Introduction Strategy and Feedback from an Experience Management Project

Mirjam Minor

Institut für Informatik, Humboldt-Universität, Berlin
`minor@informatik.hu-berlin.de`

Abstract. The ExperienceBook is case-based system to support Experience Management activities in organizations. In this article, we will describe lessons learned from an employment of the ExperienceBook II within a discussion forum for students. This application has a been in use at our university's institute for informatics from October 2003 until January 2004 and in a second turn since October 2004. The article focusses rather on organisational aspects than on the technical details of the used concepts. Success factors and drawbacks are discussed. The article gives the basic functions of the system, presents the system introduction and motivation strategy, explains the feedback results, and discusses some conclusions and related work.

1   Introduction

Experience Management (EM) deals with the experiential knowledge of an organisation, i.e. the experiences of the people that are mainly stored in their heads and are used for solving current problems. EM is a special form of knowledge management that is restricted to task-based knowledge (see [Bergmann, 2002]). The main processes of an EM system handle the following four issues (compare also with [Minor, 2001]): making knowledge explicit, storing the knowledge, making it available in a task-based context, and keeping the knowledge up-to-date. It is crucial for the success of an EM system to let the users participate in all developing phases of the system: before the system is started, during the first weeks of usage, and after the system has been established or failed.

The ExperienceBook II is an EM application that provides a discussion forum and a shared repository for experiential knowledge. The repository is a case base consisting of textual cases with descriptions of the students' daily problems and their solutions. This includes computer science related problems like hints how to use a certain software but also very general issues like a list of the best pubs on the campus. The ExperienceBook has an intelligent search mechanism for the case data that computes the best matching cases concerning a query. An advertising strategy aims to motivate the students to query the system, write new cases, and comment existing cases. So, experiential knowledge that is hidden in the heads shall be transformed into explicit knowledge.

2 Basic Design of the System

Before we will discuss the organisational aspects of the ExperienceBook, we will give a quick introduction to the basic design of the system. It has a client-server architecture with a central server holding the case data in a Case Retrieval Net [Lenz and Burkhard, 1996]. The server waits for requests from CGI clients. The Web interface allows easy access from different operating systems via html browsers; the server queue handles multi-user access.

Queries consist of a text in natural language that can be enriched with an attribute value for *SUBDOMAIN* representing the application area of the requested cases. The real application deals with German texts, but we have translated the examples into English. A sample query in the subdomain UNIX is 'How can I list the content of a directory?'. For the users, the cases are semi-structured texts that are enriched with attribute-value pairs. The left hand side of Table 1 contains a sample case in XML, Figure 3 shows the same case as it is displayed on the graphical user interphase.

Table 1. Sample case in both representation forms: in XML and as set of information entities

<CASE> **<CASE_NUMBER>** 12 **</CASE_NUMBER>**	**CASE_IDENTIFIER** = 12
<RETRIEVAL_ATTRIBUTES> SUBDOMAIN=UNIX **</RETRIEVAL_ATTRIBUTES>**	**IE_LIST_RATTR** = [(SUBDOMAIN, UNIX)]
<INFO_ATTRIBUTES> author = M. Minor email = minor@informatik.hu-berlin.de time_stamp = 1064844043 last_edited_at = 09-29-2003 revision = 1 **</INFO_ATTRIBUTES>**	
<DESCRIPTION> How to send an email from the UNIX pool? **</DESCRIPTION>**	**IE_LIST_DESCRIPTION** = [SEND, EMAIL, UNIX POOL]
<SOLUTION> mail <mail_address> from command line sends the following text to <mail_address>, end of input with Ctrl-D **</SOLUTION>**	**IE_LIST_SOLUTION** = [MAIL, COMMAND LINE, SEND, TEXT, END, INPUT, CTRL-D]
<COMMENTS> send the content of a file as email, e.g. myLetter: mail hdb@informatik.hu-berlin.de < myLetter.txt use a mail tool like elm or pine **</COMMENTS>** **</CASE>**	**IE_LIST_COMMENTS** = [SEND, CONTENT, FILE, EMAIL, MAIL, MAIL TOOL, ELM, PINE]

Das ExperienceBook II

Ein fallbasiertes Wissensmanagement–System
zum Austausch von Erfahrungswissen der Studierenden der Informatik

Das ExperienceBook sucht zu Ihrer Anfrage die besten Fälle aus der Fallbasis heraus.
Bitte geben Sie eine Anfrage ein (z.B. "SWI–Prolog unter UNIX starten"):

```
Wie kann ich mir ein Verzeichnis
ansehen?
```

Anfrage abschicken

Möchten Sie Ihre Anfrage durch Auswahl des **Problemgebiets** präzisieren?

Gebiet:

UNIX–Probleme

neuen Fall eingeben | ganze Fallbasis | ...und so funktioniert's

M. Minor (minor@informatik.hu–berlin.de)
Erstellt am 21. Mai 2003, zuletzt geändert am 28. Oktober 2003.

Fig. 1. Query page of the ExperienceBook II

The meaningful concepts are extracted automatically from the textual sections following the method of [Lenz *et al*., 1998]: A text parser uses dictionaries with important technical terms of the domain and general language terms. The dictionaries contain also different spellings, grammatical forms, and abbreviations of a concept. After the parsing process, a case has a set of information entities for each section (see right hand side of Table 1). The queries are parsed in the same way.

The similarity of a query and the particular cases of the case base is computed by means of a composite similarity function *SIM*. It compares the query with the *DESCRIPTION* and the *RETRIEVAL_ATTRIBUTES* section of the case by means of the extracted sets of information entities of both. *SIM* uses as well a dictionary of local similarity values between information entities, e.g. sim(__LINUX__, __UNIX__) = 0.5 or sim(__ASCII PRINTER__, __POSTSCRIPT PRINTER__) = 0.3. The similarity of a query and a case is a weighted sum of local similarity values:

$$SIM\,(Query, Case) = \sum_{e_i \in Query} \sum_{e_j \in Case} sim(e_i, e_j).$$

Ergebnis des Retrieval–Prozesses:

Bitte klicken Sie auf die Fallnummer, wenn Sie einen Fall ansehen möchten.

Fall–Nr.	Datum	Aktivierung	Begriff
4.1	29. 09. 2003	1.000000	Verzeichnis mit ls anzeigen lassen: Wie kann man unter UNIX...
5.1	29. 09. 2003	0.666667	neues Verzeichnis mit mkdir erstellen: Wie erzeuge ich unte...
6.1	29. 09. 2003	0.666667	cd–Befehl: Verzeichnis wechseln: Wie wechsle ich unter UNIX...
7.1	29. 09. 2003	0.666667	Inhalt einer Datei ansehen mit more: Wie kann ich mir den I...
9.1	29. 09. 2003	0.666667	UNIX–Prozesse ansehen: Wie kann ich herausbekommen, welche ...
3.1	29. 09. 2003	0.333333	Manual lesen: Wie bekomme ich Informationen zu einem UNIX–B...
8.1	29. 09. 2003	0.333333	Beenden einer UNIX–Shell: Wie beende ich eine UNIX–Shell?...
10.1	29. 09. 2003	0.333333	Befehle durch eine Pipe verketten: Wie funktioniert "|" ?...
11.1	29. 09. 2003	0.333333	UNIX–Prozess killen: Wie kriege ich (z.B. einen abgestuerzt...
12.1	29. 09. 2003	0.333333	Mail abschicken: Wie schicke ich vom UNIX–Pool aus eine Mai...
13.1	29. 09. 2003	0.333333	Umlenkungsoperator < benutzen: Wie lenke ich die Standardei...
14.1	29. 09. 2003	0.333333	Umlenkungsoperator > benutzen: Wie lenke ich die Standardau...
15.1	29. 09. 2003	0.333333	tar–Archive auspacken: Wie packt man tar–Archive aus?...
16.1	29. 09. 2003	0.333333	UNIX–Editoren: Wie schreibt man Texte in UNIX?...

Fig. 2. Retrieval result for the query in Figure 1

The query result is an ordered list of best matching cases. An example of a query and the result of the retrieval is shown in the snapshots in Figure 1 and Figure 2. This case-based retrieval mechanism is more precise than a simple string matching, for instance. To ensure the quality of the retrieval results over time, the dictionaries with information entities and local similarity values have to be maintained when some new cases have been written. Our experiences have shown that 'non-CBR experts' quickly understand the dictionaries and that the time effort for filling them is between 30 seconds and 10 minutes per new case. So, this text comparison method lies in a good
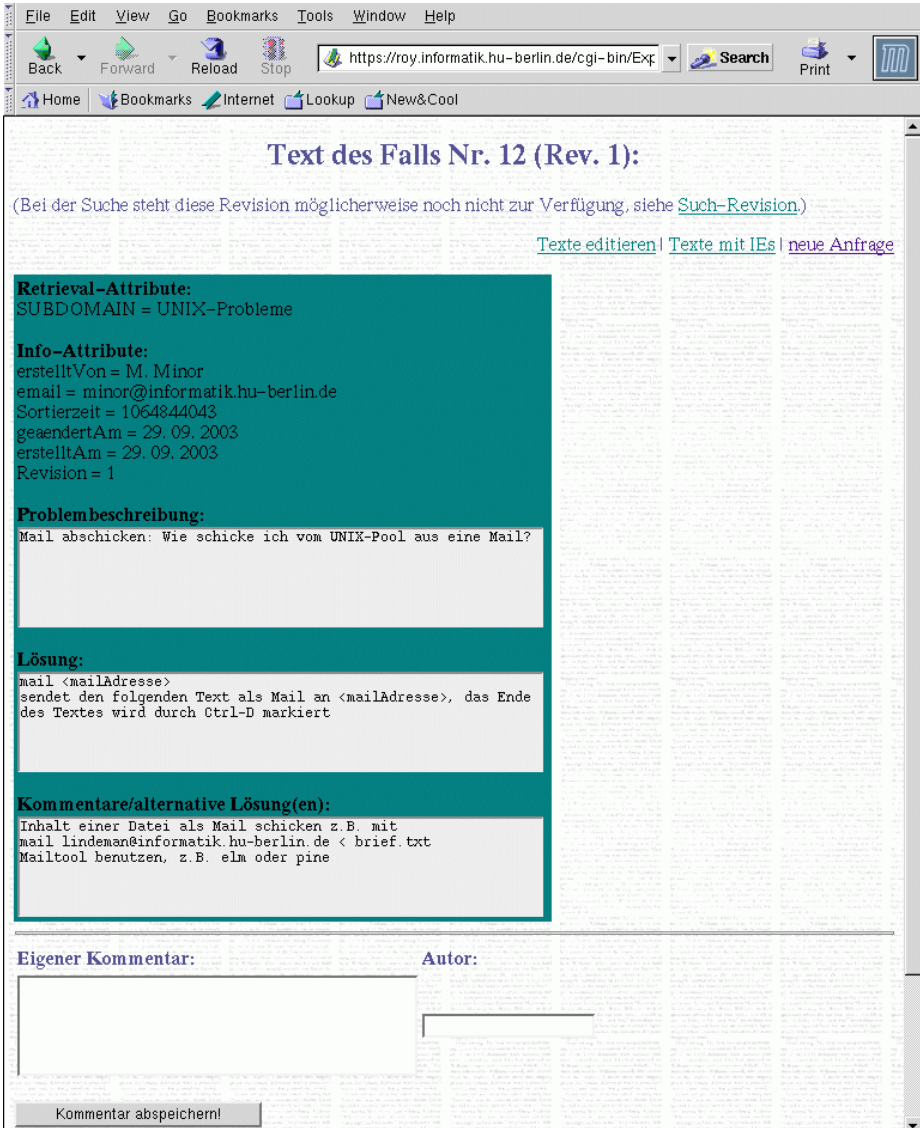
Fig. 3. Sample case of Table 1 as it is presented to the user

balance between a low-effort modelling process and a satisfying solution of the text comparison problem that is independent on the actually used terms (the so-called 'paraphrase problem'). Before the system waits for queries, the texts are mapped on sets of information entities and linked in the net of local similarity relationships. With this pre-compiling, the retrieval is very quick and has a performance of less than a millisecond.

3   System Introduction and Motivation Strategy

We followed the principle of user participation in all developing phases of the system: before the system has been started, during the first weeks of usage, and after the system has failed in the first turn and has been established in a second turn with a new generation of users. This principle led us to four different kind of activities:

- Some of the users helped us to prepare an initial case base.
- All potential users were informed by advertising activities.
- The system works with intrinsic motivation and in a privacy-keeping manner only.
- The system is embedded in an environment that aims to stimulate communication.

The system has been filled with some dozens of initial cases to motivate the users to ask queries from the beginning. These cases have been created from three different sources:

1. From teaching material to a lecture on practical informatics from the year before
2. From the Web pages of the system administration group of our institute
3. From the results of a writing session of second-year students after a presentation of the project

Secondly, the potential users of the system have been informed on several channels: per email, per link from the Web page of the lecture on practical informatics, and via face-to-face communication in a meeting of the students' self-administration and in the lecture. The access statistics showed that the face-to-face advertisment has been the most successful. The results of a feedback questionnaire (see Section 4) has confirmed this observation.

The retrieval is anonymous to avoid an atmosphere of being controlled by lecturers. We decided to abstain from extrinsic motivation like giving extra scores for writing a case, for instance. The usage of the system and the authoring of new cases is voluntary, the motivation is intrinsic and has to be done by convincement only.
We aim to stimulate the communication between the users:

- All users can read and write cases. The access is restricted to institute members.
- Cases can be commented, e.g. extended with an alternative solution.
- Cases can be edited, i.e. false information can be deleted by each person detecting it. A copy of all revisions of the case base is dumped to be able to restore cases that have been damaged by uncautious or malicious users.
- The case format includes email addresses for contacting the authors.
- The case-based system is integrated with a discussion forum (see below).

The discussion forum is situated on the Web page of the lecture on practical informatics that has to be visited by many first-year students. The ExperienceBook can be queried directly from the forum's user interface via a simple text field as search mask. A CGI script transforms the search text into a query by extending it with the attribute-value pair 'SUBDOMAIN=PI1 problems' and sends it to the ExperienceBook. The retrieval result is presented on the ExperienceBook's usual result page.

4   Feedback Results, First Extensions, and Second Turn

We took two means to measure the success of the initiative and to improve the quality
of the system: The access statistics from the Web log files (see Table 2) and the
feedback of the users. Some weeks after the introduction of the ExperienceBook II at
the end of October 2003, we could see that the system works in principal and was
queried frequently. However, the plan to let the community members contribute new
case knowledge failed. Very few new cases had been typed in. Only two authors
wrote cases without being explicitly asked to do this. This was surprising as we
made good experiences with a similar system, the ExperienceBook I, which is in use
at the AI lab successfully. The ExperienceBook I gets regularly new cases by the
members of our lab. After some more weeks, the usage rates were decreasing, too. In
opposite to this, the discussion forum 'survived' until July 2004. 60 different authors
produced 33 entries with one to 14 contributions each.

Table 2. Access statistics of the first turn of the ExperienceBook II

	# accesses	# queries	# new cases
October 2003	291	9	19
November 2003	1925	369	10
December 2003	289	43	3
January 2004	299	44	0
Oct. 2003 – Jan. 2004	2804	465	32

Additionally to the access statistics, we got oral feedback from discussions with
two initiatives of the students some weeks after introduction of the ExperienceBook
II. Furthermore, we sent a questionnaire to 298 students in January 2004. The return
rate was low (15 of 298 students, i.e. about 5%), but rather informative. The target
community had installed an own discussion forum meanwhile that was stored outside
the university. We analysed what had happened and draw our conclusions for the
second turn.

Still during the first turn of the system, we extended the case base by new cases
generated from the entrie of the discussion forum and, secondly, installed an additional
Web page for open questions derived from the Web log files of the query page. But
this did not help much and we assume that the main problem were psycho-social
barriers. We took care on two potential psycho-social barriers for the second turn:

1. *The power relationships*: We told the new generation of students several
   times that the lecturers don't read the students' cases and comments.
2. *The access from the normal working context*: We placed a link to our system
   on the 'GOYA' course management Web page that is used by all students
   nearly daily to download exercises, look up credits and so on.

The success of the second turn confirms this assumption, but can not proof it in
general. Since October 2004, the system is accessed constantly about 1,500 times per
month. Nearly 500 accesses come from the GOYA Web page. The number of new
cases written by students is still low (about one or two per month), but lecturers are
contributing new cases so that we have over 50 cases meanwhile. This number is, of
course, still improvable.

5 Discussion and Related Work

In this paper, we presented a holistic approach of a case-based EM project that integrates technical, organisational and psycho-social aspects. We extracted the following success factors and drawbacks:

- It worked well to integrate a case-based system with a discussion forum. 227 of 421 analysed queries from the first turn came directly from the lecture's forum page. A few forum contributions could be transformed into new cases for the case base. [Morgan *et al*., 2003] makes similar observations.
- Advertising is necessary, face-to-face motivation was the most successful: The access rate was significantly increasing for some days after each oral presentation.
- Participating the users was useful to improve the system. Discussions with two initiatives of the students led to the creation of an additional Web page for open questions gained from the Web log files.
- Both, advertising and user participation helped to overcome psycho-social barriers and to create:

  - A well-informed and positive attitude of many users towards the system,
  - A basic trust in the use of the system concerning power relationships and misuse,
  - A direct access from the usual working context, as we learned where to reach the users.

- The intrinsic motivation of the community worked partly. The students asked many queries but wrote few cases in both turns. [Wikipedia, 2004] is a prominent example for the success of the voluntariness principle. In the questionnaires, the ExperienceBook students asked for more cases from the lecturers. This or extrinsic motivation might be countermeasures for the low authoring rate.
- The possibility to evaluate single cases in the questionnaire has not been used at all. [Nick *et al*., 2003] come to similar results. For the purpose of acquiring case-related feedback, we think about an automatic observation of the navigation behaviour (see [Berendt *et al*., 2003]).

References

[Bergmann, 2002] Ralph Bergmann. *Experience Management Foundations, Development Methodology, and Internet-Based Applications*. LNAI 2432, Springer Verlag, 2002.

[Berendt *et al*., 2003] Berendt, B., Hotho, A., Mladenic, D., van Someren, M., Spiliopoulou, M., Stumme, G. (Hrsg.): *Proceedings of the First European Web Mining Forum*, Ruder Boskovic Institute, Zagreb, Croatia, 2003, Workshop at the 14th European Conference on Machine Learning / 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD-2003), Cavtat, Croatia, September 22nd, 2003 http://km.aifb.uni-karlsruhe.de/ws/ewmf03

[Lenz and Burkhard, 1996] Mario Lenz, Hans-Dieter Burkhard. *Case Retrieval Nets: Basic Ideas and Extensions*, In: Görz, Hölldobler (eds.), Proceedings of the KI'96, LNAI 1137, pages 227 -- 239, Springer, 1996.

[Lenz *et al*., 1998] Mario Lenz, André Hübner, Mirjam Kunze. *Textual CBR*, In: Lenz, Bartsch-Spörl, Burkhard, Wess (eds.), Case-Based Reasoning Technology –From Foundations to Applications, LNAI 1400, Springer Verlag, 1998.

[Minor, 2001] Mirjam Minor. *Experience Management – Case-Based Reasoning for Knowledge Management*, In: Czaja (ed.), Proceedings of the CS&P'01, pages 150 -- 159, Warsaw, 2001. online version http://www.informatik.hu-berlin.de/~minor/Publications/CsuP01.ps.gz.

[Morgan *et al*., 2003] Alexander P. Morgan, John A. Cafeo, Diane I. Gibbons, *et al.   The General Motors Variation-Reduction Adviser: Evolution of a CBR System,* In: Proceedings of the ICCBR'03, LNAI 2689, pages 306 -- 318, Springer Verlag, 2003.

[Nick *et al*., 2003] Markus Nick, Klaus-Dieter Althoff, Andreas Jedlitschka. *Acquiring Knowledge for Linking Software Engineering Experience Maintenance with Evaluation*, Proceedings of the Net.ObjectDays, pages 93 – 107, Ilmenau, 2003.

[Wikipedia, 2004] http://www.wikipedia.org (last visited November 2004)