

Using Weblogs for Eliciting New Experiences and Creating Learning Elements for Experienced-based Information Systems

Gabriela Avram

(Fraunhofer IESE, Kaiserslautern, Germany
gabriela.avram@iese.fraunhofer.de)

Eric Ras

(Fraunhofer IESE, Kaiserslautern, Germany
eric.ras@iese.fraunhofer.de)

Stephan Weibelzahl

(Fraunhofer IESE, Kaiserslautern, Germany
stephan.weibelzahl@iese.fraunhofer.de)

Abstract: Experience-based Information Systems (EbIS) become more and more sophisticated from a technical point of view. However, there are several shortcomings that seem to limit the input and usage of these systems. The problems identified in this paper are related to knowledge acquisition, to learning issues, as well as to the users' motivation and trust. We introduce an approach to both enhance the content of the experience base (EB) and to improve learning from experiences at the same time: weblogs maintained during daily work can serve as input for experience bases and learning element bases (LEB). Retrieved experiences are then enriched with additional learning elements in micro-didactical learning arrangements. An empirical study explores the suitability of using weblogs for such an approach.

Keywords: Experience-based Information System, wiki, weblog, micro-didactical arrangement
Categories: A.1, H.4, J.4, K.3

1 Introduction

Especially in the Information and Communication Technologies sector, continuous competence development is essential to beat the increasing flood of information, the fast ageing of knowledge, the continuous change of requirements for problem understanding and solving. Hence, these facts require lifelong learning to remain competitive in the information society.

One of the domains where knowledge based systems (KBSs) were profitably implemented is Software Engineering, a quickly changing, knowledge-intensive business involving many people working in different phases and activities. Though, organisations encounter problems identifying the content, location and use of knowledge [Rus, 02].

Other problems are related to the acquisition of new knowledge, to the fact that existing knowledge is not suitable for learning purposes, and that maintenance and evaluation of the already documented knowledge are not done appropriately.

In this paper, we focus on the possible use of personal and workgroup content management systems, such as wikis and weblogs, for capturing basic information, knowledge and experiences [Angeles, 03]. The input from wikis and weblogs can be used for enriching the experience base (EB) and for producing learning material. A *weblog* is a website which contains periodic, reverse chronologically ordered posts in a common webspace. A *wiki* is a special type of hypertext document enabling collective authoring in a simple markup language using a web browser. While “*knowledge* is the range of learned information or understanding of a human or intelligent information system, *experience* is considered to be knowledge or practical wisdom gained through human senses, from directly observing, encountering, or undergoing things during the participation in events or in a particular activity” [Tautz, 2000].

By combining experiences with learning elements, as suggested in this paper, the advantages are twofold: first, the applicability of experience packages increases by adding learning elements to the package; second, the application of the experience and the new gained knowledge in practice deepens the learning effect and improves the applicability in different situations.

2 General Shortcomings and Problems to Be Solved

This section is describing two specific situations related to the update and the use of experiences included in an EB: the elicitation of new experiences and learning based on these experiences.

The field of Experience Management (EM), as a sub-field of knowledge management (KM), is aiming at supporting the management and transfer of relevant experiences. The software system used for managing, storing, retrieving and disseminating these experiences is called an Experience-based Information System (EbIS). EbISs are a special type of KBS that build their intelligent features on the explicit representation of knowledge or experience [Tautz, 00].

One of the most difficult problems for maintaining an EbIS is to elicit and document new experiences. Knowledge should be captured immediately after the new experience’s occurrence. Then, the new experience has to be described in a structured way, and the record must also include information on the context of the specific experience. This is an activity that requires a lot of effort and dedicated skills [Hellström, 00].

Considering the deployment process of KBSs, learning is considered to be a fundamental part of KM, since employees must internalise (learn) shared knowledge before they can use it to perform specific tasks [Rus, 02]. It was assumed that KBSs could solve the problem of continuous competence development by providing intelligent retrieval mechanisms and innovative presentations techniques. Hence KBSs focus mainly on knowledge, (i.e., the product of learning processes), and less on the learning processes themselves and on the needs of individuals.

A detailed analysis of the shortcomings of current EbISs based on the literature, as well as on our own experience, led to a variety of problems that are either related to the quality of the EB content, learning issues, or to psychological factors like motivation and trust.

2.1 Problems Related to the Content of the Experience Base

The quality of the EB content is obviously a crucial point for various reasons. First of all, the elicitation of knowledge from experts and users might cause problems, because new experiences must be first identified in order to achieve a decent coverage. If there are no regular triggers to reflect the recent experiences, important insights might get lost. People must decide to report what might be useful for others and find the appropriate granularity level.

Second, in most systems the elicitation procedure imposes some barriers to providing a given experience. In fact, there is a trade-off of structured report vs. flexibility: the more formalised and standardised the way the experiences are reported, the better the retrieval, but the lower the ability to suite the different preferences of those who are supposed to share these experiences.

Third, the quality of the reported experience depends highly on the individual communication skills of the contributor, e.g., the ability to structure the content, to formulate precisely and to adapt to the possible audience. Further, the lack of ability in coping with the ambiguity of experiences could prevent both the contribution to the EB or their use afterwards.

Fourth, the maintenance of the EB requires regular evaluation of all experience packages and removal of outdated entries. However, with the growing size of the EB, it is difficult to keep an overview in order to connect related packages and to avoid inconsistencies.

2.2 Learning Related Problems

Even if the EB has a high coverage and precision, the packages might still be inappropriate for learning due to two reasons: the quality of the learning content and the fact that the learning processes are ignored by the EbIS. The experience package usually does not contain information about the required competence level for applying a specific experience into practice. Often users need additional information about the subject domain, because experts provided the experience without giving extensive explanations of the background and because the users lack knowledge of domain concepts. However, creating such an additional learning content is a cost intensive task.

2.3 Motivation and Trust Related Problems

The users need to be motivated for both using and contributing to the EB. Sharing knowledge is sometimes perceived as dangerous if competitors could use the shared knowledge. Potential contributors might have concerns of supplying the others with knowledge without any profit. Moreover, both contributing to and consulting the EB can be seen as time and effort consuming activities. However, incentives seem to help less than the statutory obligation to provide feedback both during and at the end of each project.

In fact, nobody wants to be a pioneer in sharing knowledge, but everybody would like to start by re-using somebody else's experience. Thus, it is important to facilitate the initial usage of the system by providing a sufficient fundament of relevant experience packages.

Finally, users might also refuse to apply an experience package, because of a lack of confidence in its validity. It might be unclear whether a packaged experience is still up-to-date or whether applying it might involve a certain risk. It is also possible that the relationships to other experiences were not appropriately captured. The quality of packaged experiences could also be influenced by contradictory interests: employees are much more inclined to share positive experiences, because bringing to attention their failures could have a negative connotation. The “not invented here” syndrome can be another cause of reluctance in reusing experiences, and to overcome it, there must be a shift in the culture of the organisation [Rus, 02]. Particularly for global organisations, work-related cultural differences related to power, individualism, or gender could also have a negative impact.

3 Possible Solutions

Through our approach, we will try to address some of these problems. The capture of knowledge from weblogs will give full flexibility to the contributors, and the content has to be detailed later on and structured by a specialised team. This will give people the opportunity to capture new knowledge immediately.

The EM approach is typical for centralised KBSs. At the other end, the alternative solution is offered by the distributed KBSs, focusing more on people interaction.

Our approach will also provide an alternative to the creation of courseware content starting from packaged experiences. In courseware development, we will use examples, definitions, and cases from weblogs, and we will employ pedagogical agents for creating knowledge packages with the appropriate content.

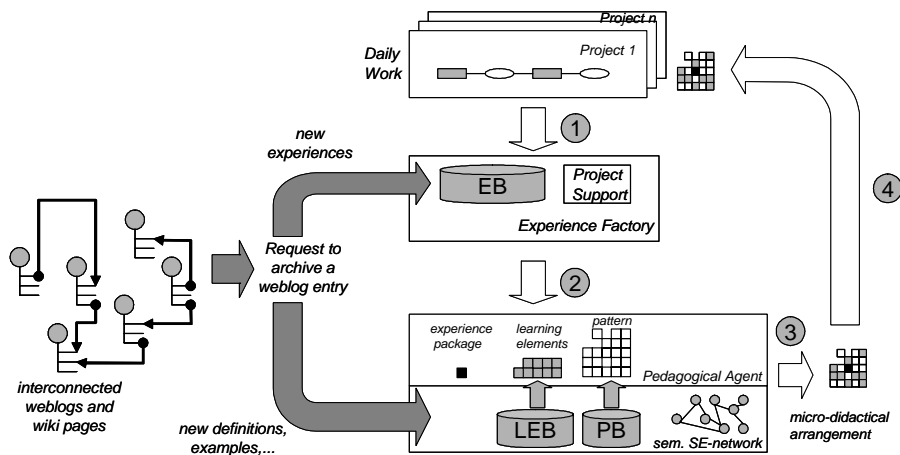


Figure 1: Overview of the approach using weblogs as input for the experience base (EB) and the learning element base (LEB)

Last, but not least, such an approach will help to create an atmosphere of trust by encouraging people to a reciprocal change of experiences that could lead them to recognised expert positions.

We claim that the use of weblogs for recording knowledge will supply new experiences to be added to the EB and to be used in providing content for new learning elements to be added to the LEB.

3.1 Experience Elicitation Related Solutions

A possible scenario for addressing some of these problems is:

- complementing the EB, which is a centralised experience repository, with distributed repositories—belonging to individuals work groups;
- through regular observation, the re-usable experiences posted in weblogs could be identified; they could be included in the EB after being documented and restructured;
- the same experiences (or others) could be identified as suitable for learning purposes.

Of course, the usage of very simple and affordable software tools for sharing knowledge available to any employee at any moment in time could overcome some of both the technical and the psychological barriers to sharing knowledge. However, the practice showed that even if this category of open editing tools—such as weblogs and wikis—are easy-to-use and very user-friendly, their sole availability is not sufficient for changing the employees' habits and for encouraging them to plainly share knowledge. In order to support such a type of changes, we tried to define a special role: that of a *facilitator* who should work permanently on coaching and supporting the employees to adapt to the new situation. This person has to encourage permanently the sharing of experiences, by providing examples, success stories and access to technical support. The facilitator is supposed to signal the valuable contributions and to send them to the EB or to the LEB. After a while, it is possible that employees themselves will be able to detect them and will send requests to the appropriate team for including these new experiences either in the EB, or in the LEB.

3.2 Learning Related Solutions

By using an EbIS, informal learning takes place when users access and apply experience packages to solve problems during daily work. The learning effect relies a great deal upon the experience package quality, i.e., its description, presentation and applicability in a work context. Low quality packages are difficult or impossible to use and could have a low or inexistent learning effect.

The approach that we currently develop focuses on the combination of plain experiences and learning elements (LE). LEs could be term definitions, application examples, questionnaires, explanations etc. The approach aims at improving the suitability of the delivered input for learning by embedding the experiences in so-called *micro-didactical arrangements*, i.e., complete learning units with an experience as central element. While usually the experience packages in EbISs are just retrieved and presented without any further explanation, we propose to augment the presentation with additional learning elements to facilitate a better understanding and applicability of the experience packages.

After the occurrence of a problem during daily work (see step 1 in Figure 1), the users search for suitable experiences in the repository to solve the problem. The retrieved and selected experience is forwarded to a pedagogical agent (see step 2). The pedagogical agent selects an appropriate pattern from the pattern base (PB see figure 1) according to the type of experience, i.e., technology, product or product package, and the role of the user within the software engineering process. A pattern defines the structure, and the number and type of LEs that should be combined with the selected experience. The pedagogical agent searches LEs in the learning element base (LEB) by using a semantic network that describes the Software Engineering domain. Then, the agent produces an instance of the pattern by inserting the experience and the LEs into the pattern (see step 3). The produced micro-didactical arrangement is provided to the user, who employs it for solving the problem. This is done by acquiring knowledge about the experience domain in small steps.

To illustrate this new approach, we introduce an example from the software engineering domain. A software project manager might find that the rework effort to fix errors that have been made in early development phases is too high. An EbIS provides a possible solution for this problem by presenting the following experience package: "By introducing perspective-based reading (PBR) in our company we were able to reduce the rework effort in our projects by about 30%." We suggest to augment the presentation of this experience by creating a micro-didactical arrangement that supports the learner in using the information. Such an arrangement is based on Merrill's instruction principles [Merrill, 00]: The micro-didactical arrangement should *activate* the learners' existing knowledge by relating PBR to the software development process, i.e., it could tell the learners that PBR is an inspection technique, and is thus used for verification. It should also *demonstrate* and illustrate the knowledge by offering a more detailed description of PBR, as well as providing a process description on how to perform PBR and giving examples. The new acquired knowledge could even be *applied* in exercises and tests. Finally the learners should be encouraged to *integrate* the knowledge and apply it in order to solve the initial daily work problem.

In summary, the micro-didactical arrangement is a set of learning elements organized around an experience package to support self-organised learning.

4 Empirical Studies

We evaluated whether informal knowledge sharing through weblogs and wikis is able to provide content for LEs, and hence reduce the effort for content producing, by performing two empirical studies. These studies involved the monitoring of ten external weblogs, and the fostering of a Content Management System (CMS) including weblog and wiki facilities in a company Intranet.

For the analysis of entries from already existing external weblogs, ten such weblogs focusing on software engineering, knowledge management and e-learning were selected. For two months, their posts were monitored using news aggregators. We retained and re-structured posts with content that was considered meaningful for software engineers destined to be included in the EB or in the LEB. Three experts with different backgrounds were involved in this activity.

The case study directed at using weblogs and wikis for collecting experiences in a company Intranet was also planned and launched. The PHP-based CMS selected for this study was TikiWiki (<http://tikiwiki.org>).

The employees were initially introduced to open editing software allowing rapid, flexible and “on-the-spot” acquisition of knowledge (wikis, weblogs). They were encouraged to start and maintain personal weblogs for a two-month period and to post all the experiences they consider meaningful for their work. Five employees started personal weblogs, and nine community and project weblogs were created (Table 1). For organising and structuring work-related content, wiki pages were recommended.

The content of four of these weblogs was also monitored during the two months period. The participants were provided with coaching and on-line support. A weekly newsletter was issued in order to provide information and to maintain their interest in knowledge sharing. Aiming at encouraging knowledge exchange, the monitoring experts also acted as knowledge brokers for connecting the participants who shared the same interests. Meaningful content that could be used for documenting and including a new experience in the EB, or adding it to the LEB, was identified.

Weblogs and posts statistics: 114 out of 238 posts coming from the ten external and four Intranet monitored weblogs were retained as serving for our purposes.

The retained posts were counted using a) the classification of Meder [Meder, 2000], and b) a SE-specific classification. The summary showed that these 114 selected posts contained (a) 62 definitions, 90 descriptions, 170 references, and 41 examples. We also found the descriptions (b) of 5 products, 5 processes, 12 techniques and 14 tools.

	No of users	No of wiki pages	No of weblogs	No of posts
After 1 month	26	66	13	181
After 2 months	26	72	14	215

Table 1: TikiWiki usage statistics

The questionnaire-based survey organised after the two months period obtained feedback from eight users and included 19 questions grouped in five categories (content, learning and motivation-related problems, usability and possible developments). The interviews took on average 15 minutes. The results are summarised here:

The TikiWiki environment had two categories of users: seven active contributors and 19 passive readers. Because the content was generated without providing any guidelines, it was extremely various and it contained experiences, success and failure descriptions, research ideas, hints, tips and tricks, various links, references and comments. Some other content, related or not to Software Engineering, was also generated: conference announcements, hints for current and future PhD students, food recipes, and baby feeding hints. Because the TikiWiki environment was only accessible from inside the company, there was no concern regarding trust or intellectual property problems. Contributors were more motivated by the reuse of knowledge for personal advantage than by sharing or adding new experiences to the experience base. A possible automation of knowledge discovery was also discussed. The opinions ranged from “definitely not possible” to “partial automation could be

possible, but the decision on including a post, or a page content partially or as a whole in the EB or LEB should be made by humans”.

We will try to illustrate these findings with two examples. The first example is about a lesson learned that was included in the EB: A specific approach description for Test-First Programming was located in the *Mistaeks I Hav Made* weblog (http://nat.truemesh.com/archives/cat_testfirst_programming.html) written by Nat Pryce. “The problem was that I had written tests for each method, testing pre- and post-conditions and class invariants. This is the wrong approach to writing programmer tests. Instead, each test case should specify a describable aspect of the functionality an object provides to its clients. That aspect will probably involve multiple methods of the class. As a maintenance programmer using the tests as documentation you want to know how the behaviour of those methods is interrelated, not how each method acts individually.”

This post had to be re-structured in order to be included in the EB (Figure 2).

Subject	Test-First Programming
ID	1234
Topic Areas	SW-Technologies
Validity	0
Origin	#Nat Pryce (http://nat.truemesh.com/archives/cat_testfirst_programming.html)
Problem	The problem was that I had written tests for each method, testing pre- and post-conditions and class invariants.
Cause	
Situation	This is the wrong approach to writing programmer tests.
Reaction	Instead, each test case should specify a describable aspect of the functionality an object provides to its clients. That aspect will probably involve multiple methods of the class. As a maintenance programmer using the tests as documentation you want to know how the behaviour of those methods is interrelated, not how each method acts individually.
Prevention	
Current reusers	
Previous reusers	
Log	

Figure 2: Experience extracted from weblog post and prepared for inclusion in the experience base

The post also contained a reference to the definition of Test-Driven Programming from the C2 wiki (<http://c2.com/cgi/wiki/>), dedicated mostly to eXtreme Programming. The definition was also re-used as a learning element (Figure 3).

Test Driven Programming

Extreme Programming supports the use of tests as a development tool. Given an object, the developers devise tests for all interesting methods even before programming them. This means that:

- they are forced to define precisely what a method does
- they know where to begin writing a method
- they know when you are done writing a method
- they know the minimal scaffolding needed to run a method
- like scientists, they target reproducible results.

This means that they will recognize dependencies among the objects early, and they will work to minimize them.

Figure 3: Learning element extracted from C2 wiki

The second example is about some learning elements identified in a post from *Bliki* (<http://martinfowler.com/bliki/DomainSpecificLanguage.html>), *Martin Fowler's weblog & wiki*. It included a description of domain specific programming languages, directly reused as a learning element. Moreover, we were able to extract from the post some pros and cons for this type of languages to create another learning element. Certainly the extraction cannot be done in an automatic way, but weblogs seems to motivate people to make their knowledge explicit, which can serve as starting point for knowledge elicitation. The extracted learning elements could be part of a "micro-didactical arrangement" supporting the comprehension of an experience that discusses the choice of a programming language for a certain project or that reports a lesson learned of using a certain language for a specific domain.

While only weblogs were specifically monitored during our study, they frequently contained references to wiki pages. Usually, weblog posts contain references, but also definitions, theories, strategies that are easier to retrieve when they are afterwards organised in wiki pages.

5 Conclusions and Future Work

In summary, we identified several problems originating in a technical focus and in a disregard or even ignorance of human issues in socio-technical systems [Clegg, 00]. A system that is designed to manage, retrieve and present experience data, but does not fulfil the organisational, social, and psychological requirements will not get accepted by the users and finally, due to a lack of input, will be abandoned.

In this paper, we propose two solutions to overcome at least some of the problems that are related to knowledge elicitation and learning using EbISs. First, by integrating weblogs into the knowledge acquisition phase, we aim at achieving a more up-to-date condition of the EB. The knowledge sharing and elicitation is integrated into daily work, rather than separated in a formal process. Our studies try to demonstrate that the reported experiences and learning elements are nevertheless rich, important and useful. Second, we propose to support the usage of experiences by enriching them with learning elements in so-called micro-didactical arrangements. The users get

additional information on the subject to enhance comprehensibility and thus the learning effect. An empirical study demonstrated that blogs can actually be used as input for lessons learned and learning elements.

In the future, we plan to explore in more detail, on the one hand, the quality of experiences that can be found in weblogs and wikis, and on the other hand, the impact and effects of introducing new knowledge sharing methods such as weblogs and wikis in organisations.

Acknowledgements

Dr. Gabriela Avram's work was carried out during the tenure of an ERCIM fellowship.

References

[Angeles, 03] M. Angeles, Making sense of weblogs in the intranet, Presentation, Usability Professionals Association, 16 September 2003, <http://studioid.com>

[Clegg, 00] Clegg CW, "Sociotechnical principles for system design", *Applied Ergonomics*, 31, pp. 463-477, 2000

[Hellström, 00] T. Hellström, U. Malmquist, J. Mikaelsson, Decentralizing Knowledge – Managing Knowledge Work in a Software Engineering Firm, VRR-00-26, published in the Journal of High Technology Management Research, vol. 2, nr 3, http://www.viktoria.se/results/result_files/131.pdf

[Meder, 00] N. Meder. Didaktische Ontologien. In H. P. Ohly, G. Rahmstorf and A. Sigel, Eds., Globalisierung und Wissensorganisation: Neue Aspekte für Wissen, Wissenschaft und Informationssysteme. Vol 6: Fortschritte in der Wissensorganisation, Ergon Verlag, Würzburg, 2000

[Merill, 00] M.D. Merrill, First principles of instruction. Paper presented at the international conference of the Association for Educational Communications and Technology (AECT), Denver, 2000

[Rus, 02] I. Rus, M. Lindvall, Knowledge Management in Software Engineering, IEEE Software, May/June 2002, 26-38

[Tautz, 00] C. Tautz, Customizing Software Engineering Experience Management Systems to Organizational Needs, PhD thesis, University of Kaiserslautern, Germany, 2000. Published in 2001 by Fraunhofer IRB Verlag