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.

K.-D. Althoff et al. (Eds.): WM 2005, LNAI 3782, pp. 284 – 292, 2005.

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>** **<RETRIEVAL_ATTRIBUTES>** SUBDOMAIN=UNIX **</RETRIEVAL_ATTRIBUTES>** **<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>** **<SOLUTION>** mail <mail_address> from command line sends the following text to <mail_address>, end of input with Ctrl-D **</SOLUTION>** **<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>**	**CASE_IDENTIFIER** = 12 **IE_LIST_RATTR** = [(SUBDOMAIN, UNIX)] **IE_LIST_DESCRIPTION** = [SEND, EMAIL, UNIX POOL] **IE_LIST_SOLUTION** = [MAIL, COMMAND LINE, SEND, TEXT, END, INPUT, CTRL-D] **IE_LIST_COMMENTS** = [SEND, CONTENT, FILE, EMAIL, MAIL, MAIL TOOL, ELM, PINE]