the benefits of this EB approach have already been demonstrated [17]. 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. Other applications not yet considered (e.g., human resource and educational systems) may provide valuable information, too. Additional information can lead to a more precise and better aggregation and adaptation of knowledge to users needs, but also requires the integration of the respective applications.

3 The Content Framework of the EF

The Content Framework (CF) presented in Figure 1 is a forward evolution of the knowledge management infrastructure set up within [17]. CF acts as a vision for a comprehensive management of experience within an organization, thus, representing a generic blueprint of an EB.

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

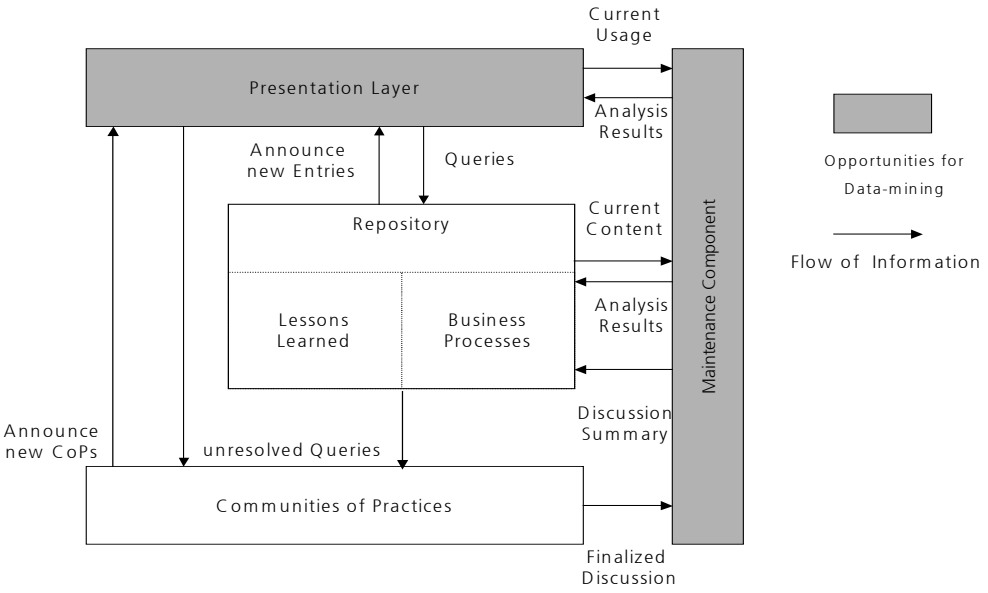

Figure 1 The EF Content Framework (CF)

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. 5).

The *Repository* contains the explicitly captured and consolidated experience of an organization. As mentioned earlier (Sec. 2) for COIN, a combination of business process descriptions and lessons learned was chosen as a starting point (Sec. 4). Further experience management activities can be set up on that base [10].

The *Communities of Practice* component is a forum for the members of an organization to discuss current problems, questions, and open issues (Sec 5).

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.

4 Capturing and Presenting Lessons Learned

Lessons learned (LL) at this time are mainly acquired using structured interviews (a) regularly for running projects and (b) at project touch-down meetings. This is shown in an exemplary fashion in Figure 2. Supported by a questionnaire, an experience engineer (COIN team member) interviews the project team about their experiences, which are collected and summarized in a project analysis report. The report has to be reviewed by the project team. Finally, the project analysis report is split by the experience engineer (EE) into reusable experiences, which are then stored in the EB

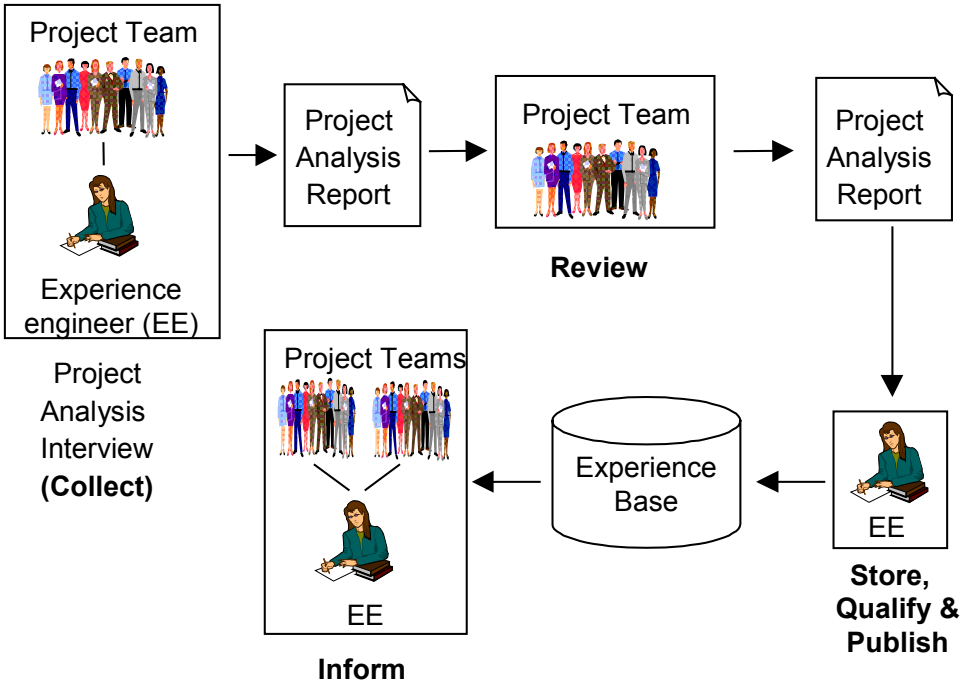

Figure 2 Experience acquisition with project analysis interviews

LLs can cover different topics and take on different forms [8]. Within the current EB we focus on LLs about project management. One LL can take on the form of an *observation*, a *problem*, *guideline*, *pragmatic solution*, or an *improvement suggestion*. Each LL states its origin to allow a querying IESE member to ask a colleague for further information. The context of the LLs is modeled by the two concepts "project" and "process": *"project"* is a characterization of the project in which the lesson learned was gained (e.g., project type, customers, person month, duration). *"Process"* names the business process and

thus the project phase in which the LL was gained. Therefore, a project team member can specify his current environment as well as the current situation to search the EB for similar experiences. Figure 3 shows the interrelations between the context and the different types of LL.

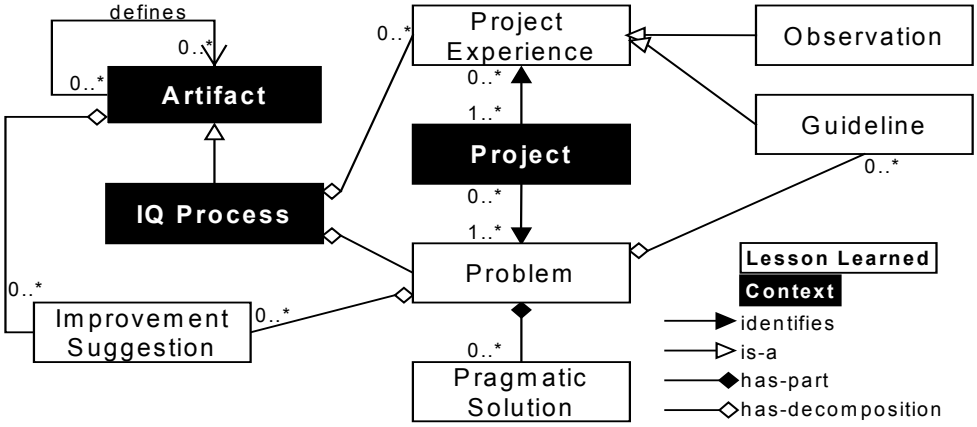

Figure 3 COIN Ontology according to [17]

Observations are facts that are of interest to future projects, often expressing some baseline (e.g., "it took 10% of the total effort to manage the project") or some positive effect (e.g., "the customer was happy because we provided him with a ready-to-use tutorial"). *Problems* are descriptions of negative situations that occurred during a project (e.g., "the expectations of the customer were not met"). Guidelines, improvement suggestions, and pragmatic solutions relate to one or more problems. *Guidelines* are recommendations on how a particular business process should be performed in order to avoid related problems. For example, a guideline could be the following: "Interact with the customer frequently, at least twice a month." An *improvement suggestion* is a proposal to change an artifact to avoid problems that occurred during its usage. *Pragmatic solutions* are sequences of immediate countermeasures taken by a project team in response to a recognized problem. While a guideline aims at preventing a problem from occurring in the first place, a pragmatic solution is applied after a problem has already occurred.

These project LLs (a) complement process execution differently and (b) are integrated into the process descriptions in several ways. Two examples should illustrate this relation: Observations can be used to build mental models or validate assumptions about project work (e.g., customer preferences). Consolidated guidelines can be integrated into process descriptions.

5 New Strategies to Capture, Process, Disseminate, and Exchange Knowledge

Knowledge has actually been identified as the "fourth factor of production"[1]. Therefore, unstructured, not personalized flooding with information can be counterproductive for building up and exchanging knowledge. 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 concept for aggregation and adaptation of information to users' context and needs.

5.1 "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 Figure 1

With his login in combination with stored user data (organizational role, project roles, skills, and interests) and a chosen view (concrete project), the user provides the actual context, for example: "role: developer; project: x; task: code testing" (the task is determined from the project plan). The given context is used to deliver knowledge collected within similar contexts without an explicit user query ("push" of information). The user can ignore this but, hopefully, he will at least evaluate the utility of the delivered information within his actual context. The evaluation is used to "educate" or "edge" agents for users' business and personal information needs. Personal needs can be context-sensitive 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 or directly after his login on the physical presentation layer. 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 Figure 4).

[1] Besides work, capital, raw material

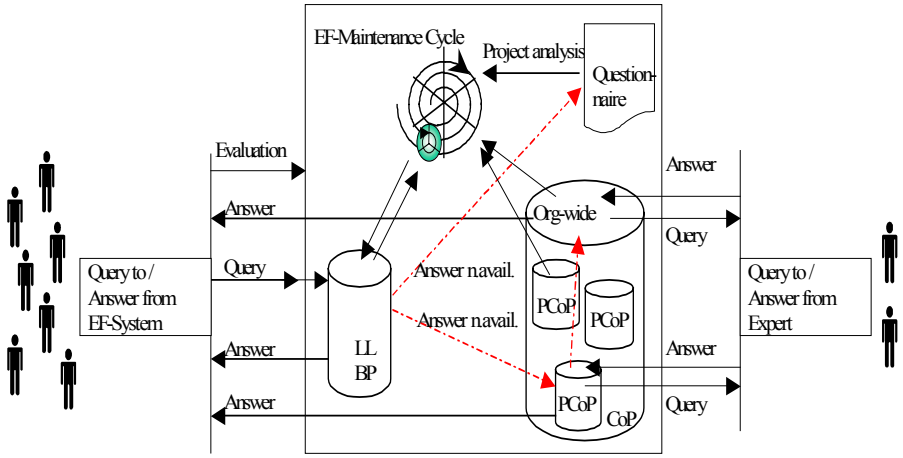

Figure 4 Correlation: Experience base (EB) and communities of practice (CoP)

5.2 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 experience package 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 about the gained experience. According to [15] 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). The project members tell their experiences within those interviews 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 to put them into the EB (Sec. 4). 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 [13].

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 Figure 4). 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.

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. Therefore 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 (KU), 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 evolution 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, 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 [6]. 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 [14]. With every new input to the EB, existing experience packages can be confirmed or questioned. This work will be supported by the introduction of the utility evaluation by the users. The EF Maintenance cycle mentioned in Figure 4 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, in-

formation regarded as valuable is forwarded to the EF Maintenance Cycle for final inspection before it is stored in the EB.

5.3 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. 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 mostly cannot 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 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 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.

The technical realization resies within the representation layer of Figure 1

Further research work has to be done here.

6 Outlook

In the future, we do not want to burden users with overhead for searching information or asking for experience. We are looking towards a solution for a single point of access 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 according role the employee plays, and (c) the owner of a piece of information. The knowledge and information offered shall be aggregated and adapted based on experiences to users' actual needs, determined by the actual context. New tools like the communities of practice will help to explicate more valuable knowledge and information residing tacitly in experts' heads. The use of data mining methods will lead to a better understanding of correlations between knowledge, information, and data stored within the organization wide system. Our vision is to provide users with valuable information and knowledge at the right time, in an adequate representation, and within the actual context ("just-in-time"). Further research work still to be done includes information adaptation and aggregation, complete instantiation of the current Fraunhofer Institute for Experimental Software Engineering corporate information network architecture as well as an approach for its (continuous) improvement.

References

[1] Abecker, A.; Bernardi, A.; Hinkelmann, K..; Kühn, O.; Sintek, M.: Towards a Technology for Organizational Memories, IEEE Intelligent Systems 1998.

[2] Althoff, K.-D. (2001). Case-Based Reasoning. To appear in: S.K. Chang (Ed.), Handbook on Software Engineering and Knowledge Engineering. Vol.1, World Scientific (40 pages).

[3] 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.

[4] Althoff, K.-D.; Bomarius, F.; Tautz, C.: Knowledge Management for Building Learning Software Organizations, Information Systems Frontiers 2:3/4, 349-367, 2000; Kluwer Academic Publishers

[5] 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.

[6] 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/).

[7] 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.

[8] Birk, A.; Tautz, C.: 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.

[9] Decker, B.; Althoff, K.-D.; Nick, M.; Tautz, C.: Integrating Business Process Descriptions and Lessons Learned with an Experience Factory; In Professionelles Wissensmanagement - Erfahrungen und Visionen; Aachen 2001; Shaker Verlag S.54-58

[10] Decker, B.; Jedlitschka, A.: The Integrated Corporate Information Network iCoiN: A Comprehensive, Web-based Experience Factory; IESE-Report 031.01/E; Fraunhofer IESE; Kaiserslautern; 2001

[11] Gordon, T.F.; Voss, A.; Richter, G.; Märker, O.: Zeno: Groupeware for Discourses on the Internet; in Künstliche Intelligenz, Heft 2/01, Seiten 43-45, ISSN 0933-1875, arendtap Verlag Bremen, (2001)

[12] Jedlitschka, A.; Althoff, K.-D.; Decker, B.; Hartkopf, S.; Nick, M.; Corporate Information Network (COIN): The Fraunhofer IESE Experince Factory. In Proc. 4th

International Conference on Case-Based Reasoning (ICCBR'01) Workshop on Process Oriented Knowledge Management

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

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

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

[16] Schulz, S. 1999. CBR-Works - a state-of-the-art shell for case-based application building. In 7th German Workshop on Case-Based Reasoning, Wuerzburg, Germany.

[17] 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

[18] van Heijst, G., van der Speck, R., and Kruizinga, E. 1996. Organizing corporate memories. In Proceedings of the Tenth Banff Knowledge Acquisition for Knowledge-Based Systems Workshop, SRDG Publications, Department of Computer Science, University of Calgary, Calgary, Alberta, Canada T2N

Document Information

Title:	Corporate Information Network (COIN): The Fraunhofer IESE Experience Factory
Date:	May 2001
Report:	IESE-034.01/E
Status:	Final
Distribution:	Public