Experience Retrieval in Learning Software Organizations: Do you find what you are looking for?

Raimund L. Feldmann[*], Jörg Rech[†], Alfon J. Wenzler[*]

[*] Fraunhofer Center Maryland, 4321 Hartwick Road, Suite 500,
College Park, MD 20742, USA
rfeldmann@fc-md.umd.edu, alfon@wenzler.info

[†] Fraunhofer IESE, Fraunhofer Platz 1,
67663 Kaiserslautern, Rhineland-Palatine, Germany
rech@iese.fraunhofer.de

Abstract. A Learning Software Organization (LSO) often supports such information retrieval needs by installing an Experience Base (EB) or another Knowledge Management (KM) system. However, even in such environments, searching for the right information can still be a challenge. In this paper, we identify some of the reasons for these challenges. Based on our observations we provide requirements to improve existing EB or KM systems, and to guide developers in implementing better system from scratch. Our observations are based on personal experience with implementing and running EB / KM systems. In addition, a study focusing on search requirements and habits was conducted in a small research center.

1 Introduction

Ever had that feeling that you are looking for a piece of information but don't know how to find it? If you are working in a Learning Software Organization (LSO), you may first, check your intranet using some of the sophisticated Knowledge management (KM) tools or Experience Base (EB) systems which are available. If these repositories do not satisfy your information need, you then may try one of the desktop search engines (e.g., [5], [16], [1]) to search the unstructured information on your desktop, shared drives, old company archives, or other parts of your organizations intranet. Finally, if you still don't find the information in the intranet of your LSO you may even go extern and try the sources of the WWW. A variety of search engines offer these services – however, the results are often poor. So at the end of the day for many people the old U2 song: "*I still haven't found what I'm looking for*" [15] becomes a bitter reality.

Not only to prevent this waste of time or to avoid such unpleasant feeling, it is more and more important for any LSO to offer easy and effective access to the information that is needed by its employees. According to [14] "*The information-based economy is in danger of drowning in a sea of irrelevant, unstructured data.*" In the same report a study is presented according to which more than 65% of the people

agreed that finding information to do their job is difficult. Furthermore, over 50% of these people spend two or more hours each day searching for information. After more then a decade of KM and even with the most sophisticated search engines it still seems to be a challenge to find the information we need in time. Either we retrieve the wrong information, too many, too less, too late, or none at all.

In this paper we identify some of the existing challenges for a LSO to provide the needed information to its employees. First, we give a short overview of background and related work in Section 2. Then we document our experiences with existing EB and KM systems (Section 3). Common shortcomings and problems are summarized before we present the findings of a small user study in Section 4. Based on these results we formulate a set of requirements for EB and KM systems to overcome some of the identified shortcomings (Section 5). These requirements may help in improving the existing systems and their acceptance or provide guidelines for developing new systems in the future. Finally, Section 6 summarizes the paper and gives some conclusions.

2 Background & Related Work

There is an immense rise of electronically available data. Continuous advancement of data base technology as well as of new requirements for the usage of this data results in more and more data sources that store potentially relevant information. These data sources come in all shapes and flavors such as data bases, file systems, archiving systems, application systems, web servers, email systems, digital libraries, or information retrieval systems. *Uncontrolled growth* of the data as well as the data sources produces a heterogeneous mixture of data sources in the organization that differ by their functionality, speed, and technology.

Today, we speak of the *information flood* that hinders our ability to retrieve, reuse, and comprehend information in time. Different technologies used to realize these data sources prevent the standardized and coherent retrieval, processing, and representation of the data within. This *heterogeneity of the data* sources makes their unification and the common storage of the data more difficult. On the one hand old data (i.e., legacy data) and services (i.e., legacy services) are to be maintained and improved while on the other hand new data has to be administered more efficiently and integrated into the existing data source.

2.1 Information Retrieval and Information Integration

Achievements in the fields of information retrieval, information integration, and semantic desktop search environments offer new opportunities to exploit the information available in an organization in order to support these software engineers.

In information integration at least two main concepts are differentiated: On the one hand the concept “universal storage” contains approaches that are concerned with the collection of information and its storage in one location. This is a concept that might be used in a controlled environment but is problematic with large amounts of data and heterogeneous environment such as the Internet. On the other hand stands a concept

called “universal access” that contains approaches concerned with the translation of queries and results in order to integrate the data sources. The following approaches are part of this concept [12]:

- Mediator Systems: These systems mediate or translate between two “query languages”. For example, wrappers encapsulate a data source and translate incoming queries in a standardized format (e.g., SQL) to the special format of the encapsulated data source (e.g., a file system).
- Federated Databases: Based on databases as it uses either a multi-database language that’s “spoken” by all integrated data sources (e.g., SQL) or includes the integration of schemata of all data sources.
- Common Interface: The client is build to understand every search mechanism of every data source that should be used in the information retrieval activities.
- Common Gateway: The client only understands a subset of the search mechanism of the data sources. So-called gateways bundle and integrate the remaining search mechanism (e.g., a SQL gateway that translates the query for the available SQL-databases).
- Common Protocol: The client(s) and data sources are both encapsulated by a wrapper that translates into a common protocol and produces / “re-translates” search results.

2.2 Semantic Desktop Search

Desktop search engines (e.g., Google Desktop search [5], [16]) are currently a research trend that enables the retrieval of information from documents on one’s own desktop. Another concept, the Semantic Desktop Search Environments [7] promises to improve the retrieval and reuse of information distributed in the data source of a company and take it one step further to a LSO.

While desktop search engines enable search based on an index including every term from all documents of the computer, semantic desktop search engines use additional semantics in form of metadata provided by the users, data sources, or additional ontologies.

Our requirements for EB and KM systems will turn the systems into the direction of a semantic desktop environment (such as the Memex [3]) for knowledge workers and especially software engineering in a LSO. It will not only integrate documents form the personal desktop but also include data sources from the intranet of the LSO as well as specific (e.g., role-oriented) data sources from the internet.

3 Experiences With Today’s EB And KM Systems

In the following we identify some search and retrieval issues with today’s EB and KM systems. Our observations are mainly based on first-hands experiences with systems that we developed for industrial customers and partners (e.g., indigo [1], ESERNET [11], SFB-EB [9], VSEK [10], or DoD Acquisition Best Practice

Clearinghouse [6]), but also include studies of such systems based on literature reviews.

3.1 The System Functionality

EB and KM systems often offer a variety of different and sometimes very complex access interfaces. Especially when the stored amount of information is relatively small these offered functionalities are rarely necessary. If a system is used (at all), users tend to access and search the systems content through always the same interface. Frequently, this is the most simple and easy one to use.

To be accepted, EB and KM systems have to be regarded as useful by their users. On the one hand this includes that the systems are easy to use. On the other hand it is required that all information can be found with the system. The later sometimes is hard to achieve when the latest project information has not been included yet.

3.2 The Content of the Systems

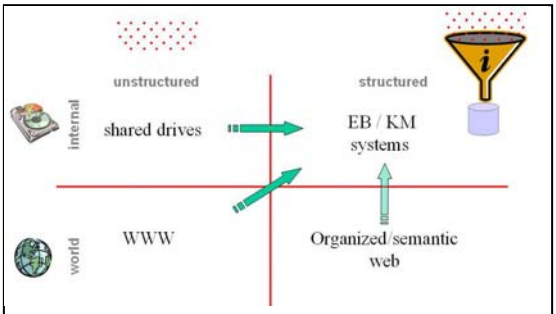

Fig. 1. Information sources and EB / KM systems

When a new EB or KM system is initially implemented it needs to be filled with information. The so-called initial seeding of a system is crucial to its acceptance. On the one hand, it is usually not possible to initially represent and store solutions for all circumstances in the repository. On the other hand, a critical mass of validated and useful information must be available to support the intended users. If they start using the system and frequently do not find anything for their daily work, they will not accept, use, or support the system. For the same reasons, a LSO has to constantly maintain the system content to keep it up-to-date.

As indicated in Figure 1, there are different sources for filling EB or KM systems. In most cases the majority of information comes from the organizations existing internal data sources (e.g., shared drives or data bases). From the point of the EB or KM system, these sources usually can be regarded as unstructured, and therefore need to be processed and transferred into the system. (see arrow from left to right in Figure 1). But not only internal sources can be used to fill or up-date a system. Other sources of information might be integrated into the system from the WWW, public-archives, the semantic web, or specific knowledge brokers (see other arrows in Figure 1). Again, from the viewpoint of the EB or KM system most of these sources can be regarded as unstructured and need to be integrated. In some cases specific external sources can be regarded as structured because the system was designed to deal with those structures (e.g., a part of the semantic web or data available from a knowledge broker) – however, these cases are exceptions and not the rule. Because of these necessary transformations and the existing amount of data, EB or KM system

usually cannot be turned into “universal storage” systems. Only selected information can be and should be transformed and integrated.

4 A User Study

To gain more information about the (Internet) search behavior and strategies of users we conducted a quick study during the summer of 2005. A total of 18 members of a small IT organization in the US filled in our questionnaire. To get further inputs we also made the questionnaire available through one of our web-sites. Since we only received two complete data sets, we decided not to use the data collected through the WWW. However, a quick analysis showed that the data of these two anonymously collected questionnaires showed results similar to our findings.

4.1 Tool Preferences

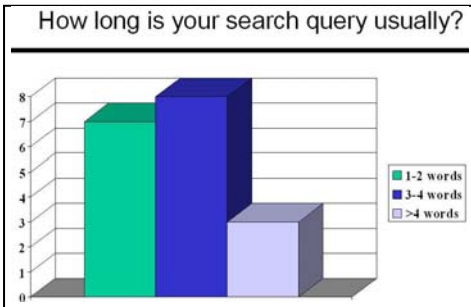

Fig. 2. (Initial) Search query length

Users tend to use only one search tool. With the results of this search tool they are more or less satisfied – even though almost all users have some improvement requests for their preferred tool. Examples for this improvement requests are: Better highlighting of the original search text in the results or a possibility to customize the query based on a personal profile. Additional / other search tools seemed to be only used to cross check results or when the preferred search tool fails to bring the desired results.

Advanced search functionalities (e.g., such as logic operations, similarity search, or typing) are sparely used. This matches our findings regarding the number of search terms used in a query. Most search queries, especially the initial ones, are short (see Figure 2 for details).

4.2 Search Results

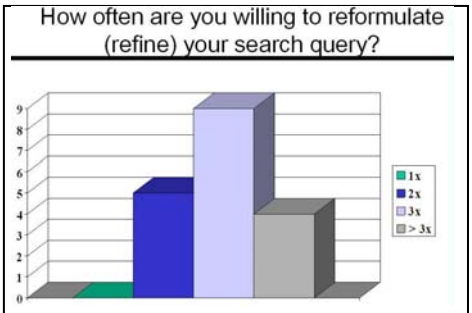

Fig. 3. Query reformulation / refinement

Seven out of 18 users expect to find satisfying results in less then three minutes. Only three out of 18 are willing to search longer then five minutes; but only, if they are looking for important information to finish a work task. These results match the results in Figure 3, according to which

only four out of the 18 users are willing to refine or reformulate their search query more then three times. Consequently, only eight out of 18 users are willing to make more then ten mouse clicks to get their final search results.

5 Extended Search Functionalities For EB And KM Systems

By taking a look at information retrieval theory [4], there seem to be two approaches to improve the actual search results for users of EB and KM systems: 1) By integrating more information sources into our EB and KM systems (upper branch in Figure 4), and 2) by manipulating the users' search queries (lower branch in Figure 4).

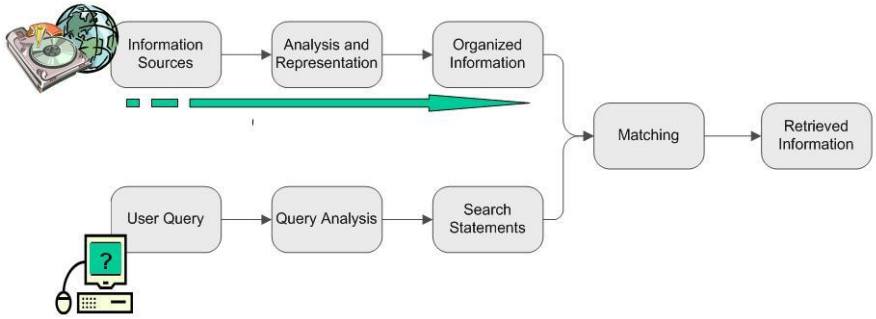

Fig. 4. General outline of an Information Retrieval System (IRS)

5.1 More Sources

As we described in Section 3, there are several sources to fill EB / KM systems with content. For most of these sources an integration process is necessary to match the information provided to the specific EB / KM system content. However, there always will be some information that initially is not worth the effort to be *fully* integrated into our systems. Nevertheless, such information may become relevant for users in the future.

Instead of receiving no search results at all, most users should have access to such information without the need to (re)formulate a new search query with another tool (e.g., a desktop search engine or WWW search tool). Remember, users seem not to like the idea of reformulating search queries and want their results quickly (see user study in Section 4). In addition, they tend to use only one search tool – so we better make sure that users are satisfied with our EB / KM systems, and accept them as their primary tools for *all* search requests.

Figure 5 illustrates the required extended search capability for EB / KM systems. User will issue *all* their search requests to the EB / KM system. Primarily the EB / KM system processes the search request itself. If it is not able to return matching results from its own content, the system automatically issues secondary queries to

other systems in the Inter- or Intranet. The results of these secondary queries are delivered to the user through the EB / KM system, but should be clearly marked as system external results.

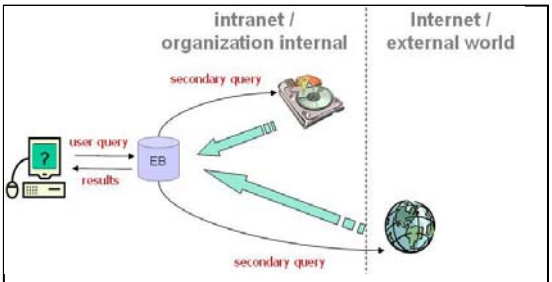

Fig. 5. Extended EB / KM system search functionality

In addition, the same mechanism allows users to compare and check the EB / KM system results with other results, namely from the secondary queries, without leaving the system (compare to usage of additional / other systems in Section 4.1). Initially, it is not be necessary to fully integrate these results into the EB / KM system; this can be done later using the standard processes if necessary.

5.2 Query Manipulation

According to our model of an information retrieval system a user query is analyzed and transferred before it is used to match the structured and organized information. (see lower branch in Figure 4). Obviously, query manipulation seems to be necessary to submit secondary queries to external systems. But even user queries directly addressed to our EB / KM systems can be manipulated and improved.

Probably the best known tools for query manipulations are spelling checkers and thesauruses. Such tools and algorithms can be used to proactively extend or correct the user queries and are especially helpful if a full text search interface is offered as part of the EB / KM system. More sophisticated tools for query manipulation are semantic networks [13]. Such tools can expand short user queries (see Figure 2) based on additional information, such as user profiles or a specific domain. Since users tend to formulate only short search request, it is important to get the most out of these terms and automatically improve the quality of the user queries. This aspect is frequently missing in today's EB / KM systems and needs to become a standard requirement for future implementations.

6 Summary and Conclusions

Our quick study showed that users tend use only one to four search terms and at most reformulate their query three times. This lack of time and patience supports the problem of the increasing information flood. Consequently we outlined a framework for the integration of *all* search activities in a LSO that has the potential of improving the usability, reliability, and performance of today's EB / KM systems. We hope our research ideas shed some light on the potential and benefit such extended EB / KM systems will have for a LSO. A first prototype instantiation of this framework is currently implemented and will be soon available for further user studies. Future research will include the tailoring of this framework for specific roles of users in a

LSO (e.g., a requirements engineer or a programmer). Furthermore, data mining techniques [8] might be used on top of this architecture to discover previously unknown knowledge (e.g., associated experiences) or to cluster similar knowledge that might be aggregated to more general knowledge as in the knowledge dust to pearls approach [2].

Acknowledgments

Part of this work has been conducted in the context of the project RISE, indiGo, ESERNET, VSEK, and SFB. We appreciate the support of our colleagues Mikael Lindvall and Iona Rus from the Fraunhofer Center Maryland and from Stefan Topp and Benjamin Lux who used to be interim students at the FC-MD.

References

[1] K. D. Althoff, K. U. Becker, B. Decker, A. Klotz, E. Leopold, J. Rech, and A. Voss: "The indiGo project: enhancement of experience management and process learning with moderated discourses," in Data Mining in Marketing and Medicine, P. Perner (Ed), Berlin, Germany, Springer Verlag, 2002, pp. 53-79.

[2] Basili, V. R., Costa, P., Lindvall M., Mendonca, M., Seaman, C., Tesoriero R. and Zelkowitz, M. (2001). “An Experience Management System for a Software Engineering Research Organization”. Maryland.

[3] V. Bush: As We May Think. The Atlantic Online, vol. 176, no. 101-108, 1945.

[4] G. G. Chowdhury: Introduction to Modern Information Retrieval. 2nd Ed., facet publishing, 2004.

[5] T. Claburn and T. Kontzer: Heart of the search [Google Desktop Search]. *InformationWEEK, USA* (1040), 2005, pp. 18-20.

[6] K. Dangle, L. Dwinnell, J. Hickok, and R. Turner: Introducing the Department of Defense Acquisition Best Practices Clearinghouse. CrossTalk, May 2005, pp. 4.

[7] S. Decker, J. Park, D. Quan, and L. Sauermann: Workshop2: The Semantic Desktop - Next Generation Information Management and Collaboration Infrastructure. The International Semantic Web Conference (ISWC 2005), Galway, Ireland, 6. Nov. 2005.

[8] U. Fayyad, S. G. Piatetsky, and P. Smyth, "From data mining to knowledge discovery in databases," AI Magazine, vol. 17, pp. 37-54, 1996.

[9] R. L. Feldmann: On developing a repository structure tailored for reuse with improvement. Workshop on Learning Software Organizations (LSO 1999), Kaiserslautern, Germany, 1999.

[10] R. L. Feldmann and M. Pizka: An on-line software engineering repository for Germany's SME - an experience report. Advances in Learning Software Organizations. 4th International Workshop (LSO 2002), Chicago, IL, USA, 6 Aug. 2002.

[11] A. Jedlitschka and M. Ciolkowski: Towards evidence in software engineering. The International Symposium on Empirical Software Engineering, Redondo Beach, CA, USA, 19-20 August 2004.

[12] U. Leser: Query planning in mediator based information systems. PhD Thesis, Fachbereich 13 – Informatik, Technischen Universität Berlin. Berlin, http://informatik.hu-berlin.de/~leser/publications/dissertation00.pdf, 2000, 195 pages.

[13] Semantic-Networks. Artikel about Semantic Networks. Retrieved 1.12.2005 from http://en.wikipedia.org/wiki/Semantic_network, 2005.

[14] Taxonomy & Content Classification – Market Milestone Report. Delphi Group, Boston, MA, USA, 2002.

[15] U2 (artist): The Joshua Tree (album), 1987.

[16] D. Winder: Desktop detectives [desktop search]. *Information World Review* (213), 2005, pp. 19-21.
[17] X1 Technologies: X1 Desktop Search Platform, http://www.x1.com.